



LIBRO DE PONENCIAS

15-17 de Septiembre de 1999
Universidad Carlos III de Madrid
Leganés (Madrid)

Editores:

A. Azcorra Saloña
C. Delgado Kloos
J.I. Moreno Novella
L. Sánchez Fernández

II Jornadas de Ingeniería Telemática



II Jornadas de Ingeniería Telemática JITEL'99

Libro de Ponencias

15-17 Septiembre 1999
Universidad Carlos III de Madrid
Leganés (Madrid)

Editores:

Arturo Azcorra Saloña
Carlos Delgado Kloos
Jose Ignacio Moreno Novella
Luis Sánchez Fernández

Organizadores:



Area de Ingeniería Telemática
Universidad Carlos III de Madrid



Dpto. Ingeniería de Sistemas Telemáticos
Universidad Politécnica de Madrid

ISBN: xxxxxxxxxxxxxxxxx

Depósito Legal: M-xxxxx-1999-07-29

Impreso por ANCARES artes gráficas, s.l. Avda del Petróleo 6. 28918 Leganés (MADR)

Comité de Programa

Antonio Alabau Muñoz	Univ. Politècnica de València
Javier Aracil Rico	Univ. Pública de Navarra
Arturo Azcorra Saloña	Univ. Carlos III de Madrid
Julio Berrocal Colmenarejo	Univ. Politècnica de Madrid
Melquiades Carbajo Martín	Univ. de Alcalá
Carlos Delgado Kloos (presidente)	Univ. Carlos III de Madrid
Victor M. Carneiro	Univ. da Coruña
Justo Carracedo Gallardo	Univ. Politècnica de Madrid
Yannis Dimitriadis	Univ. de Valladolid
Guillem Femenias Nadal	Univ. de les Illes Balears
Iñaki Goiricelaya Ordorica	Euskal Herriko Unib.
Klaus Hackbart	Univ. de Cantabria
Javier López Muñoz	Univ. de Málaga
Manuel Ramos Cabrer	Univ. de Vigo
Jose Ruiz Mas	Univ. de Zaragoza
Emilio Sanvicente Gargallo	Univ. Politècnica de Catalunya
Alvaro Suárez Sarmiento	Univ. de Las Palmas de Gran Canaria
Juan Manuel Vozmediano Torres	Univ. de Sevilla

Comité de Organización

Laura Acebes	Universidad Carlos III de Madrid
David Larrabeiti López	Universidad Carlos III de Madrid
Natividad Martínez Madrid	Universidad Carlos III de Madrid
Tomas de Miguel Moro	Universidad Politècnica de Madrid
Jose Ignacio Moreno Novella (co-presidente)	Universidad Carlos III de Madrid
Luis Sánchez Fernández (co-presidente)	Universidad Carlos III de Madrid
Victor Abraham Villagrà González	Universidad Politècnica de Madrid

Lista de Revisores

Alberto Domingo Ajenjo, Univ. de Alcalá
Alberto García, Univ. de Cantabria
Alberto García Martínez, Univ. Carlos III de Madrid
Alberto Izquierdo Fuente, Univ. de Valladolid
Alvaro Suárez Sarmiento, Univ. de Las Palmas de Gran Canaria
Ana Gómez Oliva, Univ. Politécnica de Madrid
Andrés Marín López, Univ. Carlos III de Madrid
Ángel Fernández del Campo, Univ. Politécnica de Madrid
Ángel García Ramírez, Univ. de Sevilla
Antonio Barba, Univ. Politécnica de Catalunya
Antonio Cuadro, Univ. de Cantabria
Antonio Fernández Coca, Univ. de les Illes Balears
Antonio J. Sierra Collado, Univ. de Sevilla
Antonio Maña Gómez, Univ. de Málaga
Arturo Azcorra Saloña, Univ. Carlos III de Madrid
Aurelio Berges García, Univ. Politécnica de Madrid
Bernardo Alarcos Alcázar, Univ. de Alcalá
Carlos Ángel Iglesias Fernández, Univ. Politécnica de Madrid
Carlos Delgado Kloos, Univ. Carlos III de Madrid
Carlos Enrique Palau Salvador, Univ. Politécnica de València
Carlos García Rubio, Univ. Carlos III de Madrid
Carlos Maraval Lozano, Univ. de Málaga
Carlos Miguel Nieto, Univ. Politécnica de Madrid
Carmen Fernández Panadero, Univ. Carlos III de Madrid
Carmen Guerrero López, Univ. da Coruña
Daniel Morato, Univ. Pública de Navarra
David Larrabeiti López, Univ. Carlos III de Madrid
Eduardo Jacob, Euskal Herriko Unib.
Eduardo Magaña Lizarrondo, Univ. Pública de Navarra
Elsa M. Macías López, Univ. de Las Palmas de Gran Canaria
Emilia P. Belleboni, Univ. Politécnica de Madrid
Emilio Sanvicente Gargallo, Univ. Politécnica de Catalunya
Enrique Vazquez Gallo, Univ. Politécnica de Madrid
Esteve Pallarès Segarra, Univ. Politécnica de Catalunya
Eusebio Fernández López, Univ. de Valladolid
Fernando Bellas Permy, Univ. da Coruña
Fidel Cacheda Seijo, Univ. da Coruña
Francisco Javier Ruiz Piñar, Univ. Politécnica de Madrid
Francisco Valera Pintor, Univ. Carlos III de Madrid
Germán Madinabeitia Luque, Univ. de Sevilla
Gregorio Fernández, Univ. Politécnica de Madrid
Guillem Femenias Nadal, Univ. de les Illes Balears
Ignacio Soto Campos, Univ. de Valladolid
Ignasi Furió, Univ. de les Illes Balears
Iñaki Goirizelaia, Euskal Herriko Unib.
Itziar Alonso González, Univ. de Las Palmas de Gran Canaria
Javier Aracil Rico, Univ. Pública de Navarra
Javier López, Univ. de Málaga
Jesus de Mingo Sanz, Univ. de Zaragoza
Jesús Redondo, Univ. Politécnica de Madrid
Joan Vinyes Sanz, Univ. Politécnica de Madrid
Jorge García Vidal, Univ. Politécnica de Catalunya
Jorge Martínez, Univ. Politécnica de València
Jorge Mata Díaz, Univ. Politécnica de Catalunya
Jose Ángel Gómez Argudo, Univ. de Sevilla
Jose Angel Irastorza, Univ. de Cantabria
Jose Antonio Omedes Capdevila, Univ. de Zaragoza
José F. Díez Higuera, Univ. de Valladolid
Jose Ignacio Moreno Novella, Univ. Carlos III de Madrid
Jose M. Fornes Rumbao, Univ. de Sevilla
José Manuel Arco Rodríguez, Univ. de Alcalá
Jose Ruiz Mas, Univ. de Zaragoza
Josemaria Malgosa Sanahuja, Univ. de Zaragoza
Josep Lluís Ferrer Gomila, Univ. de les Illes Balears
Juan A. Ternero Muñoz, Univ. de Sevilla
Juan Antonio Carral, Univ. de Alcalá
Juan Antonio Rodrigo Yanes, Univ. de Alcalá
Juan Carlos Guerri Cebollada, Univ. Politécnica de València
Juan García Haro, Univ. Politécnica de Catalunya
Juan Ignacio Asensio Perez, Univ. de Valladolid
Juan José Igarza, Euskal Herriko Unib.
Juan Jose Ortega Daza, Univ. de Málaga
Juan Manuel Vozmediano, Univ. de Sevilla
Juan Riera, Univ. Politécnica de Madrid
Juanjo Unzilla, Euskal Herriko Unib.
Julián Fernández Navajas, Univ. de Zaragoza
Justo Carracedo Gallardo, Univ. Politécnica de Madrid
Klaus Hackbarth, Univ. de Cantabria
Koldo Espinosa Acereda, Euskal Herriko Unib.
León Vidaller, Univ. Politécnica de Madrid
Lidia Fuentes Fernández, Univ. de Málaga
Liliana Patricia Santacruz Valencia, Univ. Carlos III de Madrid
Loren Carrasco, Univ. de les Illes Balears
Lourdes López Santidrián, Univ. Politécnica de Madrid
Luis Guijarro Coloma, Univ. Politécnica de València
Luis J. de la Cruz Llopis, Univ. Politécnica de Catalunya
Luis Muñoz, Univ. de Cantabria
Luis Zabala Alberdi, Euskal Herriko Unib.
M^a Victoria Higuero Aperribai, Euskal Herriko Unib.
Manuel Ramos Cabrer, Univ. de Vigo
Margarita Amor López, Univ. da Coruña
María Calderón Pastor, Univ. Politécnica de Madrid
Marifeli Sedano Ruiz, Univ. de Alcalá
Mario Marrero Ruiz, Univ. de Las Palmas de Gran Canaria
Marta Garcia, Univ. de Cantabria
Martín Llamas Nistal, Univ. de Vigo
Melquiades Carbajo, Univ. de Alcalá
Mercedes Garijo, Univ. Politécnica de Madrid
Miguel Soriano Ibáñez, Univ. Politécnica de Catalunya
Mikel Izal Azcarate, Univ. Pública de Navarra
Mónica Aguilar Igartua, Univ. Politécnica de Catalunya
Mónica Gorricho Moreno, Univ. Politécnica de Catalunya
Natividad Martínez Madrid, Univ. Carlos III de Madrid
Oscar I. Lepe, Univ. Politécnica de Catalunya
Pablo Beneit, Univ. Politécnica de València
Pedro Merino Gómez, Univ. de Málaga
Roberto Sanz, Univ. de Cantabria
Rubén de Diego Martínez, Univ. Politécnica de Madrid
Salvador Sánchez Hernández, Univ. Politécnica de Madrid
Teresa Ariza Gómez, Univ. de Sevilla
Toni Bibiloni, Univ. de les Illes Balears
Vicente Luque Centeno, Univ. Carlos III de Madrid
Victor A. Villagra, Univ. Politécnica de Madrid
Víctor M. Carneiro Díaz, Univ. da Coruña
Victor M. Sempere Paya, Univ. Politécnica de València
Xavier Hesselbach Serra, Univ. Politécnica de Catalunya

Presentación

Las Jornadas de Ingeniería Telemática vieron la luz en su primera edición del 15 al 17 de septiembre de 1997, en la Universidad del País Vasco. Se organizaron entonces en el contexto de la celebración del 100 aniversario de esa universidad. Es ahora, exactamente dos años más tarde, cuando se organiza la segunda edición, y también coincidiendo con un aniversario, en concreto, del X aniversario de la Universidad Carlos III de Madrid.

Estas Jornadas pretenden constituir un foro que reúna a la comunidad científica relacionada con el Área de Ingeniería Telemática, un punto de encuentro en el que se muestren los últimos avances en investigación e innovación de los grupos universitarios y las empresas. El Área de Ingeniería Telemática en las universidades españolas ha tenido una fuerte expansión en los últimos años. Estas Jornadas permitirán que los recién incorporados estrechen relaciones con los que ya llevan algún tiempo, que se conozcan los trabajos realizados en los distintos centros para así identificar posibles problemas en común y sus soluciones.

Se recogen en estas actas los artículos aceptados para su presentación en las II Jornadas de Ingeniería Telemática, JITEL'99. El número de artículos enviados superó todas las previsiones: se recibieron 120 artículos, entre ellos muchos de excelente calidad. La labor de selección del Comité de Programa por ello no fue fácil, ya que la duración del congreso estaba prefijada. Finalmente, se seleccionaron 57 artículos para su presentación e inclusión en las actas. Además de seleccionaron otros 39 como pósters. Los restantes tuvieron que rechazarse porque presentaban una idea todavía demasiado preliminar o simplemente porque ya no había forma de encajarlos dentro del programa. Complementan el programa dos tutoriales de figuras destacadas sobre temas actuales, conferencias invitadas y dos mesas redondas, una sobre el futuro del Sector de las Telecomunicaciones y otra sobre temas docentes en el Área de Ingeniería Telemática.

Las Jornadas no se podrían haber llevado a cabo si no hubiese sido por el apoyo y trabajo de muchas personas e instituciones. En primer lugar, quisiera agradecer el patrocinio de Caja Madrid, Banco Santander y Ernst & Young Consultores, patrocinadores de todas las actividades conmemorativas del X aniversario de la Universidad, en las que se encuadra también este evento, y de Madritel, patrocinador de las actividades tecnológicas. Muy de agradecer es también el patrocinio directo de la CICYT (Comisión Interministerial de Ciencia y Tecnología) y de BEA Systems.

Pero además de a instituciones, nos corresponde dar las gracias a personas. En primer lugar a los ponentes –e incluso a los autores de aquellos artículos que no han podido ser incluidos en el programa por falta de espacio–, a los miembros de mesas redondas y a los conferenciantes invitados y tutorialistas. El Comité Organizador –con miembros de las universidades Politécnica de Madrid y Carlos III de Madrid– merece una mención muy especial por el gran trabajo realizado para dar respuesta a los pequeños y grandes retos que supone organizar un congreso como éste. También el Comité Gestor, que fue el germen de las Jornadas. Finalmente, quisiera agradecer la labor realizada por el Comité de Programa, por su participación entusiasta, y a los revisores de los artículos por la calidad de las revisiones.

Esperamos y deseamos que las Jornadas transcurran de la forma más fructífera para todos. Esperamos también que vean su continuidad en el futuro. En realidad estamos convencidos de ello: el gran éxito de participación en esta edición lo demuestra.

Carlos Delgado Kloos
Presidente del Comité de Programa

Tabla de Contenidos

Capítulo 1A: Integración de Servicios

Multiplexores Inversos ATM: Funcionamiento, Configuraciones de Red y Modelo de Prestaciones para la Estimación de la Probabilidad de Pérdida de Celdas.

M. Aguilar Igartua, J. García Haro, M. Postigo Boix (U. Politécnica de Cataluña)

Reserva Eficiente de recursos para canales de reserva en redes en tiempo real.

E. Hernandez, J. Vila (U. Politécnica de Valencia)

Satélites para la Interconexión en Malla de Redes ATM: Arquitectura y Protocolos.

J. Pérez Otero, C. Miguel Nieto, J. Ruiz Piñar (Telefónica I+D)

Gestión Integral de la Red de Señalización de Telefónica: Sistema SERES.

A. Arias Díaz, J. C. Luengo Patrocinio (Telefónica I+D)

Control de tráfico mediante Class-Based Queueing gestionado por RSVP usando Linux como router con capacidad QoS.

A. Gómez Skarmeta, A. López Toledo (U. Málaga)

Capítulo 1B: Sistemas Móviles

Protocolo de handover y sincronización para comunicaciones móviles multimedia.

V. Martínez Barres, V. Casares Giner (U. Politécnica de Valencia)

Red ATM Experimental para Comunicaciones Móviles Multimedia.

J. Martínez, J.R. Vidal y L. Guijarro (U. Politécnica de Valencia)

Macro-Diversidad con Combinación Pos-Decodificación.

M. Gorricho, J. Paradells (U. Politécnica de Cataluña)

Análisis de Throughput y Retardo en un Sistema CDMA Celular de Paquetes con Control de Potencia Imperfecto.

J. Romero Jerez, M. Ruiz García, A. Díaz Estrella (U. Málaga)

Protocolos de Control de Acceso al Medio para Tráfico Multimedia en Redes de comunicaciones Móviles.

L. Carrasco, G. Femenias, I.F. Akyildiz, J. McNair Broadband (U. Islas Baleares)

Análisis de controles de admisión en un sistema CDMA para tráfico heterogéneo.

R. Pérez Sánchez, M. Ruíz García (U. Málaga)

Capítulo 2A: Implicaciones Socio-Económicas e Integración de Servicios

De la Ingeniería Telemática a la Ingeniería de la Sociedad de la Información.

M. Tomás, A. Alabau (U. Politécnica de Valencia)

Parámetros de medida de la calidad de servicio en ISPs y análisis comparativo de proveedores.

A. Ferro Vázquez (U. País Vasco)

ATM Sobre Ethernet Mediante el Protocolo CIF.

J. M. Arco, B. Alarcos, A. M. Hellín, A. García, D. Meziat (U. Alcalá)

Simulador ATM basado en Componentes.

J. M^a Quinteiro, M. Pilar Rosales, Á. Santana, A. Sarmiento (U. Las Palmas)

Funcionamiento y prestaciones del protocolo HARAL (Hybrid Adaptive Random Access Protocol) para redes de cable HFC.

L. Gutierrez y S. Sallent (U. Politécnica de Cataluña)

Control de Acceso en entornos Multicast: Un enfoque a la autenticación de emisores.

A. F. Gómez Skarmeta, A. L. Mateo Martínez, P. Miguel Ruiz Martínez (U. Málaga)

Capítulo 2B: Plataformas Distribuidas y Gestión de Red

Modelo distribuido para la construcción de servicios de red multimedia programable.

J.A. Lozano, M.C. Oria, F. Sandoval (U. Málaga).....

Arquitectura Composicional de Seguridad para Servicios Telemáticos.

L. Fuentes, J. López (U. Málaga)

Una experiencia de vídeo bajo demanda soportada en tecnología web, CORBA y SDL.

R. de Diego, J. Carracedo, C. Ramos, V. Hernández (U. Politécnica de Madrid).....

El papel de los servicios de intermediación en el contexto del comercio electrónico.

V. Villagra, J.I. Moreno, J.I. Asensio, J. Berrocal (U. Politécnica de Madrid)

Mecanismos de comunicación y gestión de servicio de un broker de información multiagente.

F. Valera, J.I. Moreno, V. Villagrà, J. Berrocal (U. Carlos III de Madrid)

Sistema de Gestión de Servicios de Telecomunicación en un Entorno Cambiante.

R. Koehn, C. Guerrero, L. Bordon y G. Fernández (U. La Coruña)

Capítulo 3A: Modelado de Tráfico

TREESOL: Algoritmo para la Optimización de la Capa Física de una Red de Telecomunicación Implementada de Forma Proprietaria.

J.R. Salas Blanco, K.D.-Hackbarth (U. Cantabria)

Modelos y métodos para el diseño y enrutamiento de tráfico de redes de conmutación de circuitos.

A. Cuadra Sánchez, K.D.-Hackbarth (U. Cantabria)

Modelado Markoviano de Tráfico Agregado ATM.

A. Jurado, E. Casilari, A. Reyes, A. Díaz-Estrella y F. Sandoval (U. Málaga)

Modelado Wavelet de tráfico de vídeo VBR

E. Casilari, A. Reyes, A. Díaz-Estrella y F. Sandoval (U. Málaga)

Un estudio por simulación del efecto de la correlación en el retardo medio del tráfico asíncrono en una red FDDI.

J.C. López Ardao, C. López García y A. Suárez González (U. Vigo)

Técnicas eficientes de filtrado y análisis de tráfico para la monitorización continua de redes de comunicaciones.

J.J. Ruiz, E. Magaña, J. Aracil, J. Villadangos (U. Navarra)

Disciplina de Planificación Basada en Rondas con Disminución de la Distorsión del Patrón de Tráfico.

J.M^a Quinteiro, A. Sarmiento, A. Santana, F. Cabrera (U. Las Palmas)

Capítulo 3B: Sistemas Multimedia

Ubicación de Contenidos Multimedia con Velocidad de Representación Variable en Servidores Multidisco.

A. García Martínez (U. Carlos III de Madrid)

Diseño y evaluación de un sistema para el transporte de vídeo en redes industriales mediante MMS.

V.M. Sempere Payá, J. Mataix Oltra, E. Tordera Santamatilde (U. Politécnica de Valencia)

Particionado Hardware/Software para Sistemas Multimedia.

J. P. Castellano, D. Sanchez y A. Suarez (U. Las Palmas)

Sistema Multicanal de Audio Digital sobre protocolo IEEE-1394.

J. Soler Lucas, F.J. Cortés Santaolalla, M.A. Pérez Costero (Instituto Tecnológico de Aragón)

Desarrollo de un Servidor de Vídeo MPEG sobre ATM nativo y JAVA.

J.M. Fornés y J.A. Gómez Argudo (U. Sevilla)

Visión general de las técnicas de marcas de agua (watermarking) y sus aplicaciones.

J. Unzilla, E. Jacob, A. Ferro (U. País Vasco)

TV Interactiva Terrestre - El proyecto ACTS iTTi.

J. A. Carral, F. Scalise, R. Kopp, L. Lauer, O. Souloumiac (U. Alcalá)

Capítulo 4A: Tecnologías de Agentes y Redes Activas

Planificación en Internet basada en una Arquitectura Multiagente.

D. Camacho Fernández, J.M. Molina López, D. Borrajo Millán (U. Carlos III de Madrid).

JavaMAST: Un Entorno de Desarrollo de Agentes Inteligentes.

M. García, M. Garijo, C.A. Iglesias, E. Martín, J.L. Mulas (U. Politécnica de Madrid)

Gestión de recursos en conmutadores mediante agentes inteligentes móviles.

A. Barba Martí (U. Politécnica de Cataluña).....

Principios y Aplicaciones de las Redes Activas.

M. Calderón Pastor, M. Sedano Ruiz, S. Eibe García (U. Politécnica de Madrid)

Especificación y verificación formal de un protocolo de gestión de sistemas de administración de redes distribuido con arbitro.

M. Marrero y A. Suarez (U. Las Palmas).....

Dimensionado del Nodo de Acceso con Control Explicito de Tasa DMRCA para Redes de Paquetes.

X. Hesselbach Serra, S. Sallent Ribes (U. Politécnica Cataluña)

Capítulo 4B: Aplicaciones de Teleeducación, Telemedicina y Diseño de Interfaces

Teleconferencia y Trabajo Cooperativo en Entornos Universitarios.

E. Pastor, J. Carracedo, D. Fernandez, J. A. Sanchez (U. Politécnica de Madrid)

Experiencias en el uso del lenguaje Java para la enseñanza de Ingeniería Telemática.

J.C. Dueñas, G. Huecas, S. Pavón, T. Robles, J. Salvachúa (U. Politécnica de Madrid)

Acceso a bases de datos de imágenes de radiología mediante ActiveX scripting.

J. Alvarez, C. del Barrio, M. de Cabo, E. Fernandez, G. Garcia y M. Rodriguez. (U. Valladolid).....

Sistema de videoconferencia aplicado en telepatología.

N. Pérez, M. de Cabo, Y.A. Dimitriadis, E. Fernández. (U. Valladolid)

Generación de programas gráficos en X Window. Herramienta xgui.

C.M. Ramírez-Casañas, J.R. Ansó Magaña (U. Las Palmas)

Interfaz de Usuario basado en Mundos VRML para un Museo Virtual Tridimensional en Internet.

J.F. Díez Higuera, F.J. Díaz Pernas (U. Valladolid)

Capítulo 5A: Análisis de Tráfico

Algoritmo Distribuido de Reconfiguración de Redes Virtuales ATM basadas en VPs.

M. Nicolau, J. Abella, G. Corral, F. Garcia (U. Ramón Llul)

Arquitectura de Encaminamiento para Servicios Garantizados.

M.C. Oria, J.A. Lozano, A. Ariza y F. Sandoval (U. Málaga)

Análisis comparativo de modelos dinámicos de control de admisión en redes ATM.

M. Fernandez Veiga, C. Lopez Garcia (U. Vigo)

Capítulo 5B: Sistemas Multimedia y Móviles

Evaluación del Rendimiento de un Protocolo de Acceso Múltiple, basado en un TDMA Dinámico.

M. García, A. Ferrándiz, L. Muñoz, R. Sanz (U. Cantabria)

Propuesta de un Protocolo de Acceso P-TDMA para Redes Inalámbricas de Banda Ancha.

L. Muñoz, J. Choque, M. García, R. Sanz (U. Cantabria)

Técnicas de gestión para la mejora del protocolo TCP en entornos radio.

R. Pérez Sánchez, M. Ruíz García (U. Málaga)

Resumen de Posters

Capítulo 1A: Integración de Servicios

Multiplexores Inversos ATM: Funcionamiento, Configuraciones de Red y Modelo de Prestaciones para la Estimación de la Probabilidad de Pérdida de Celdas.

M. Aguilar Igartua, J. García Haro, M. Postigo Boix (U. Politécnica de Cataluña)

Reserva Eficiente de recursos para canales de reserva en redes en tiempo real.

E. Hernandez, J. Vila (U. Politécnica de Valencia)

Satélites para la Interconexión en Malla de Redes ATM: Arquitectura y Protocolos.

J. Pérez Otero, C. Miguel Nieto, J. Ruiz Piñar (Telefónica I+D)

Gestión Integral de la Red de Señalización de Telefónica: Sistema SERES.

A. Arias Díaz, J. C. Luengo Patrocinio (Telefónica I+D)

Control de tráfico mediante Class-Based Queueing gestionado por RSVP usando Linux como router con capacidad QoS.

A. Gómez Skarmeta, A. López Toledo (U. Málaga)

Multiplexores Inversos ATM: Funcionamiento, Configuraciones de Red y Modelo de Prestaciones para la Estimación de la Probabilidad de Pérdida de Celdas.

Mónica Aguilar Igartua, Joan García Haro, Marcos Postigo Boix
E.T.S. Ingeniería de Telecomunicación de Barcelona (U.P.C.)
Departamento de Matemática Aplicada y Telemática
c/ Jordi Girona 1-3, Mod. C3, Campus Norte, Barcelona 08034
e-mail: {maguilar, teljgh, mpostigo}@mat.upc.es

Abstract

The growing bandwidth demand users made, indicates that access to ATM networks is an important issue. The Inverse Multiplexing for ATM (IMA) is a technology that combines multiple T1/E1 links to create a single logical data trunk which bandwidth is the aggregate of the T1/E1 ones, minus a small amount due to overhead. By using this technology, it is possible to have ATM connections with data rates between the T1/E1 (an economical but too slow option) and T3/E3 ones (with high bandwidth, but expensive). Also, it is interesting to develop expressions easy to compute that approximate the main Quality of Service parameters of Inverse Multiplexers (IMUX) to help network planners in the dimensioning of these systems. In this paper, we study an estimate for the Cell Loss Ratio (CLR) measured in the IMUX devices.

1. Introducción.

La frecuente aparición de aplicaciones y servicios que requieren grandes anchos de banda (videoconferencia, telemedicina, interconexión de redes de área local, vídeo bajo demanda, etc.) ha llevado al desarrollo de la Red Digital de Servicios Integrados de Banda Ancha (RDSI-BA). Esta red está basada en el modo de transferencia asíncrono (ATM, *Asynchronous Transfer Mode*). Los usuarios desean acceder a dicha red a un coste razonable y los operadores de red intentan ofrecer acceso a la red de transporte ATM utilizando la infraestructura disponible. Simultáneamente se trabaja en la instalación de nuevas infraestructuras de telecomunicaciones que den lugar a una RDSI de alta velocidad.

Para soportar las nuevas aplicaciones y servicios, la infraestructura más usual con la que se cuenta en la actualidad en lo que respecta a enlaces físicos disponibles, consiste en enlaces T3/E3 (44,736 Mbps/34,368 Mbps) y T1/E1 (1,544 Mbps/2,048 Mbps). Los enlaces T3/E3 ofrecen un ancho de banda considerable aunque difícilmente utilizado en su totalidad por algunos usuarios, y las tarifas que imponen las operadoras para dichos enlaces son aún muy elevadas. En cambio, el coste de contratación de un enlace T1/E1 es más asequible, aunque el ancho de banda que proporciona suele ser insuficiente. Los precios de dichos enlaces dependen de varios factores, como la distancia y el proveedor. Como ejemplo, los enlaces T1 tienen un coste medio aproximado de \$850 mensuales y un E1 de 25 Km de \$2.900 mensuales. Para los enlaces T3 el precio sube a \$7.500 al mes y para un E3 de 25 Km es de \$29.000 al mes [1]. Por otra parte, los enlaces T3/E3,

en la mayoría de los casos tienen sus puntos de presencia y son sólo accesibles desde las grandes ciudades [1, 2]. Dado este escenario, resulta interesante para compañías y usuarios disponer de una solución que ofrezca una capacidad de transmisión intermedia a un coste asequible, contratando varios enlaces T1/E1 en lugar de un enlace T3/E3.

En Julio de 1997, el *ATM Forum* publicó una nueva especificación denominada Multiplexación Inversa para ATM [3], conocida por las siglas IMA (*Inverse Multiplexing for ATM Specification, version 1.0*). Esta normativa explica cómo transportar un flujo de celdas ATM de alta velocidad, de manera transparente al nivel ATM, distribuyéndolas sobre varios enlaces de baja velocidad, y como reconstruir el flujo original en el extremo remoto de la conexión para ser entregado de nuevo a la capa ATM, que lo procesará normalmente. La especificación IMA permite agrupar hasta 32 enlaces sencillos T1/E1 que pueden unirse formando un grupo IMA, alcanzando una tasa agregada (múltiplo de la tasa de un enlace T1/E1) de hasta unos 48/64 Mbps, suficiente para dar servicio a la mayoría de las aplicaciones actuales de banda ancha. De esta forma, la tecnología IMA cubre el salto en ancho de banda existente entre los enlaces T1/E1 y T3/E3, permitiendo utilizar los recursos actuales de manera más eficiente y consiguiendo un ancho de banda más ajustado al volumen de tráfico ATM que los usuarios desean transferir.

Con ello, se pueden ofrecer servicios que requieren una tasa elevada antes de que se despliegue la red ATM global, es decir, con enlaces puramente ATM de muy alta velocidad (155/622 Mbps y superiores).

Además, la utilización de *IMUX* proporciona una migración inmediata a *ATM*, suave y a un coste razonable, puesto que la naturaleza de ambas redes es la misma (transporte de celdas *ATM* previo establecimiento de enlaces lógicos y manteniendo una Calidad de Servicio determinada).

El dispositivo encargado de agrupar los diversos circuitos físicos formando un único enlace lógico se denomina *IMUX* (*Inverse Multiplexer*). El *IMUX* acepta tanto flujos de celdas *ATM* originadas por distintos tipos de fuentes de tráfico, como datos procedentes de redes de área local - véase tráfico procedente de redes *Ethernet* o *Token Ring*- que será previamente adaptado a formato celda. En ambos casos el *IMUX* distribuye las celdas resultantes sobre los enlaces físicos manteniendo la Calidad de Servicio (*QoS*) requerida por cada conexión.

Ante esta situación, nos planteamos el desarrollo de unas expresiones que permitan hallar de manera sencilla el tamaño del *buffer* y el número de enlaces de salida que garanticen los parámetros de Calidad de Servicio contratados por los usuarios. Así, se dispondrá de una herramienta útil para dimensionar los dispositivos *IMUX* de las redes constituidas mediante dicha tecnología. En la Secciones 2 y 3 de este artículo, explicamos el funcionamiento de la multiplexación inversa para *ATM* y las principales aplicaciones de red típicas de la tecnología *IMA*. En las Secciones 4, 5 y 6 se presenta el desarrollo del modelo analítico del dispositivo *IMUX*. La Sección 7 muestra los resultados para el *CLR* (*Cell Loss Ratio*) obtenidos mediante simulación y mediante la aproximación analítica propuesta. Finalmente, en la Sección 8 se presentan las conclusiones más relevantes y las líneas futuras de trabajo.

2. Modelo para el Dispositivo *IMUX*.

Las celdas adicionales añadidas por el *IMUX*, suponen una sobrecarga y comparten los mismos recursos que las celdas de información de usuario. Esta carga añadida debe ser tenida en cuenta al dimensionar los recursos de un *IMUX*. Esto significa decidir qué número de enlaces va a formar el grupo *IMA*, en función del ancho de banda requerido; y cuál es el tamaño del *buffer* adecuado para cumplir con los parámetros de *QoS* contratados, básicamente el retardo medio, el *CLR* (*Cell Loss Ratio*) y *CDV* (*Cell Delay Variation*). La capacidad del *buffer* también debe ser seleccionada en función de la carga soportada por el dispositivo, es decir, la carga debida a las celdas procedentes del nivel *ATM* y la carga adicional debida a las celdas de control del protocolo *IMA*.

Para realizar este estudio, se ha desarrollado una herramienta en lenguaje C++ [6] que permite la simulación, tanto de dispositivos *IMUX* como de sistemas *IMA* extremo a extremo. El análisis se puede llevar a cabo bajo modelos de tráfico diferentes, cubriendo un amplio margen para la carga de tráfico *ATM*, con un número de enlaces

realista y teniendo en cuenta un amplio rango de valores para tamaños de *buffer*.

Asimismo, se ha desarrollado un modelo analítico obteniendo expresiones matemáticas de fácil manejo que ofrecen una aproximación del retardo medio que sufren las celdas en el *IMUX* y del *CLR*. Los resultados obtenidos mediante simulación y los obtenidos del análisis matemático, se han comparado para apreciar la bondad de las estimaciones propuestas. Como primera aproximación, el dispositivo *IMUX* ha sido modelado mediante un sistema de colas de tamaño finito B , con s enlaces de salida idénticos y tiempo de servicio constante, al tener las celdas *ATM* una longitud constante. El tiempo se divide en intervalos de duración fija (igual al tiempo de transmisión de una celda) tomado como la unidad para simplificar el análisis. Se supone también, de forma preliminar, que el proceso que rige la llegada de celdas al sistema es de *Poisson*. De este modo, el dispositivo *IMUX* puede ser modelado mediante un sistema de colas $M/D/s/s+B$ con tiempo de servicio discreto. Aparentemente, los sistemas *IMA* y los sistemas $M/D/s/s+B$ son similares. La principal diferencia es que en un sistema *IMA* real, además de servir celdas *ATM* también se transmiten celdas *ICP*, *Filler* y *SICP*, cumpliendo la normativa expuesta en la especificación [3].

Intuitivamente, es fácil prever que el *CLR* será mayor en los sistemas *IMA* que en el sistema de colas $M/D/s/s+B$, por contar con un extra de carga debida a *overhead*. Por ello se pensó en tres formas diferentes para tratar de hallar analíticamente el *CLR* de los dispositivos *IMA* usando el modelo matemático equivalente en función del número de enlaces s , del tamaño del *buffer* B y de la carga por enlace ofrecida al sistema ρ [6]; éstas son las siguientes:

- Aumentar la carga ofrecida ρ al modelo $M/D/s/s+B$ con respecto al sistema *IMA*, manteniendo constantes tanto el número de enlaces s como el tamaño del *buffer* B . Así, al entrar más celdas por unidad de tiempo con el mismo tamaño de *buffer*, el *CLR* aumentará.
- Aumentar el número de enlaces s del sistema $M/D/s/s+B$ con respecto al *IMUX* real, manteniendo constantes tanto el tamaño del *buffer* B como la carga por enlace, ρ . De este modo, la carga total entrante al sistema (ρs) aumenta, incrementando el *CLR*.
- Reducir el tamaño del *buffer* del sistema $M/D/s/s+B$ con respecto al sistema *IMA* original, manteniendo constantes tanto la carga por enlace ρ como el número de enlaces s , lo que directamente hará aumentar el *CLR*.

La opción c se ha desarrollado en trabajos anteriores [6, 7] y presenta la ventaja de que reduce el sistema de $B+1$ ecuaciones con $B+1$ incógnitas necesario para resolver analíticamente el modelo $M/D/s/s+B$ de forma exacta [8]. La opción b ha sido estudiada

en [9]. Queda fuera del alcance de este trabajo la exposición de ambos métodos. La opción que se ha elegido en este trabajo, ha sido la primera, por resultar interesante el hecho de caracterizar la carga adicional que añade el dispositivo *IMUX*. De esta forma, se tendrá conocimiento minucioso de la cantidad total de carga que es tratada por el dispositivo *IMUX*: tanto la carga neta procedente del nivel *ATM*, como la carga ficticia correspondiente a las celdas de protocolo incorporadas por el *IMUX* (celdas *ICP*, *SICP* y *Filler*). Al decir carga ficticia, nos referimos al hecho de que la carga de control es introducida por el *IMUX* en el momento de realizar su operación, por lo que no ocupa recursos de memoria. Sin embargo, la inserción de dichas celdas causa el almacenamiento y retardo de celdas del nivel *ATM*, incrementando el *CLR*. Este efecto lo hemos reflejado en un incremento ficticio en la carga aplicada. En los apartados que vienen a continuación, se desarrolla el método llevado a cabo para evaluar el *CLR* en los dispositivos *IMUX* en el sentido de transmisión de la información.

3. Carga adicional introducida por el *IMUX*.

Nuestro objetivo es caracterizar toda la carga procesada por el *IMUX*, para así poder utilizar modelos que permitan obtener expresiones matemáticas sencillas para calcular el *CLR* en función de parámetros relevantes del dispositivo como son el tamaño del *buffer*, número de enlaces y carga ofrecida por las celdas del nivel *ATM*.

Como se ha dicho en la Sección 3, el *IMUX* introduce celdas *ICP* y *SICP* de manera fija: 1 de cada 128 celdas es una celda *ICP* y 1 de cada 2048 celdas es una celda *SICP*. El tiempo de servicio de las celdas *ATM* es constante, e igual a un intervalo temporal unitario. El tráfico entrante a un sistema de colas se mide como $A = \lambda / \mu = \rho \cdot s$ Erlang, siendo λ el número medio de celdas que llegan al sistema por unidad de tiempo, ρ la carga por enlace, s el número de enlaces y $1/\mu = 1$ la tasa de servicio. El tráfico equivalente que suponen las celdas *ICP* y *SICP*, es $A_{ICP} = (1/128) \cdot s$ y $A_{SICP} = (1/2048) \cdot s$ Erlang respectivamente. Además falta el término que incorpora el efecto que sobre la carga ρ_{total} supone la inserción de las celdas *Filler* y que más abajo desarrollamos. Así, la carga total procesada por el *IMUX* queda igual a:

$$\rho_{total} = \rho + \frac{1}{128} + \frac{1}{2048} + \rho_{Filler} \quad (1)$$

En la Ecuación (1), y en general en todo el trabajo presente, denominamos como ρ a la carga procedente del nivel *ATM* ($\rho = \rho_{ATM}$). A simple vista, lo lógico sería considerar $\rho_{Filler} = 0$, ya que estas celdas no ocupan memoria, ya que se generan directamente en los enlaces de salida a medida que son necesarias y por lo tanto no consumen espacio en el *buffer* [6]. Sin embargo, la inserción de celdas *Filler* puede ser interpretada como la inserción de

una carga adicional. Es decir, hay instantes en que ocurrirá que en lugar de transmitir por un enlace una celda disponible proveniente del nivel *ATM*, se habrá introducido una celda *Filler* con lo cual dicha celda *ATM* deberá esperar en el *buffer*. Por ello decimos que la inserción de celdas *Filler* tiene un efecto que puede interpretarse como una carga adicional de celdas. Esta carga adicional, es compleja de caracterizar, debido a la existencia de varios instantes en los que se decide su inserción. El grupo *IMA* forma un enlace lógico a tasa *IDCR*, y es a esta velocidad a la que se decide la inserción de celdas. En caso de no haber celda proveniente del nivel *ATM*, se decide la inserción de una celda *Filler* en su lugar. Por el contrario, en un sistema de colas *M/D/s/s+B* es únicamente al final de cada *slot* temporal cuando se conoce el número definitivo de celdas que van a ser servidas. Esta diferencia puede apreciarse en la Figura 1 [6].

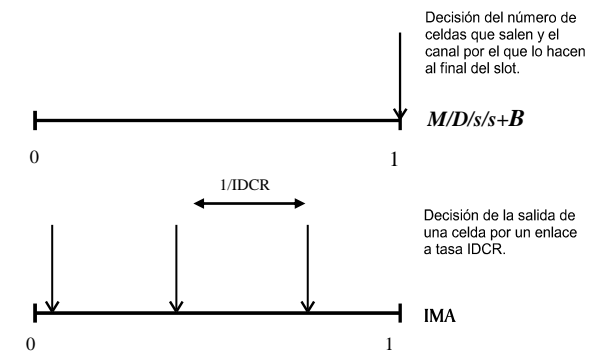
Así, en la Figura 2 puede observarse un ejemplo en el que, si en el primer instante de decisión no hay celdas *ATM* disponibles se decide introducir ya una celda *Filler*, aunque posteriormente durante esa ranura temporal lleguen s o más celdas *ATM*. Por lo tanto, el *CLR* en el dispositivo *IMUX* es mayor que en un sistema de colas *M/D/s/s+B* convencional, ya que puede suceder que haya celdas *ATM* presentes en el dispositivo durante ese *slot* temporal, pero que deberán esperarse en el *buffer*, al haber sido

Figura 1. Instantes de decisión de la emisión de celdas en los sistemas *M/D/s/s+B* e *IMA*.

introducida una celda *Filler* en un enlace de salida, ocupando un lugar que pudiera haber sido utilizado por una celda *ATM*. A efectos prácticos, es como si hubiera una carga ρ extra debida a la inserción de celdas *Filler*, carga que hemos denominado ρ_{Filler} .

4. Diseño del Dispositivo *IMUX* y Aproximación para el *CLR*.

El diseño adecuado de un *IMUX* significa hallar, como parámetros más relevantes, el número de



enlaces requerido, el tamaño de la memoria que garantiza un tiempo medio de espera en el sistema, un *CDV* y un *CLR* determinados y el valor de los umbrales de descarte de celdas en el *buffer* en caso

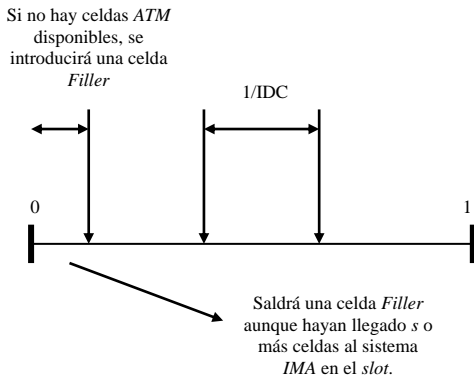
de contemplar prioridades o soportar distintas clases de servicios [10, 11].

El simulador desarrollado permite disponer de una herramienta de evaluación de prestaciones de esta tecnología bajo diferentes condiciones de carga y contemplando todos los parámetros posibles que definen al *IMUX*. La herramienta diseñada está orientada a objeto y permite el estudio de un único nodo (un único dispositivo *IMUX*) o de una conexión extremo a extremo, teniendo en cuenta el grupo *IMA* y el dispositivo *IMUX* receptor [6]. La explicación de la metodología utilizada para el cálculo del retardo medio en los multiplexores inversos y el estudio extremo a extremo quedan fuera del alcance de este documento. Nos centraremos en los detalles del modelo matemático equivalente para el cálculo del *CLR* en el dispositivo *IMUX* transmisor [12]. En las simulaciones realizadas se ha evaluado la probabilidad de pérdida medida como *CLR* y el retardo medio que sufre una celda desde que ingresa en el sistema hasta que lo abandona. La carga neta (*ATM*) por enlace varía del 50% al 90%, el tamaño del *buffer* de entrada al *IMUX* es también variable y el número de enlaces que forman el grupo *IMA* oscila entre 2 y 8, que son los valores más comunes en los dispositivos *IMUX* comerciales (ya que utilizar un número superior de enlaces deja de ser rentable frente al coste de un enlace *T3/E3*).

Figura 2. Introducción de una celda *Filler*, aunque durante el slot temporal lleguen celdas *ATM*.

Para obtener el modelo matemático que caracteriza el comportamiento del *IMUX*, se ha comparado el sistema *IMA* simulado con el sistema *M/D/s/s+B* [11, 13] al que se introduce la carga equivalente ρ_{total} expresada en la Ecuación (1). Esta carga se ha obtenido a partir de resultados empíricos, pero que han demostrado ofrecer una aproximación para el *CLR* sencilla y de utilidad, permitiendo el dimensionado del sistema real mediante la resolución de un sistema de colas conocido.

En esta Sección se presenta el modelo analítico obtenido para estimar el *CLR* en el dispositivo *IMUX*, a partir de una aproximación a la solución exacta del sistema de colas *M/D/s/s+B* [8]. Ésta



última es especialmente laboriosa si el tamaño del *buffer* *B* es grande.

Partimos de la aproximación del *CLR* para el sistema de colas *M/D/s/s+B* en tiempo continuo, propuesta por F. Barceló y M. Aguilar en [14] desarrollada a partir de [15] y trabajos anteriores desarrollados por F. Barceló [10, 16]. Así, asumiendo que el proceso de llegada de celdas es de *Poisson*, la probabilidad de que una celda sea descartada coincide con la probabilidad de que el sistema se halle congestionado (propiedad *PASTA*, *Poisson Arrivals See Time Averages*), es decir contenga *s+B* celdas. Para la cola en tiempo continuo *M/D/s/s+B* esto es [14, 15]:

$$CLR_{M/D/s/s+B}^{cont}(\rho, s, B) = P_{s+B}(M/D/s/s+B) \cong \frac{PD}{1 - \rho \left(\rho_{eq} \right)^B PD} (1 - \rho) \left(\rho_{eq} \right)^B \quad (2)$$

donde *PD* es la probabilidad de demora en la cola continua *M/M/s* dada por la fórmula de Erlang-C ($PD=C(s, \rho)$) y ρ_{eq} es la carga equivalente para la cola *M/M/s* que proporciona la misma longitud media de la cola que en la *M/D/s*. Su expresión es sencilla y puede consultarse en [14, 16].

Para obtener la aproximación del *CLR* en la cola discreta *M/D/s/s+B*, hay que añadir a esta aproximación del *CLR* para cola continua *M/D/s/s+B*, otra cantidad que denominamos CLR_{extra} [14]. Esto es debido a que en una cola con tiempo de servicio discreto, el tiempo se divide en intervalos de longitud fija o *slots*. Es decir, los servicios sólo ocurren al principio de cada *slot* (cuando como mucho las *s* primeras celdas del *buffer* empiezan a ser servidas) y las posibles salidas del sistema ocurren al final del *slot* temporal. Las celdas que llegan al sistema durante un *slot* temporal, ingresan en el *buffer*, si es que hay capacidad suficiente. Esto no se produce en una cola con tiempo de servicio continuo, en que las celdas son servidas en cuanto llegan al sistema (si es que hay algún servidor libre). Esta acumulación de celdas durante el *slot* temporal, produce la existencia de un *CLR* adicional con respecto al *CLR* de una cola con tiempo de servicio continuo. Una aproximación para el CLR_{extra} se encuentra extensamente desarrollada en [14]. Su expresión viene dada por la Ecuación (3):

$$CLR_{M/D/s/s+B}^{extra}(\rho, s, B, \bar{Q}_D) \cong 1 - e^{-\lambda} \sum_{k=0}^{\lceil B - \bar{Q}_D + \lambda/2 - 1 \rceil} \frac{\lambda^k}{k!} \cong e^{-\lambda} \lambda^L \left(\frac{L + \lambda + 1}{(L + 1)!} \right) \quad (3)$$

donde $\bar{Q}_D(\rho, s)$ es la longitud media del *buffer* en la cola *M/D/s*, computada mediante aproximación en [14, 16] a partir de la longitud media del *buffer* en la cola *M/M/s*; y $L = \lceil B - \bar{Q}_D + \lambda/2 \rceil$, siendo $\lceil i \rceil$ el menor entero superior o igual a *i*. Para entender la expresión de la Ecuación (3), de nuevo es necesario remitirse a [14]. Sin embargo, sí presentaremos las bases de su desarrollo para entender su procedencia:

Sea de *Poisson* el proceso según el cual llegan las celdas al sistema. La probabilidad de que lleguen k celdas al sistema durante un intervalo temporal igual al tiempo de servicio es:

$$\pi_k = \frac{\lambda^k e^{-\lambda}}{k!} \quad (4)$$

donde λ es la tasa de llegada de celdas al sistema. Asumimos que en el *buffer* en media hay $\bar{Q}_D - \lambda/2$ celdas al inicio del intervalo temporal, donde $\lambda/2$ es el número medio de celdas aproximado que han llegado al sistema durante el intervalo temporal de un servicio. Así, el espacio de *buffer* disponible para las celdas que ingresen al sistema durante el tiempo de servicio es $B - (\bar{Q}_D - \lambda/2)$. Sea \bar{A} el número medio de celdas admitidas en el *buffer* durante un intervalo temporal; es decir, se puede entender como la tasa de admisión de celdas en el *buffer* durante un intervalo temporal equivalente al tiempo de servicio. Entonces,

$$\bar{A} = \sum_{k=1}^{B - \bar{Q}_D + \lambda/2} k \pi_k \quad (5)$$

A partir de aquí, se realiza la siguiente aproximación para el CLR_{extra} resultante al considerar que la cola tiene tiempo de servicio discreto:

$$CLR_{extra}(\rho, s, B, \bar{Q}_D) \cong \frac{\lambda - \bar{A}}{\lambda} = 1 - \frac{\bar{A}}{\lambda} \quad (6)$$

Desarrollando las Ecuaciones (5) y (6) se llega a la expresión final en la Ecuación (3).

Finalmente, la aproximación propuesta en [14] para el CLR en una cola discreta $M/D/s/s+B$ se obtiene sumando las Ecuaciones (2) y (3):

$$CLR_{M/D/s/s+B}^{discr}(\rho, s, B) \cong CLR_{M/D/s/s+B}^{cont}(\rho, s, B) + CLR_{M/D/s/s+B}^{extra}(\rho, s, B) \quad (7)$$

Pues bien, se desea caracterizar la carga total ρ_{total} (tal como se plantea en la Ecuación (1)) presente en el dispositivo *IMUX* y poder así utilizar esta expresión para aproximar el CLR en dicho dispositivo:

$$CLR_{IMUX}(\rho, s, B) \cong CLR_{M/D/s/s+B}^{discr}(\rho_{total}, s, B) \quad (8)$$

La obtención de la carga adicional debida a las celdas *Filler* (ρ_{Filler}), se ha llevado a cabo como primera aproximación de manera experimental, observando en el dispositivo *IMUX* simulado su dependencia con la carga neta *ATM* ρ , con el número de enlaces s y con el tamaño del *buffer* B . Los

valores para la carga adicional ρ_{Filler} han sido obtenidos sustituyendo la expresión de la Ecuación (1) para la ρ_{total} en el *IMUX*, en la aproximación de la Ecuación (8) para el $CLR_{M/D/s/s+B}^{discr}(\rho, s, B)$ y comparando dichos valores con el $CLR_{IMUX}(\rho, s, B)$ conseguido mediante el simulador desarrollado en [6]. De esta manera experimental, se ha podido estimar la carga $\rho_{Filler}(\rho, s, B)$. En una tarea próxima pretendemos realizar la justificación de dicha carga mediante análisis, independientemente de los resultados de simulación.

Se dispone de los valores tabulados, facilitando su consulta y acceso para un amplio margen de valores usuales: $0,5 \leq \rho \leq 0,9$, $2 \leq s \leq 8$, $8 \leq B \leq 40$. Como ejemplo se exponen las Tablas 1 y 2, en las que se muestran los valores de ρ_{Filler} en función del número de enlaces s y del tamaño del *buffer* B , para carga neta *ATM* del 70% y 90% respectivamente.

En las Figuras 3 y 4 puede verse la distribución de la carga $\rho_{Filler}(s, B)$ en el *IMUX* debida a las celdas *Filler* para carga neta *ATM* del 60% y 80% respectivamente. Los puntos corresponden a los valores obtenidos del simulador y las curvas a los mismos valores, pero expresados según una función exponencial decreciente conveniente para cada caso. Se ve como realmente el comportamiento encaja muy bien y por lo tanto se podrán utilizar dichas expresiones para simplificar y aproximar la expresión de la carga $\rho_{Filler}(\rho, s, B)$ obtenida del simulador. Hemos caracterizado la zona exponencial decreciente de la carga $\rho_{Filler}(\rho, s, B)$ tal y cómo se ha observado que se comporta dicha carga a partir de cierto valor del tamaño del *buffer*, diferente en cada caso. Para valores inferiores del *buffer*, no es ni interesante ni realista estudiar el comportamiento del *IMUX*, puesto que las pérdidas son demasiado grandes (del orden de 10^{-2}) muy por debajo de los requisitos típicos de Calidad de Servicio propios de los tráficos *ATM*.

Tabla 1. Carga adicional debida a las celdas *Filler*, (ρ_{Filler}) para un *IMUX* con $\rho=0,7$.

$\rho=0,7$							
B	$s=2$	$s=3$	$s=4$	$s=5$	$s=6$	$s=7$	$s=8$
8	1.34E-01	1.48E-01	1.42E-01				
12	9.14E-02	1.43E-01	1.43E-01	1.66E-01	1.82E-01		
16	5.52E-02	9.34E-02	1.11E-01	1.49E-01	1.82E-01	1.84E-01	1.86E-01
20	3.49E-02	6.39E-02	7.69E-02	1.09E-01	1.43E-01	1.75E-01	1.85E-01
24	2.19E-02	4.50E-02	5.51E-02	8.09E-02	1.08E-01	1.35E-01	1.48E-01
28	1.28E-02	3.17E-02	3.99E-02	6.13E-02	8.36E-02	1.07E-01	1.17E-01
32	5.99E-03	2.19E-02	2.86E-02	4.68E-02	6.58E-02	8.56E-02	9.45E-02
36	7.85E-04	1.44E-02	1.98E-02	3.56E-02	5.21E-02	6.93E-02	7.70E-02
40				2.67E-02	4.12E-02	5.64E-02	6.31E-02

Tabla 2. Carga adicional debida a las celdas *Filler*, (ρ_{Filler}) para un *IMUX* con $\rho=0,9$.

$\rho=0,9$							
B	$s=2$	$s=3$	$s=4$	$s=5$	$s=6$	$s=7$	$s=8$
12	2.19E-02	1.24E-02					
16	2.79E-02	3.53E-02	3.26E-02	2.56E-02	2.28E-02	1.74E-02	6.38E-03
20	1.74E-02	3.02E-02	3.52E-02	3.95E-02	4.01E-02	3.47E-02	2.79E-02
24	1.12E-02	2.10E-02	2.55E-02	3.61E-02	4.50E-02	4.54E-02	4.11E-02
28	7.14E-03	1.51E-02	1.87E-02	2.72E-02	3.61E-02	4.52E-02	4.73E-02
32	4.30E-03	1.11E-02	1.40E-02	2.11E-02	2.84E-02	3.60E-02	3.96E-02
36	2.20E-03	7.86E-03	1.05E-02	1.65E-02	2.28E-02	2.93E-02	3.23E-02
40	5.78E-04	5.60E-03	7.82E-03	1.31E-02	1.86E-02	2.42E-02	2.68E-02

Para cada valor de la carga ρ , se han obtenido los valores de $a(s)$ y $b(s)$ en la expresión $\rho_{Filler}(s, B) = a(s) e^{-b(s)B}$ para un número de canales $s=2, 4, 6$ y 8 , realizando interpolación polinómica de los cuatro valores medidos para $a(s)$ en cada caso, y así poder expresar el coeficiente a en función de s . Lo mismo se ha realizado para $b(s)$, aunque en este caso con un polinomio de segundo grado ha sido suficiente. Así pues, para valores de la carga del 60% y del 80% las expresiones correspondientes son:

$$\rho_{Filler}(\rho = 0.6, s, B) = (0.8538 - 0.2268s + 0.05744s^2 - 0.003735s^3) * e^{-(0.2017 - 0.03413s + 0.002082s^2)B} \quad (9)$$

$$\rho_{Filler}(\rho = 0.8, s, B) = (0.3078 - 0.05645s + 0.01461s^2 - 0.000879s^3) * e^{-(0.1506 - 0.02561s + 0.001616s^2)B} \quad (10)$$

Tal y como puede observarse en las Figuras 3 y 4, a medida que aumenta la carga neta ATM (ρ) la carga

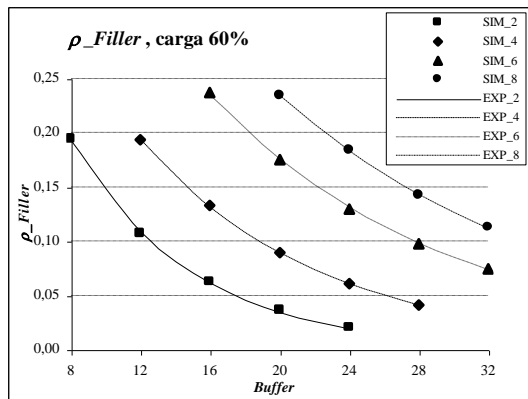


Figura 3. Distribución exponencial de la carga ρ_{Filler} en un IMUX cargado al 60%.

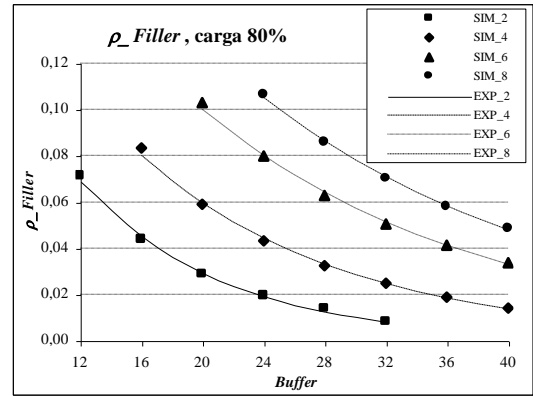


Figura 4. Distribución exponencial de la carga ρ_{Filler} en un IMUX cargado al 80%.

adicional debida a las celdas *Filler* (ρ_{Filler}) disminuye, puesto que para ρ grande hay más celdas ATM disponibles y no será necesario añadir tantas celdas *Filler* para mantener los enlaces ocupados. Por otro lado, ρ_{Filler} aumenta a medida que crece el número de canales de salida del IMUX, s . La tasa IDCR aumenta y por consiguiente la longitud de los intervalos de decisión de inserción de una celda ($1/IDCR$) disminuye (ver Figura 2), por lo que se decide más frecuentemente y por lo tanto con más probabilidad se insertan celdas *Filler* en caso de no encontrar celda ATM disponible. También se advierte que a medida que el tamaño del *buffer* aumenta (para carga muy alta sucede a partir de cierto tamaño de *buffer*) ρ_{Filler} disminuye, con tendencia exponencial. Ello es así, ya que para tamaños grandes del *buffer*, se pierden menos celdas ATM, por lo que habrá más celdas ATM disponibles en la memoria y esto hará que sea necesario introducir menos celdas *Filler*. Este efecto se ve incrementado a medida que ρ aumenta, y por lo tanto aumenta la probabilidad de generación de celdas ATM, con lo que en ese caso el *buffer* estará más lleno de celdas ATM.

5. Resultados Numéricos.

En las Figuras 5 y 6 se puede observar el CLR obtenido por simulación y analíticamente de un IMUX cargado al 60% y al 80% respectivamente, en función del tamaño del *buffer*, para un sistema IMA formado por 2, 4, 6 y 8 enlaces. Los símbolos indican los resultados obtenidos mediante simulación, mientras que las líneas representan los resultados analíticos calculados mediante las Ecuaciones (1), (7), (9) y (10). De la observación de ambas Figuras, puede concluirse que la aproximación propuesta ofrece unos valores para el CLR muy ajustados a los obtenidos mediante simulación. Lo mismo ocurre para el resto de valores de la carga, ρ . En la Figura 5 hemos representado los valores del CLR obtenidos analíticamente y mediante simulación, para un IMUX con carga por enlace igual al 60% y capacidad del buffer de 8 a 28

celdas. La Figura 6 representa el *CLR* para un *IMUX* con carga de nivel *ATM* igual al 80% y tamaño de buffer de 8 a 40 celdas. El *CLR* mínimo que se alcanza es del orden de magnitud de 10^{-6} , ya que el número de celdas procesadas en el simulador no es superior a 10 millones debido a limitaciones en tiempo de ejecución.

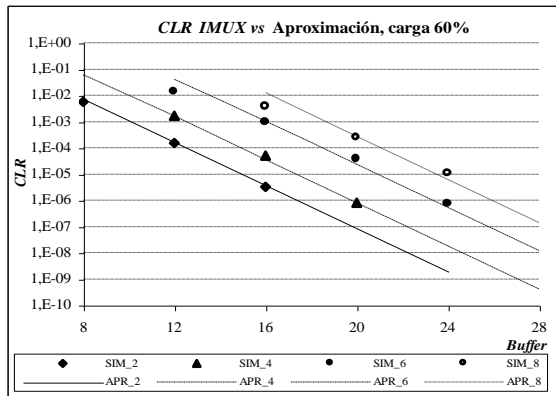
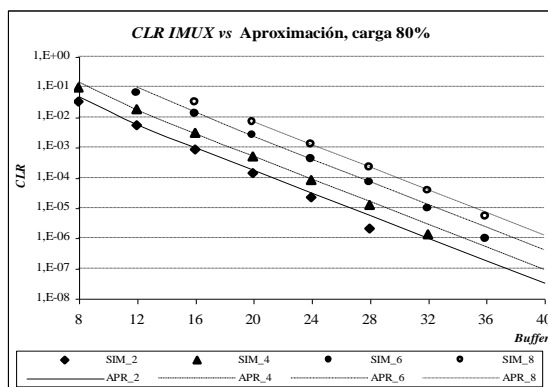


Figura 5. *CLR* en un *IMUX* cargado al 60%.

Figura 6. *CLR* en un *IMUX* cargado al 80%.

desprende que, en efecto, la probabilidad de pérdida depende de la carga del multiplexor *IMUX*, del

número de enlaces y del tamaño del *buffer*. El *CLR* disminuye a medida que aumenta el tamaño del *buffer*, pero lo hace más lentamente a medida que la carga es mayor y consecuentemente la pendiente es más suave. También se observa que la diferencia entre tener un número de enlaces u otro, es menor a medida que la carga se hace muy grande [6]. En resumen, para el diseño de un *IMUX* se podría utilizar la metodología propuesta para dimensionar sus recursos. Como ejemplo, podríamos considerar un *IMUX* que debe ofrecer un ancho de banda de 6 Mbps y un *CLR* menor o igual a 10^{-5} para una carga ofrecida del 80%. En este caso, y asumiendo llegadas de *Poisson*, sería necesario contratar 4 enlaces *T1/E1* y un *IMUX* con buffer de capacidad 28 celdas, como puede deducirse de la Figura 6. Una



vez se ha elaborado el modelo analítico, ya no es necesario realizar simulaciones costosas en tiempo

de ejecución.

6. Conclusiones y líneas futuras.

En este trabajo se ha expuesto en qué consiste un multiplexor inverso *ATM*, cuál es su funcionamiento básico, las principales aplicaciones de red para las que está ideado y algunos de los problemas potenciales que presenta. Asimismo, se evalúan sus prestaciones y se propone un método sencillo para dimensionar los parámetros más relevantes de dichos dispositivos (número de canales y tamaño del *buffer*) en función de la probabilidad de pérdida de celdas. Con el simulador desarrollado, se ha realizado un estudio exhaustivo del funcionamiento de los sistemas *IMA*, analizando el efecto producido por las celdas de control *ICP*, *SICP* y *Filler* para todos los casos posibles que cubren el rango de valores comerciales. Para la realización del simulador [6], se ha seguido fielmente la especificación publicada por el *ATM Forum* [3]. Para realizar una primera estimación sobre el comportamiento del sistema se ha considerado tráfico de entrada de *Poisson*. Ésta es una hipótesis razonable cuando los dispositivos *IMUX* se ubican en niveles altos de la red, donde el tráfico agregado presenta un alto nivel de mezcla con un coeficiente de variación bajo no mucho mayor que la unidad, como sucede cuando el proceso no es muy disperso [11, 17]. En la actualidad, se está trabajando en comprobar la bondad de la aproximación efectuada y del método de trabajo con otros tipos de tráfico, como por ejemplo tráfico a ráfagas y modelos sencillos de tráfico residencial y de entornos de negocios, que presentan ciertas características de autosimilitud, con funciones de distribución dispersas.

Asimismo, se sigue estudiando la aplicación de varios modelos analíticos basados en la resolución exacta o aproximada del sistema de ecuaciones obtenido de plantear la cadena de Markov correspondiente al sistema *IMUX*, que permitirá obtener fórmulas cerradas sencillas que calculan el *CLR* y el tiempo medio de espera en el dispositivo, en función del tamaño del *buffer*, número de canales y carga ofrecida. También, se están considerando diversos niveles de prioridad y políticas de gestión del *buffer* para dar soporte a diferentes clases de tráfico *ATM*. Las expresiones obtenidas podrán compararse con los resultados entregados por el simulador, validando el modelo. El objetivo final es desarrollar una metodología para dimensionar los recursos de un *IMUX* bajo diferentes condiciones utilizando fórmulas sencillas y evitando así realizar costosas simulaciones. Finalmente, también se utilizarán para dimensionar una red completa *ATM* compuesta por sistemas *IMA*.

Agradecimientos

Este trabajo ha sido financiado por el proyecto de investigación SIGLA (CICYT TEL96-1452).

Referencias

- [1] Kathleen Cholenka, "ATM Inverse Muxing: Budgeted Bandwidth", *Data Communications*, September 1997.
- [2] Robin D. Langdon, "ATM Inverse Muxing for Easier Acces", *Data Communications*, June 1997.
- [3] The ATM Forum Technical Committee, "Inverse Multiplexing for ATM (IMA) Specification Version 1.0", Physical Layer Working Group, July 1997.
- [4] James W. Conard, Editor, "Handbook of Communications Systems Management", Fourth Edition, Auerbach Publications, 1998.
- [5] Passport. "Inverse Multiplexing for ATM". User Guide. Nortel Magellan. Northern Telecom, November 1997.
- [6] Marcos Postigo Boix, "Multiplexores Inversos ATM. Simulación y Análisis de Prestaciones" Proyecto Final de Carrera. ETSETB. UPC. Director: Juan García Haro. 1998.
- [7] M. Aguilar, M. Postigo, J. García, "Multiplexores Inversos ATM: Operación, Aplicaciones y Aspectos de Planificación y Evaluación de Prestaciones", *VIII Jornadas de I+D en Telecomunicaciones*, TELECOM I+D'98, pp. 249-258, 28-29 Octubre 1998, Barcelona.
- [8] W. W. Chu, "Buffer Behaviour for Poisson Arrivals and Multiple Synchronous Constant Outputs". *IEEE Trans. On Computers.*, Vol. C-19, No 6, June 1970.
- [9] M. Aguilar, M. Postigo, J. García, "Estudio sobre parámetros de calidad en Multiplexores Inversos ATM", URSI'98, pp. 647-648, 16-18 Septiembre 1998, Pamplona.
- [10] F. Barceló, J. Paradells, M. Aguilar, "Mean Waiting Time in the M/H₂/s Queue: Application to Mobile Communications Systems", 15th IMACS World Congress, Berlin, 24-29 August 1997.
- [11] M. Aguilar, F. Barceló, J. García, "Estimación de los parámetros de calidad para distintos tráficos en nodos MTA con enlaces múltiples", *Jornadas de Ingeniería Telemática, JITEL'97*, pp. 233-240, 15-17 Septiembre 1997, Bilbao.
- [12] M. Aguilar, J. García, M. Postigo, "Inverse Multiplexing for ATM. Operation, Applications and Performance Evaluation for the Cell Loss Ratio", ICATM'99, 21-23 Junio 1999, Colmar, Francia. (Publicación Aceptada).
- [13] J. M. Pitts, J. A. Schormans, "Introduction to ATM Design and Performance", John Wiley & Sons, February 1997.
- [14] M. Aguilar, F. Barceló, P. Iserte, "Approximation to the CLR in ATM switches with multiserver output discrete queus and priority", *IFIP Workshop TC6, ATM'98*, pp. 63/1-63/10, 20-22 Julio 1998, Ilkley, U.K..
- [15] T. Kimura, "Approximating the Queue-Length Distribution in the Finite Capacity M/G/s Queue²". *Discussion Paper Series A*, No. 18, Faculty of Economics, Hokkaido University, Junio 1993.
- [16] F. Barceló, M. Aguilar, "Approximation to the M/D/s queue with finite buffer: Application to the CLR in ATM nodes", *International Teletraffic Seminar (ITC)*, St.Petersburg, pp. 401-409, Junio 1998.
- [17] Xuedao Gu, Kazem Sohraby, Dhadesugoor R. Vaman, "Control and Performance in Packet, Circuit and ATM Networks", Kluwer Academic Publishers, 1995.

Reserva eficiente de recursos para canales de reserva en redes en tiempo real.

Enrique Hernández Orallo y Joan Vila Carbó

ehorallo@bancaja.es, jvila@disca.upv.es

Departamento de Informática de Sistemas y Computadores (DISCA)
Universidad Politécnica de Valencia

Abstract

The goal of real-time communication services is to provide a QoS (Quality of Service) in order to guarantee some end-to-end performance parameters. This is usually accomplished using some protocol for resource reservation. In addition, there is an increasing number of critical multimedia applications that require uninterrupted service (high availability) in the presence of link or router failures. A technique proposed for coping with these failures is backup channels, but it comes at the cost of considerably increasing resource reservations. Since failures are not very frequent, some degree of overbooking can be considered when placing the reservations but, even in this case, an important problem remains unsolved: provided that failure detection latency is non-negligible, the delay requirements for the backup channel would have to be very demanding, and the corresponding resource reservations exorbitant. This paper addresses this problem and proposes an efficient scheme to place the reservation for both, the primary and backup channel. The paper evaluates the performance of this technique under three different workloads on the RSVP (ReSerVation Protocol) and how to fit these workloads into the model.

1. Introducción

Debido al aumento de las prestaciones de las redes de comunicaciones han aparecido nuevas aplicaciones multimedia basadas en red como videoconferencia, visualización científica y telemonitorización médica, etc. La calidad de servicio (QoS : *Quality of Service*) se establece entre el cliente y el proveedor del servicio para garantizar determinados parámetros como rendimiento, retraso máximo o fiabilidad.

Para asegurar retrasos en red acotados se requiere por una parte, la adopción de esquemas de planificación de paquetes basados en prioridades y por otra, un control de admisión de peticiones, basado en un protocolo de reserva de recursos. En Internet, la planificación por prioridades ya ha sido introducida en el protocolo IPv6, mientras que la reserva de recursos es el objetivo de RSVP (*Reservation Protocol*) [4,6]. Otras propuestas sobre canales en tiempo real es el *Tenet Suite* [16,17]. En general, los recursos de la red (CPU, ancho de banda, memoria, etc.) son reservados para cada sesión a lo largo de una ruta, y los mensajes se envían siempre por esta ruta. Esta especie de circuito virtual se denomina canal en tiempo real. El control de admisión chequea la disponibilidad de los recursos necesarios para un canal; y en el caso de que no se disponga de ellos se deniega su creación.

El esquema de reserva de recursos y control de admisión ha tenido éxito al proporcionar tiempos de transmisión garantizados, pero es difícil evitar los errores debidos a fallos en los nodos. Esto es debido a que los mensajes sólo pueden ir por la ruta

reservada a priori, y no pueden sortear los elementos con fallos de forma dinámica. Esto se puede evitar utilizando canales redundantes con dos esquemas opuestos : en la copia múltiple [3] todos los canales redundantes están activos simultáneamente, mientras que al establecer un canal de reserva (*backup channel*) [18] sólo se activa cuando se detecta un fallo. Este segundo esquema parece más atractivo, ya que no gasta recursos en la ausencia de fallo, pero para que sea efectivo el canal secundario se tiene que establecer de forma rápida y segura. Esto implica que también se tengan que reservar los recursos en la red aunque no se usen. Para reducir esta reserva de recursos es muy interesante el esquema introducido por *Shin* [2] basado en reservar los recursos sobrantes (*spare resources*). Sin embargo, existe un problema por resolver: los requerimientos de retraso en el canal de reserva tienden a ser muy exigentes, debido a la latencia en la detección del fallo y el tiempo de establecimiento del canal.

En este artículo se aporta un nuevo esquema para establecer canales de reserva que implica un menor uso de recursos en la red. Esta propuesta se basa en activar el canal de reserva cuando se “sospecha” un fallo, en vez de esperar a que realmente ocurra. Esto permite usar un canal de reserva con unos requerimientos menores. La demanda de recursos de esta técnica se compara con los esquemas anteriores sobre el protocolo de reserva RSVP y tres modelos de carga: HVBR (tasa de transmisión muy variable) , VBR (tasa de transmisión variable) y CBR (tasa de transmisión constante).

Para realizar el control de admisión se necesita caracterizar el tráfico que va a circular por el canal. Cada protocolo define su modelo de tráfico, y en el caso concreto de RSVP se ha definido un modelo para un servicio garantizado [7]. Esto conduce a un interesante problema: cómo obtener a partir de un tipo de tráfico los parámetros que definen su modelo en cada protocolo. Para ello se puede usar la envolvente empírica [12,14], que proporciona un límite del tráfico independiente del tiempo.

El resto del artículo está organizado de la siguiente manera: En la segunda sección se describen los diferentes mecanismos que se han diseñado para proporcionar tolerancia a fallos en redes en tiempo real y se introduce el nuevo esquema para minimizar el uso de los recursos de la red. En la tercera sección se da una breve introducción al protocolo RSVP. En la sección cuarta se describe un modelo de tráfico determinista basado en la envolvente empírica y se introducen tres tipos de tráfico que son usados en la evaluación de la reducción de los recursos en RSVP descrita en el quinto punto. Por último se establecen las conclusiones del estudio realizado.

2. Esquemas de detección de fallo para canales de reserva

En una red en la que se dispone de varias rutas entre los nodos emisor y receptor (Fig. 1) existen varios esquemas para proporcionar tolerancia a fallos en canales de tiempo real [2,3]:

- Establecimiento de **varios canales simultáneos**, y envío de paquetes duplicados para que al receptor le llegue al menos uno. Esta es una técnica costosa en recursos pero puede soportar fallos sin que se corte el servicio.
- Establecimiento de un **canal primario**, y en el caso de que falle, se restablecería la comunicación por otro canal secundario. El inconveniente es el tiempo de restablecimiento del canal secundario y la posibilidad de que no existan recursos disponibles.

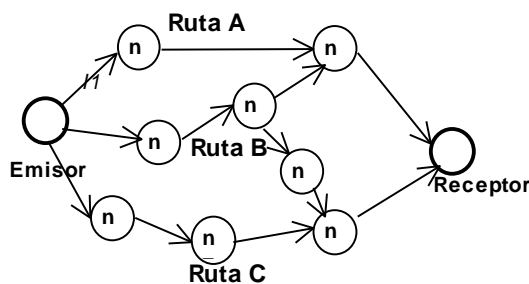


Figura 1 : Red con varias rutas

Para evitar los inconvenientes del segundo esquema y que el establecimiento del canal secundario sea rápido se puede recurrir a reservar los recursos del canal secundario con anterioridad. Este último esquema requiere que se detecte el fallo del canal primario para establecer la comunicación por un canal secundario. Las propuestas existentes para la detección del fallo [2,18] normalmente se basan en el envío de paquetes del tipo “estoy vivo” (*heartbeat*) entre los nodos o/y emisor y receptor de un canal de tiempo real. Un problema importante comentado en [19] es la latencia de detección de fallo, especialmente en la técnica de detección emisor-receptor. Esta latencia puede ser un serio obstáculo en aplicar ésta técnica, ya que provoca que los requerimientos de retraso en los canales de reserva son muy exigentes para mantener un servicio ininterrumpido, y esto causa que las correspondientes reservas sean a veces inaceptables.

Este artículo introduce un esquema de detección de fallos que permite una reserva de recursos más eficiente. El esquema propuesto se basa en “sospechar” un fallo cuando en el canal primario se produce un retraso próximo al retraso máximo, aunque inferior a éste. Siempre que se sospecha de este fallo, el canal de reserva se activa. Esto da lugar, en ocasiones, a la detección de “falsos fallos” que provocan la activación innecesaria del canal de reserva, pero se tiene como contrapartida importante que la reserva que debe realizarse para el canal de reserva es significativamente menor que si se activara sólo ante “auténticos fallos”. Así pues, se trata de sacrificar la exactitud en la detección en aras de una mayor eficiencia en la reserva. La única situación donde la inexactitud en la detección del fallo no se puede considerar aceptable sería con un tráfico a ráfagas, donde el emisor tiene periodos donde no transmite. Esto implicaría, o bien inyectar paquetes del tipo “estoy vivo” o modificar el tráfico para asegurar que el canal sea usado en estos periodos de no transmisión.

Sea d_f el tiempo de detección de fallo. Se asume que d_f es estrictamente menor que el máximo retraso en el canal primario d_A . El problema consiste en como escoger d_f para minimizar la reserva de recursos. Este artículo ofrece algunas evaluaciones experimentales para diferentes tipos de tráfico.

El retraso máximo experimentado por el primer paquete retransmitido por el canal de reserva cuando se produce un fallo en el canal primario puede ser expresado como:

$$d_{total} = d_A + d_e + d_B \quad (1)$$

donde d_A y d_B son los retrasos máximos en los canales primario y de reserva respectivamente y d_e el tiempo de establecimiento del canal de reserva desde que se descubre el fallo. Esto implica que para un

retraso total d_{total} hay que establecer canales con unos retrasos bastante menores, lo que supone una reserva de recursos en la red mucho mayor. (A menor retraso mayor consumo de recursos en la red).

De acuerdo a la técnica propuesta, la idea es escoger una límite en la detección de fallo d_f tal que $d_f < d_A$ para permitir un valor de d_B mayor. Cuando el retraso de un paquete excede d_f se empieza a restablecer el canal de reserva, pero sin descartar el primario ya que podría ser un “falso fallo”. Así el tiempo de retraso máximo se calcularía de la siguiente forma:

$$d_{total} = \max(d_A, d_f + d_e + d_B) \quad (2)$$

De esta forma d_A podría ser d_{total} y el valor de d_B será $d_{total} - d_f - d_e$. El hecho de que d_f sea siempre menor que d_A implica una importante reducción de la reserva de recursos en el canal de reserva. Además los recursos asignados al canal de reserva mientras no se usen los podría utilizar el *router* para la transmisión de otro tipo paquetes no prioritarios, con lo realmente no se están desperdiciando recursos. El único momento donde se desperdician recursos es cuando se establece el canal B cuando se detecta un fallo. Esto plantea los siguientes temas:

- Cómo calcular el valor de d_f . Si este valor es alto, el valor d_B tendrá que ser bajo con lo que habrá un reserva elevada de recursos, y si el valor d_f es bajo provocará que se establezca el canal de reserva con demasiado frecuencia con el consiguiente derroche de recursos ante falsos fallos.
- Cómo calcular el coste de establecer el canal de reserva.

3. RSVP

RSVP se ha diseñado [4,5,6] para permitir a los emisores, receptores y *routers* establecer una ruta que pueda soportar la calidad de servicio requerida. RSVP no es un protocolo de encaminamiento, se usa meramente para reservar recursos a través de la ruta que se establezca por cualquiera de los protocolos de niveles inferiores.

IETF (*Internet Engineering Task Force*) ha considerado varias clases de calidad de servicio para RSVP aunque sólo dos han sido formalmente especificados para RSVP: Servicio Garantizado (*Guaranteed Service*) [7] y Servicio de carga controlada (*Controlled-Load Service*) [8]. En este trabajo se va a limitar el estudio al servicio garantizado.

3.1. Servicio garantizado

Esta calidad de servicio está destinada para aplicaciones con requerimientos exigentes de tiempo

real. Esta calidad asegura un ancho de banda, un límite en el retraso y ninguna pérdida en las colas.

Un flujo es descrito usando un esquema de cubo con goteo (*token bucket*) [20] y dada esta descripción, cualquier elemento de la red (un *router*, una subred, etc.) calcula varios parámetros describiendo como va a manejar los datos del flujo como se describe en el siguiente apartado. Combinando los parámetros de los distintos elementos que recorre el flujo es posible calcular el retraso máximo que se producirá en el flujo.

3.2. Retraso de red

Cada *router* caracteriza el servicio garantizado para un flujo determinado, asignando un ancho de banda R , y un espacio de memoria (*buffer space*) B , que representa los recursos que el flujo puede consumir. Esto representa que existe un ancho de banda R entre emisor y receptor. Así para un flujo que siga las características de cubo con goteo de capacidad b y tasa r se puede calcular el límite del retraso como b/R siempre que R sea mayor r . Para permitir desviaciones de este modelo perfecto se añaden dos términos de error C y D ; con lo que el límite del retraso se convierte en $b/R + C/R + D$. El término C es el error dependiente de la tasa. Representa el retraso que un datagrama en el flujo puede experimentar debido a la tasa de los parámetros en el flujo. El término de error D es el error independiente de la tasa y representa el peor caso de variación de tiempo de tránsito a través del *router*.

Un límite más preciso para el retraso de cola en los *routers* en la red se puede obtener especificando una tasa pico del flujo p , y un tamaño máximo del paquete M . Esta tasa pico provoca una reducción del retraso, mientras que el tamaño máximo del paquete tiene en cuenta los efectos de la partición en paquetes del flujo. De esta forma el retraso en cola se calcula de la siguiente forma:

$$Q_{delay} = \frac{(b-M)(p-R)}{R(p-r)} + \frac{(M+C_{tot})}{R} + D_{tot} \quad (\text{caso } p > R \geq r) \quad (3)$$

$$Q_{delay} = \frac{(M+C_{tot})}{R} + D_{tot} \quad (\text{caso } R \geq p \geq r) \quad (4)$$

donde C_{tot} y D_{tot} representan el sumatorio de los términos de error C y D de cada *router* de la ruta.

Cada *router* necesita ser informado de las características del tráfico, T_{spec} , y del flujo con las características de las reservas realizadas, R_{spec} . Además necesita los términos C_{tot} y D_{tot} que representan la suma de los términos de error C y D , de cada *router* desde el origen del mensaje. Las estructuras T_{spec} y R_{spec} están descritos en la tabla 1.

Tspec	Descripción	Unidad
p	tasa pico del flujo	Bytes/s
b	Tamaño del cubo	Bytes.
r	Tasa de transmisión del cubo	Bytes/s
m	Tamaño mínimo de un paquete	Bytes
M	Tamaño máximo de paquete	Bytes
Rspec		
R	Ancho de banda a reservar	Bytes/s
S	Termino slack	μs

Tabla 1 : Estructura Tspec y Rspec.

Para utilizar este esquema los planificadores de los routers tienen que aproximarse al modelo de flujo. Existen varios algoritmos de planificación como el WFQ (*Weighted Fair Queuing*) [9], *Jitter-EDD* [10] y *Virtual Clock* [11], que se aproximan a este modelo.

3.3. Caracterización de los elementos de red

El modelo a utilizar en los nodos dependerá del algoritmo de planificación de los paquetes. Un nodo vendrá definido por un ancho de banda de salida L_i , y una capacidad de memoria B_i .

Hay que calcular los valores C_i y D_i para cada elemento de la red. Para una planificación en los nodos WFQ (*Weighted Fair Queuing*) los valores C_i y D_i se calculan así ([8]): D_i es igual al valor del MTU (unidad mínima de transmisión) del enlace dividido por el ancho de banda del enlace. Así se tiene en cuenta la posibilidad de que un paquete llegue justo cuando se empieza a transmitir el paquete más grande. El valor C_i se asume que es M para tener en cuenta los efectos de la división en paquetes

3.4. Test en el receptor

El objetivo del test en el receptor es calcular el valor R a reservar en función del retraso exigido. Para establecer un canal el emisor envía un mensaje *Path* con destino al receptor en el que se contiene la descripción del tráfico (estructura *Tspec*) y especificaciones de la red (estructura *Adspec*). Cuando este mensaje pasa por cada *router* i de la ruta va obteniendo los parámetros de cada nodo (valores C_i y D_i , ancho de banda, latencia), y actualiza la estructura *Adspec*. Los valores C_{tot} y D_{tot} se calculan por tanto de la siguiente forma:

$$D_{tot} = \sum D_i = \sum \frac{MTU_i}{AB_i}$$

$$C_{tot} = \sum C_i = \sum M$$

Cuando el mensaje llega al receptor este obtiene los siguientes parámetros del *Tspec*: r , b , p , m .

Además también extrae de *Adspec* los siguientes parámetros : latencia mínima de la ruta, C_{tot} , D_{tot} , MTU y ancho de banda de la ruta.

El límite requerido para el retraso de cola, Q_{delreq} se calcula restando la latencia mínima de la ruta, del valor del retraso de emisor a receptor requerido por la aplicación receptora. Con esto se puede calcular el valor mínimo de reserva R , aplicando en la ecuación 3 o 4 los valores determinados de C_{tot} , D_{tot} , r , b , p , M . Si el valor R excede el ancho de banda obtenido del *Adspec* recibido se reducirá. El receptor entonces puede crear una especificación de la reserva, *Rspec*, que contiene el valor R de ancho de banda que se reservará en cada *router*. Este mensaje se envía de vuelta por la ruta que ha recorrido, reservando en este momento los *routers* los recursos.

4. Caracterización del tráfico.

Para poder determinar el máximo retraso de un paquete, es importante tener una descripción determinista del tráfico. Se podría utilizar la tasa pico de transmisión pero esto provocaría una utilización no eficiente de la red. Contra más se acerque el modelo de la carga a la real, más información dispondrá la red para planificar de forma adecuada los paquetes.

Cada protocolo define su modelo de tráfico, en *Tenet Suite* [16] se tiene el modelo (X_{min} , S_{max} , I , X_{ave}) donde la fuente está limitada por los siguiente parámetros: X_{min} tiempo mínimo entre mensajes; S_{max} tamaño máximo de un paquete y para un intervalo de tiempo I el tiempo medio entre paquete es de X_{ave} . En RSVP [6] se sigue el modelo de tráfico (σ, ρ) de cubo con goteo donde para un intervalo de tiempo t , la fuente no puede transmitir mas de $\sigma + \rho t$.

El problema reside en como a partir de un tipo de tráfico obtener los parámetros que definen su modelo en cada protocolo. Este artículo resuelve este problema usando las funciones de tráfico limitado $A^*(t)$ y la envolvente empírica $E(t)$ que proporciona un limite del tráfico independiente del tiempo. Se define la función $A(t)$ como el número de bits transmitidos para un tiempo t . Esta función nos da una visión de los bits transmitidos pero dependiente del tiempo. Sin embargo, para algunos test de admisión es necesario tener un límite del peor caso del tráfico de entrada que sea independiente del tiempo.

Si el tráfico de un canal está determinado por la función A tal que $A[\tau, \tau+t]$ representa la transmisión en un intervalo de tiempo $[\tau, \tau+t]$, la función $A^*(t)$ será un límite superior de A para todos los valores de $\tau \geq 0$ si se cumple:

$$A[\tau, \tau+t] \leq A^*(t) \quad (5)$$

Cualquier función $A^*(t)$ que satisfice la propiedad se denomina “función de tráfico limitado”. Estas funciones proporcionan un límite independiente del tiempo de A , por lo que el flujo está acotado para cada intervalo de longitud t . Además, se define la función *envolvente empírica* [12] “empirical envelope” $E(t)$, que es la función de tráfico constante más exacta para una función A :

$$E(t) = \max_{\tau > 0} A[\tau, \tau + t] \quad \forall t > 0 \quad (6)$$

donde se cumple siempre que:

$$A^*(t) \geq E(t) \quad \forall t > 0 \quad (7)$$

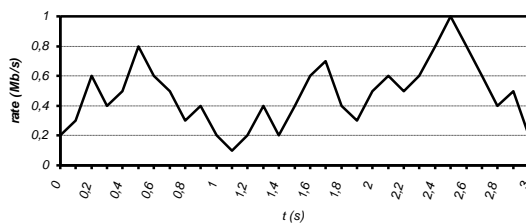
Si la tasa de transmisión de un flujo para un intervalo de 3 segundos está descrita por el gráfico de la Fig. 2(a), la función $A(t)$, la envolvente empírica $E(t)$ y las funciones de tráfico limitado para RSVP y Tenet están representados en la Fig. 2(b).

Como se ve a partir de la envolvente se pueden obtener los parámetros necesarios para describir el flujo en RSVP. La tasa pico p la obtenemos del pico máximo de la descripción del tráfico de la figura 2(a) que es 1 Mb/s . Para RSVP la función de tráfico limitado es $A^R(t) = b + rt$. A partir de la envolvente $E(t)$ se obtiene la recta tangente que más se aproxima y con ella se obtienen los parámetros b y r .

A continuación se describen tres tipos de flujos que agrupan los tipos de tráfico más comunes [13]. Se ha intentado formalizar estos tráficos para poder realizar comparaciones sobre carga ficticias aunque basadas en trazas reales. De hecho la idea parte de los ejemplos descritos en [12,14] que utilizan para calcular las funciones de tráfico limitado. A partir de estos modelos se calcula la envolvente y los parámetros de la descripción de carga para RSVP. Para todos los flujos se va utilizar un valor de intervalo $I = 3s$ y $M = 10000 \text{ bits}$.

4.1. Tráfico VBR (Variable Bit Rate)

Este flujo describe una carga con una tasa de transmisión con pequeñas variaciones, como podría ser una transmisión de videoconferencia. Un ejemplo de este tipo de flujo esta en la Fig. 3(a). A partir de aquí se puede calcular $A(t)$, $E(t)$ y la



función de tráfico limitado para RSVP $A^R(t)$. Fig. 3(b).

Los parámetros que determinan el flujo en RSVP son $b = 0.3 \text{ Mb}$ $r = 0.36 \text{ Mb/s}$ $p = 1 \text{ Mb/s}$.

4.2. Tráfico HVBR (High Variable Bit Rate)

Se va a denominar HVBR al flujo con una gran variación de la tasa de transmisión. Este flujo representa una transmisión de información a ráfagas y poco determinista. Un ejemplo de este flujo podría ser el telecontrol o una transmisión de una conversación (Fig. 3(c)). Se calculan también la envolvente y la función de tráfico limitado en la Fig. 3(d).

Los parámetros que determinan el flujo en RSVP son $b = 0.9 \text{ Mb}$ $r = 0.03 \text{ Mb/s}$ $p = 3 \text{ Mb/s}$.

4.3. Tráfico CBR (Constant Bit Rate)

Este flujo tiene una tasa de transmisión constante o con mínimas variaciones (Fig. 3(e)). En este caso las funciones son equivalentes (Fig. 3(f)).

Los parámetros que determinan el flujo en RSVP son $b = 0 \text{ Mb}$ $r = 0.5 \text{ Mb/s}$ $p = 0.5 \text{ Mb/s}$.

5. Evaluación de la reserva de recursos

El objetivo es comparar el esquema de reserva propuesto con los dos esquemas de tolerancia a fallos descritos. Se evalúa el ahorro de recursos que supone el nuevo esquema con los tres tipos de tráficos definidos en el punto anterior. La evaluación se va a realizar en una red con un emisor, tres nodos y un receptor, descrita en la Fig. 4 en el que se establecen un canal primario A y un canal de reserva B, cuyas características están en la tabla 2. Con estos datos se calculan los valores C_i y D_i :

$$D_{tot} = \sum D_i = \sum \frac{MTU_i}{AB_i} = 0.000617$$

$$C_{tot} = \sum C_i = \sum M = 0.04$$

El valor de M es de 0.01 Mbits aproximadamente 1Kbyte. La latencia mínima de la red no se tiene en cuenta al tener un valor despreciable ($l = 0.0006 \text{ s}$). El tiempo de establecimiento d_e se asume cero.

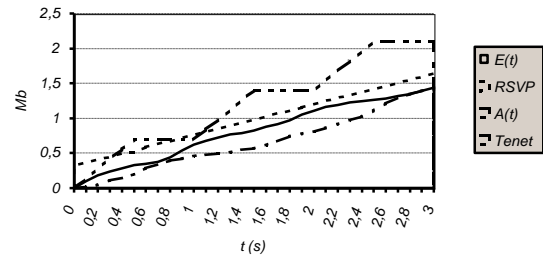


Figura 2 : (a) Ejemplo de tráfico (b) Funciones $A(t)$, $A^*(t)$ y envolvente empírica

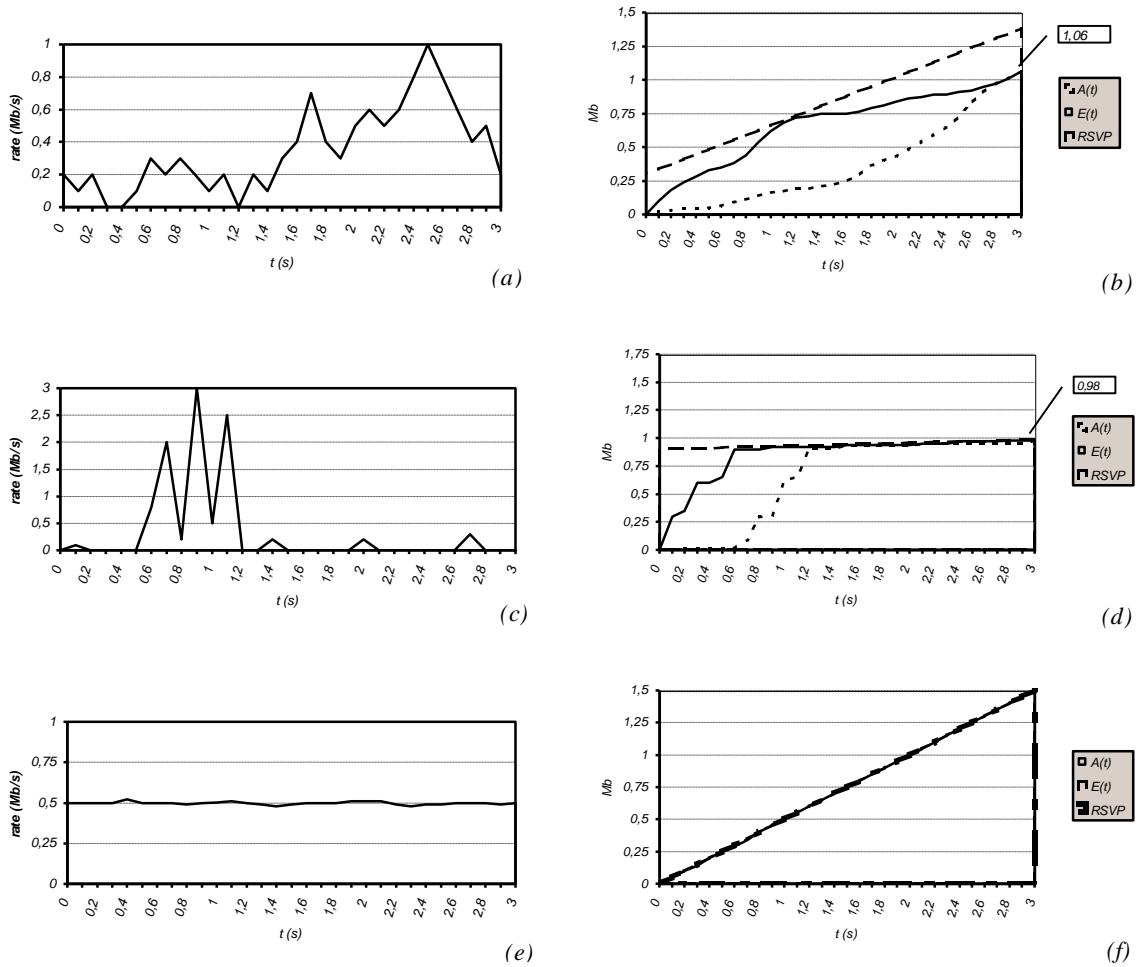


Figura 3 (a) Tráfico VBR, (b) Funciones $A(t), E(t)$ y $A^R(t)$: (c) Tráfico HVBR (d) Funciones $A(t), E(t)$ y $A^R(t)$. (e) Tráfico CBR y (f) Funciones $A(t), E(t)$ y $A^R(t)$.

Se va a comparar los recursos reservados para los siguientes esquemas:

A&B: El emisor envía siempre por los canales A y B. En este esquema el tiempo de retraso máximo de los canales es igual al requerido d_{total} . Este esquema implica que el canal B siempre está usado.

A|B: Este esquema espera a detectar un fallo en el canal A para emitir por el B. La detección del fallo se producirá cuando un paquete se retrase más de un valor $d_{total}/2$. Por lo tanto, para calcular los retrasos en cada canal se dividará d_{total} por 2. Este esquema no utiliza el canal B hasta el momento en que se produce un error. Aunque se tengan reservados los recursos del canal, los routers pueden utilizar el ancho de banda reservado para transmitir tráfico no prioritario, con lo que no se desperdician los recursos asignados.

A^B: El esquema propuesto da a A el retraso d_{total} y a B la mitad de retraso, con lo que se asume que d_f es también la mitad del retraso. Al igual que

el esquema A|B el canal B no se utiliza hasta que el retraso sea mayor que d_f . Pero aquí se pueden producir falsos fallos ya que puede que llegue el paquete esperado por el canal A, con lo que no se hubiera producido el fallo. A partir del tráfico se puede calcular aproximadamente que porción del tráfico provocará que se empiece a restablecer el canal B. Denominaremos a este valor TFF (tasa de falso fallo).

5.1. Tráfico VBR

Sustituyendo los valores del flujo en (3) y (4) nos queda:

$$Q_1 = 0.5/R - 0.45 \quad \text{para} \quad 1 > R > 0.36$$

$$Q_2 = 0.05/R + 0.0006 \quad \text{para} \quad R > 1$$

La tabla 3 describe las reservas necesarias para los tres esquemas. A partir de d_{total} se calculan los retrasos para los canales A y B (d_A y d_B) y la reserva necesaria en cada canal.

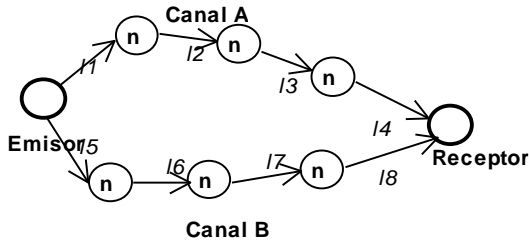


Figura 4 : Red de ejemplo

Enlace	Ancho de banda Mbits)	MTU
L1,15	10	0.002
L2,16	15	0.001
L3,17	20	0.003
L4,18	10	0.002

Tabla 2 : Parámetros de los enlaces

A partir del tráfico se puede calcular la tasa de falsos fallos (TFF) de forma aproximada por el hecho de que WFQ retrasa más los paquetes de mayor tamaño. Cogiendo el ejemplo para $d_{total} = 0.02$ habrá que ver que porción del tráfico pasaría los límites de 0.01 s.

$$Q_2 = 0.01 = 0.05 * r / 1 + 0.0006 \text{ para } R > 1$$

despejando r (tasa) nos da 0.188 Mb/s, que se sobrepasa en un 70% del tráfico.

5.2. Tráfico HVBR

Sustituyendo los valores del flujo en (3) y (4) nos queda:

$$Q_1 = 0.94/R - 0.3 \text{ para } 3 > R > 0.03$$

$$Q_2 = 0.05/R + 0.0006 \text{ para } R > 3$$

En la tabla 4 se dan los resultados para este tráfico. Para calcular el % del tráfico que sobrepasaría el tiempo $d_{total} = 0.02$ se hace el mismo cálculo que en el punto anterior:

d_{total}	d_A	d_B	R_A	R_B	R_{total}	
	A&B					
0.1	0.1	0.1	0.91	0.91	1.82	
0.05	0.05	0.05	1	1	2	
0.02	0.02	0.02	3	3	6	
	A B					
0.1	0.05	0.05	1	1	2	
0.05	0.025	0.025	2.05	2.05	4.1	
0.02	0.01	0.01	5.32	5.32	10.6	
	A^B					TFF
0.1	0.1	0.05	0.91	1	1.91	3%
0.05	0.05	0.025	1	2.05	3.05	30%
0.02	0.02	0.01	3	5.32	8.32	70%

$$Q_1 = 0.01 = 0.05r/3 + 0.0006 \text{ para } R > 1$$

despejando r (tasa) nos da 0.564 Mb/s, que se sobrepasa en un 20% del tráfico.

5.3. Tráfico CBR

Como $r = p$ se tienen las mismas consideraciones que para un nodo con lo que el retraso vendrá dada por (2) que es:

$$Q_2 = 0.05/R + 0.0006 \text{ para } R > 0.5$$

En este caso los valores son equivalentes a la tabla 3 aunque dado que hay mínimas variaciones en la tasa de transmisión provocaría que el canal B estuviera siempre establecido para el esquema A^B con lo que es totalmente ineficiente.

5.4. Comentarios sobre los resultados

El esquema propuesto A^B reserva en total una cantidad de recursos a medio camino entre los otros dos esquemas. Cuanto mayor es el tiempo de retraso máximo total pedido menor es el porcentaje de fallos falsos, con lo que el esquema es más eficiente. También hay que constatar que en estos casos el ahorro de recursos también es menor, debido al propio mecanismo de reserva utilizado. Así en el tráfico VBR para un d_{total} de 0.05s se ahorra casi 1 Mb/s de ancho de banda pero con un tasa de error del 30%. Para 0.1s la tasa de error disminuye al 3% pero el ahorro es de sólo de 0.1 Mb/s.

Para flujos HVBR los resultados son mejores ya que para un retraso de 0.05s se ahorra 0.32 Mb/s para una tasa de error del 13%. Incluso para un retraso más exigente de 0.02s con el esquema A|B el ahorro es de 2.34 Mb/s con una tasa de error un poco superior (20%).

d_{total}	d_A	d_B	R_A	R_B	R_{total}	
	A&B					
0.1	0.1	0.1	2.35	2.35	4.7	
0.05	0.05	0.05	2.68	2.68	5.36	
0.02	0.02	0.02	2.94	2.94	5.88	
	A B					
0.1	0.05	0.05	2.68	2.68	5.36	
0.05	0.025	0.025	2.9	2.9	5.8	
0.02	0.01	0.01	5.32	5.32	10.6	
	A^B					TFF
0.1	0.1	0.05	2.35	2.7	5.05	3%
0.05	0.05	0.025	2.68	2.9	5.58	13%
0.02	0.02	0.01	2.94	5.32	8.26	20%

Tabla 3 y 4 : Reserva para tráfico VBR y HVBR.

Si solo contamos las reservas efectuada sobre el canal A asumiendo que los recursos del canal B los utilizarán los *routers* cuando no esté establecido el canal, los resultado mejoran considerablemente. Así en el tráfico VBR para $d_{total} = 0.05s$ se ahorra 1.05 Mb sobre A|B y 1Mb sobre el A&B. Se podría incluir el coste que supone los falsos fallos en el uso del canal B, que sería $30/100 * 2.05 = 0.68$, con lo que la reserva total sería 1.68, menor todavía a los otros dos esquemas.

6. Conclusiones

Se ha establecido una comparación entre los dos esquemas propuestos para canales de reserva y el nuevo esquema propuesto. De los resultados se concluye:

- El esquema **A&B** es el que menos recursos de la red reserva, pero al coste de estar siempre en uso.
- El esquema **A|B** es el que más recursos reserva, aunque el canal de reserva sólo se activará en el caso de que realmente se detecte un error.
- El esquema propuesto **A^B** supone un compromiso entre las dos soluciones anteriores. Reserva menos recursos que el A|B pero dependiendo del tipo tráfico puede restablecer el canal de reserva sin que se produzca un error. Además reserva menos recursos en el canal primario.

En los resultados de las pruebas realizados se ve claramente que ante un tipo de tráfico HVBR o ante retrasos no muy bajos se provocaría muy pocos restablecimientos del canal de reserva con lo cual sería bastante eficiente. Para el caso de tráfico de tipo CBR o retraso exigentes no resultaría eficiente.

Para intentar resolver este problema se está trabajando en un modelo de descripción del tráfico que provoque un mínimo de restablecimientos de los canales de reserva, añadiendo dos parámetros: la porción del tráfico T que supera una tasa de transmisión r_b . Por ejemplo se podría indicar que sólo el 5% del tráfico supera una tasa $r_b=2Mb/s$. A partir de estos parámetros se podría ajustar los valores d_f y d_b para que la reserva de recursos sea a la vez eficiente y la tasa de falsos fallos sea mínima.

Referencias

- [1] Q. Zheng and K.G.Shin "Fault-Tolerant Real-Time Communication Computing Systems", IEEE FTCS-92
- [2] S. Han and K.G. Shin "Eficcient Spare-Resource Allocation for Fast Restoration of Real-Time Channels from Network Component Failures" IEEE 1997. Pag 99-108.
- [3] P.Ramanathan and K.G.Shin. "Delivery of Time-Critical Messages Using a Multiple Copy Approach". ACM Transaction on Computer Systems, Vol. 10, No. 2, May 1992, Pag. 144-166.
- [4] P.P.White. "RSVP and Integrated Services in the Internet : A tutorial". IEEE Communications Magazine . May 1997 Pág. 100-106
- [5] L. Zhang, S.Deering, D. Estrin, S. Shenker, D. Zappala. "RSVP : A New Resource Rservation Protocol". IEEE Network Magazine September 1993.
- [6] IETF : "Internet Draft: Resource Reservation Protocol (RSVP). Versión 1 Funcional Specification. (expires 14 dec 1997) .
- [7] S. Schenker, C.Partridge, R.Guerin "Specification of Guaranteed Quality of Service", RFC 2212
- [8] J.Wroclawski, "Specification of the controlled-Load network Element Service". RFC 2211
- [9] A. Demers, S. Keshav and S. Shenker, "Analysis and Simulation of a Fair Queuing Algorithm", in Internetworking : Research and Experience, Vol 1, No. 1., pp. 3-26. October 1990. Also in Proceeding of ACM SIGCOMM'89, pp 3-12
- [10] D. Verma, H. Zhang, and D.Ferrari, "Guaranteeing Delay Jitter Bounds in Packet Switching Networks", in Proc. Tricomm '91.
- [11] L. Zhang, "Virtual Clock : A new traffic Control Algrith for Packet Switching Networks", in Proc. ACM SIGCOMM '90, pp. 19-29.
- [12] E.W.Knightly, H. Zhang, "Traffic Characterization and switch Utilization using a Deterministic Bounding Interval Dependent Traffic Model", In Proceeding of IEEE INFOCOM'95
- [13] E. Hernández "Control de la calidad del servicio en redes en tiempo real". DISCA Doctoral report.1998
- [14] E.W.Knightly, R.F.Mines, H. Zhang, "Deterministic Characterization and Network Utilizations for Several Distributed Real-time Applications", In Proceeding of IEEE WORDS'94
- [15] A. Banerjea, D. Ferrari, B. Mah, M. Moran, D. Verma, and H. Zhang, "The Tenet Real-Time Protocol Suite: Design, Implementation, and Experiences", Technical Report TR-94-059, International Computer Science Institute, Berkeley, CA, November 1994; also IEEE/ACM Transactions on Networking, vol. 4, n. 1, pp. 1-10, February 1996.
- [16] D. Ferrari. "Real-Time Communication in an Internetwork", Technical Report TR-92-001, International Computer Science Institute, Berkeley, CA, January 1992; also Journal of High Speed Networks, vol. 1, n. 1, 1992, pp. 79-103, 1992.
- [17] A. Banerjea, D. Ferrari, B. Mah, M. Moran, D. Verma, and H. Zhang, "The Tenet Real-Time Protocol Suite: Design, Implementation, and Experiences", Technical Report TR-94-059, International Computer Science Institute, Berkeley, CA, November 1994; also IEEE/ACM Transactions on Networking, vol. 4, n. 1, pp. 1-10, February 1996.
- [18] A. Banerjea, C. Parris and D. Ferrari "Recovering guaranteed performance service connections from single and multiple faults", Technical Report TR-93-66, International Computer Science Institute, Berkeley, CA.
- [19] S. Han and K.G. Shin. "Experimental Evaluation of Behavior_based Failure-Detection Schemes in Real-Time Communication Networks", To appear in IEEE Transactions In Parallel and Distributed Computing.
- [20] A.S. Tanenbaum. "Computer Networks", 3rd Edition. Prentice Hall Interantional. 1998.

Satélites para la Interconexión en Malla de Redes ATM: Arquitectura y Protocolos

Joaquín Pérez Otero
Telefónica Investigación y Desarrollo
Emilio Vargas 6, 28043 MADRID
Telf: 913373945, Fax: 913374202
E-mail: jperez@tid.es

Carlos Miguel Nieto, Fco. Javier Ruiz Piñar
Dpto. Ingeniería de Sistemas Telemáticos, Universidad Politécnica de Madrid
Ciudad Universitaria s/n, 28040 MADRID
Telf: 915495700, Fax: 913367333
E-mail: cmn@dit.upm.es, fruiz@dit.upm.es

Abstract

Recent standardization works being carried by several international organizations (TIA, ITU, ATM Forum), and the new proposals of commercial satellite systems, specially in Ka band, aim to integrate current and future satellite systems within asynchronous transfer mode -ATM- based terrestrial communications systems. The paper shows the main features of the identified architectures, and analyses the interconnection of both fixed and mobile networks by means of satellites using different orbits, transparent and with on board processing -OBP- capabilities. The relationship between the signaling mechanisms of the interconnected networks is described, and the problems arising from the different satellite medium access methods that can be envisioned are also discussed.

1. Introducción

La tendencia actual de las redes de comunicación de voz y datos es converger hacia una red global que permita ofrecer todo tipo de servicios a los potenciales usuarios compartiendo los recursos y reduciendo por tanto los costes. Esta tendencia se manifiesta en la aceptación de Internet como red global y en la aparición de un número creciente de aplicaciones de red con múltiples contenidos (voz, texto, vídeo, datos, imágenes), hechos que han provocado, por ejemplo, que las redes de telecomunicaciones se planteen el uso de la conmutación de paquetes basada en modo de transferencia asíncrono (ATM), o que las tradicionales redes informáticas se hayan visto obligadas a soportar aplicaciones sensibles al tiempo.

Esta convergencia impone a las nuevas redes de comunicaciones una serie de requisitos que se deben tener en cuenta desde su fase de diseño:

- La demanda de los usuarios de acceder a una red global de servicios integrados, junto con la gran base instalada de redes construidas con los más diversos medios físicos y protocolos de acceso, obligan a considerar el problema de la **interconexión de redes heterogéneas**.
- La **calidad de servicio** como capacidad de la red de garantizar unos niveles acordados de servicio según el comportamiento predecible, o definible, de las aplicaciones, es un aspecto fundamental a tener en cuenta, en especial cuando los operadores de las redes se plantean como estrategia la facturación según la calidad de servicio que ofrecen. El modo de transferencia asíncrono (ATM), además de la solución adoptada por la Red Digital de

Servicios Integrados de Banda Ancha (BISDN) por constituir un mecanismo de conmutación rápida capaz de ofrecer un conjunto de clases de servicio, es una tecnología que en la actualidad cuenta con una base importante de equipamiento instalado, fundamentalmente en el núcleo, o *backbone* de las redes. Así, ATM se perfila como una solución de ingeniería que ofrece mayoritariamente sus capacidades de soporte de calidad de servicio a las redes, más que directamente a las aplicaciones. Otros mecanismos de provisión de calidad de servicio (MPLS [5], IntServ [1], DiffServ [2]) todavía se encuentran en fase de definición.

- Otra característica muy importante que demandan las nuevas aplicaciones multimedia de las redes de comunicación es el soporte de **transmisión a grupos de direcciones destino**, o *multicast*. Para conseguir un transporte eficiente de la información, aplicaciones como la videoconferencia, el trabajo cooperativo o el vídeo bajo demanda requieren esta capacidad de la red.

1.1. Las Redes Satélite en el Entorno de Convergencia

En los últimos tiempos se observa un interés creciente por el desarrollo de una nueva generación de redes satélite capaces de integrarse en el escenario descrito al que tienden las redes terrestres, y de tratar eficientemente tráfico de muy distinto tipo, como el que transporta mayoritariamente Internet, y el que generan las aplicaciones multimedia [7][13][8]. Las características inherentes a las redes satélite, como la provisión de conectividad global, el alcance de áreas de difícil acceso, o el soporte implícito de servicios de transporte *broadcast* y *multicast*, las hace especialmente atractivas en relación con los requisitos

mencionados de interconexión de redes y soporte de servicios multipunto.

Con respecto al requisito de la provisión de distintos niveles de calidad de servicio, han comenzado esfuerzos de estandarización en organismos como la TIA (*Telecommunications Industry Association*), el ATM Forum dentro de su grupo sobre Wireless ATM, o la ITU (*International Telecommunications Union*) [12][4][9], directamente orientados al estudio de los aspectos técnicos del transporte del modo de transferencia asíncrono sobre satélite, lo que se ha dado en llamar SATM, o SATATM.

Este esfuerzo para desarrollar modelos de referencia y reflejar en los estándares las diferentes interfaces involucradas en las redes SATM no es más que la respuesta lógica de los organismos reguladores ante la aparición del gran número de sistemas comerciales de satélite propuestos en todo el mundo para proporcionar servicios de comunicaciones multimedia, principalmente en la banda Ka (20/30 Ghz), la más adecuada para obtener las elevadas tasas de datos que se pretenden, debido a la alta disponibilidad de su espectro. *Spaceway* [17] de Hughes, *Astrolink* [18] de Lockheed Martin, *CyberStar* [19] de Loral, *Teledesic* [20] del Consorcio Teledesic o *SkyBridge* [21] de Alcatel, son ejemplos de sistemas de satélite que ofrecerán diferentes servicios de banda ancha a los usuarios, mediante configuraciones diversas en cuanto a tipo de órbita empleada, número de satélites, tipo de *payload*, o capacidades de conmutación a bordo.

2. Arquitecturas de Red SATM

Una configuración de red satélite se compone de los segmentos espacial, terrestre y de usuario.

El segmento terrestre comprende las Estaciones de Tierra (ETs), el Centro de Control de Red (CCR) y los Centros de Control de Operación (CCOs). Los usuarios se comunican con el satélite a través de las ETs, que se encargan además del control de la posición de los terminales y de los *handoffs* de las conexiones, en caso de sistemas que soportan movilidad. El CCR proporciona fundamentalmente el servicio de gestión y asignación de los recursos de red, pero también el registro de móviles, la tarificación y la distribución de bases de datos de red. Los CCOs a su vez son responsables del seguimiento y control orbital del satélite, y de procesar y distribuir la información obtenida a las ETs. El segmento de usuario lo componen los terminales, fijos, portables o móviles, que requieren servicios de comunicación integrados y globales, mientras el segmento espacial consiste en los satélites, con diferentes posibilidades en cuanto a órbita (geoestacionaria –GEO-, o de media –MEO- o baja –LEO- cota), enlaces entre satélites y capacidad de procesamiento a bordo (OBP).

En los trabajos de estandarización de la TIA se consideran una serie de posibles arquitecturas de red SATM, según el tipo de funcionalidad ofrecida (*relay*, acceso o interconexión), de los usuarios (fijos o móviles), y del tipo de satélite (transparente o OBP, con o sin enlaces entre satélites).

En la Fig. 1 se muestran estas posibilidades. Tanto los satélites transparentes como los dotados de capacidad de conmutación y procesamiento a bordo pueden emplearse para dar acceso a redes ATM terrestres a usuarios fijos y móviles, o para interconectar redes ATM, que también pueden ser fijas o móviles –entre éstas últimas se cuentan las que pueden instalarse en aviones o camiones-. Por otra parte, un satélite transparente puede ofrecer el simple servicio de *relay*, proporcionando un enlace de comunicaciones fijo entre dos o más puntos de tierra, mientras que un satélite OBP puede interconectarse a su vez con otros para formar una red ATM en el espacio, en una configuración mallada.

2.1. Rango de Oportunidades de las Redes SATM

El verdadero rango de oportunidades para este tipo de redes satélite frente a las correspondientes redes terrestres es todavía incierto.

En el programa europeo RACE de principios de los 90 ([16]) se consideraba que el papel de los satélites y su integración en la infraestructura de comunicaciones terrestre de banda ancha seguiría dos pasos evolutivos:

- Una fase de introducción, donde incorporarían tanto la funcionalidad de acceso de los usuarios a las emergentes islas de banda ancha, como la de tránsito, para interconectar dichas islas.
- Una fase de madurez, caracterizada por la solidez de la infraestructura terrestre, en la que representarían el papel de complemento de dicha infraestructura. Se consideran de especial interés sus capacidades de difusión y de comunicación multipunto, y su uso para dar servicio a zonas remotas o áreas en vías de desarrollo, y como respaldo de enlaces terrestres.

Más recientemente, un estudio de tarifas realizado por Swiss Telecom en el marco del proyecto de la Agencia Europea del Espacio (ESA) y Eutelsat

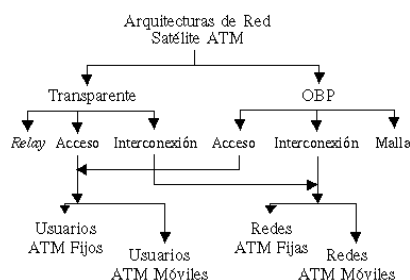
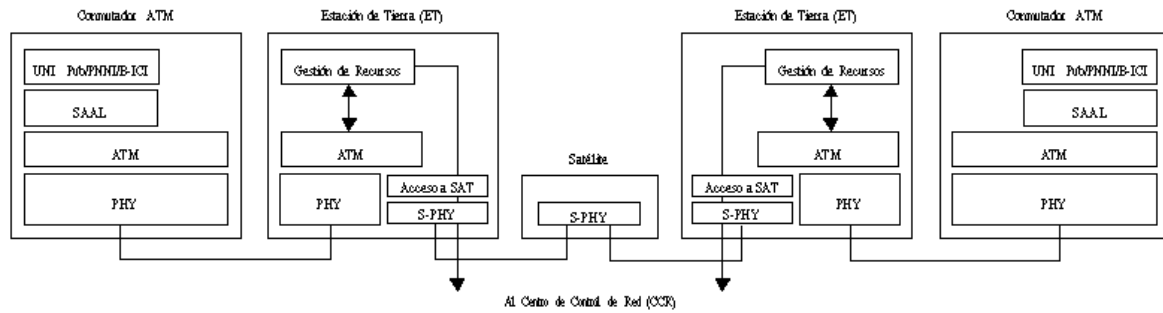


Fig. 1. Arquitecturas de Red Satélite ATM



Broadband Communication over Satellite [15] obtenía como conclusión que, a corto plazo, para pequeñas y medianas distancias, los enlaces punto a punto de satélite no son competitivos en precio. Según esta conclusión, un claro rango de oportunidad lo constituye, para grandes e incluso medianas –por ejemplo, dentro de Europa Occidental– distancias, una red satélite totalmente mallada con un número limitado de nodos. El mismo estudio afirmaba que, a largo plazo, la evolución de las tarifas es muy impredecible, aunque se espera que la desregulación del mercado europeo también afecte a la capacidad de los satélites. En cualquier caso, no considera aceptables configuraciones en las que los usuarios deban pagar el enlace de satélite incluso aunque no lo utilicen, sino que se deben encontrar fórmulas de facturación según un esquema de asignación por demanda de la capacidad disponible.

Otros estudios consideran que el mercado tradicional de satélite de provisión de capacidad de transmisión intercontinental está seriamente amenazado por los sistemas de cable de alta capacidad. Como en la actualidad una de las principales fuentes de valor de las redes de telecomunicaciones está en el tramo de acceso, ha surgido un interés creciente en las comunicaciones de satélite precisamente como tecnología de acceso, en especial en regiones remotas o áreas de baja densidad de población, porque el coste por circuito es casi independiente de la posición geográfica o la densidad de demanda. No obstante, también se estima que sistemas de satélite de banda ancha, como *Spaceway* o *Teledesic*, que pretenden ofrecer servicios de alta velocidad, tanto de acceso como de interconexión, cuentan con un riesgo de mercado bajo, siendo su principal obstáculo el riesgo tecnológico en que se hallan inmersos.

En las secciones siguientes se van a describir diferentes aspectos tecnológicos de la arquitectura de red SATM empleada para la interconexión de redes ATM terrestres.

3. Interconexión de Redes ATM Mediante Satélite

Como se ha mencionado, esta configuración cuenta con un rango de oportunidades de mercado y ha sido, y sigue siendo, objeto de investigación por parte de diversos proyectos.

La interconexión de redes ATM mediante satélite ofrece varias posibilidades, según el satélite sea de

órbita geoestacionaria, o de baja o media cota, o bien sea transparente o cuente con capacidades de procesamiento a bordo. A continuación se describen las características de cada una de estas posibilidades agrupadas en tres categorías. Las dos primeras corresponden a satélites transparentes mientras la tercera emplea satélites con capacidad de conmutación a bordo.

3.1. Arquitectura de Interconexión de Redes ATM Fijas Mediante Satélites Transparentes Geoestacionarios

El escenario de interconexión correspondiente a esta configuración se muestra en la Fig. 2, donde el satélite transparente debe ser del tipo GEO para evitar que las redes que se interconectan parezcan estar en movimiento continuo debido a la órbita de los satélites.

Como se observa en la figura, las interfaces S1.2.A que ofrecen las estaciones de tierra (ETs) a las redes ATM que acceden a los enlaces de satélite pueden ser de tres tipos¹: PNNI (*Private Network Network Interface*), si se interconectan dos redes ATM privadas, UNI (*User Network Interface*) Pública, si la interconexión es entre una red pública y una privada, o B-ICI (*BISDN Inter-carrier Interface*), si se trata de dos redes públicas. La interfaz satélite S1.1.B define el nivel físico, el control de acceso al medio, con o sin calidad de servicio, el control de errores y la gestión de recursos de red. Con respecto al control de acceso al medio y la gestión de ancho de banda, debido a la ganancia de multiplexación estadística que se consigue en la red ATM terrestre, cabe pensar que el tráfico que acceda a los enlaces del satélite sea menos intermitente que los flujos individuales que generan las aplicaciones.

La Fig. 3 muestra el modelo de referencia completo, con los planos de control y de usuario, de los protocolos involucrados en la configuración de red SATM para interconectar redes ATM fijas, con el requisito de emplear un satélite geoestacionario transparente, que ET de red de sat nivel de

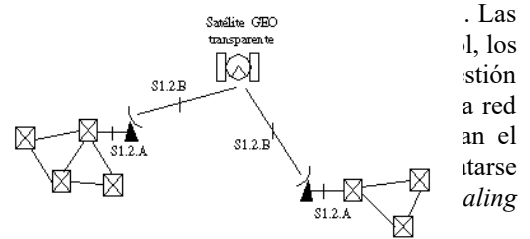


Fig. 2. Interconexión de Redes ATM Fijas Mediante Satélites GEO Transparentes
¹ En orden sucesivo, al referirnos al conjunto de los tres tipos de interfaces, lo haremos como “interfaces NNI” o “NNI”

AAL) para transportar de modo fiable la señalización.

3.2. Arquitectura de Interconexión de Redes ATM con Soporte de Movilidad de Red

La movilidad de las redes ATM que se interconectan puede deberse tanto al carácter móvil de las propias redes como al movimiento de los satélites no geostacionarios (LEO, MEO) empleados. Así, la característica fundamental de esta arquitectura es la

Fig. 3. Modelo de Referencia de Arquitectura SATM para Interconexión de Redes ATM Fijas Mediante Satélites GEO Transparentes

necesidad de gestionar el movimiento relativo de las ETs entre sí y/o con respecto al satélite. La Fig. 4 ilustra el escenario descrito.

La gestión de la movilidad incluye una serie de extensiones a los protocolos de ATM para tratar, entre otros, los siguientes aspectos:

- Control de los *handoffs*, o mecanismos para conmutar dinámica y transparentemente una conexión activa por causa de la movilidad de los terminales.
- Gestión de la posición de los terminales móviles.
- Control del encaminamiento de las conexiones móviles.

La incorporación de estos aspectos requiere básicamente extender el protocolo de interfaz entre nodos de red (NNI, *Network Network Interface*) con nuevas primitivas de señalización ATM, cuyo conjunto compone los mecanismos de movilidad, "M", correspondientes (en la figura, la interfaz S1.2.A es M + NNI). No obstante, en el escenario de movilidad mencionado, no todos los posibles tipos de *handoffs* que pueden ocurrir necesitan el intercambio de primitivas de movilidad para gestionar la conexión. Así, cuando una ET abandona el área de cobertura de un satélite y entra en la de otro, o cuando abandona el área cubierta por un haz, o *spot beam*, y entra en el área de otro haz del mismo satélite, no es necesario reencaminar la conexión a nivel ATM, sino que basta conmutarla al nivel de la interfaz radio. En el caso en que todas, o alguna de las ETs involucradas en una conexión activa quede fuera de toda área de cobertura de algún satélite, sí se hace preciso reencaminar la conexión ATM hacia otra ET que sí esté cubierta, mediante un algoritmo de *handoff* lo más eficiente posible. En este sentido es preferible el mecanismo de *path re-routing* que el de *path extension*, por conseguir una nueva conexión óptima a partir del *crossover switch*, o ancestro común a las dos ETs implicadas en la jerarquía de conmutadores de la conexión. El mecanismo de *path extension*, más simple, sólo extiende la conexión desde la antigua ET hasta la nueva, lo cual introduce un retardo adicional a las celdas de una conexión que ya incluye un enlace de satélite.

Tanto la problemática de los *handoffs* como la gestión de la posición de los móviles y el encaminamiento están siendo tratados por el Wireless Group del ATM Forum. Existen propuestas de nuevas primitivas de señalización para resolver los dos primeros aspectos que sólo se añaden al conjunto de mensajes de los protocolos ATM estándar, lo que facilita su incorporación a la base instalada de conmutadores mediante nuevos módulos *software* [10]. En cuanto a la movilidad en el encaminamiento,

también se pretenden incluir las extensiones correspondientes en la versión 2 de la especificación del PNNI. El empleo de B-ICI en un entorno móvil no es viable debido a la condición estática de sus rutas, basadas en direcciones E.164.

El modelo de referencia de los protocolos para esta configuración se muestra en la Fig. 5. Ahora el conmutador ATM es un MES (*Mobility Enhanced Switch*), es decir, se ha extendido con capacidades para soportar la movilidad de la red, mientras la entidad de gestión de recursos de red de la ET, en plano de control, también incluye funciones para gestionar la movilidad, como los *handoffs* al nivel físico de satélite y el control de la posición del resto de ETs.

3.3. Arquitectura de Interconexión Mediante Satélites OBP

La principal característica de esta arquitectura es el hecho de que el satélite se comporta como un conmutador ATM más. Las interfaces que ofrecen las ETs a las redes ATM terrestres siguen siendo del tipo NNI, pero ahora el satélite interpreta los protocolos ATM, encaminando las celdas que recibe, en función de los valores de su cabecera, hacia el haz adecuado. Incluso es posible dotar a la arquitectura con enlaces entre satélites, para formar una red ATM en el espacio.

El modelo de referencia para este caso sólo difiere de los anteriormente descritos en que el satélite debe incorporar el nivel ATM en su torre de protocolos.

En las secciones siguientes se van a realizar una

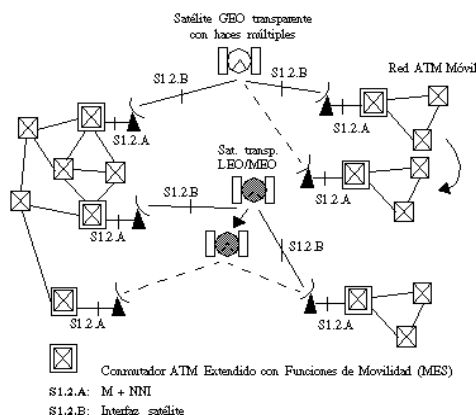


Fig. 4. Interconexión de Redes ATM con Soporte de Movilidad de Red

serie de consideraciones sobre aspectos relacionados con el acceso de las celdas ATM a la red de satélite. En particular, se van a describir las posibles aproximaciones para relacionar la señalización ATM con la señalización de la red satélite, y las posibilidades existentes en cuanto al método de acceso al medio (MAC, *Medium Access Control*) del satélite.

4. Relación entre la Señalización ATM y la de la Red Satélite

En las Figs. 3 y 5 se han presentado unos modelos de referencia en los que aparece una versión simple de las relaciones entre las entidades de protocolos de las ETs que intervienen en el acceso al satélite. A continuación se muestra una visión más detallada, en la que se desdoblán los planos de usuario y de control del modelo de referencia de ATM, para profundizar en las diferentes posibilidades.

La Fig. 6 muestra la aproximación que se emplearía para interconectar redes ATM mediante una plataforma de satélite genérica, no específicamente diseñada para transportar conexiones ATM. Como puede verse en la parte superior de la figura, correspondiente al plano de control, la señalización ATM se transporta de forma transparente sobre la red de satélite. La entidad de gestión de recursos de red de la ET no interpreta por tanto dicha señalización para encaminar y establecer las conexiones dentro de la red satélite, con lo que resulta más difícil mantener adecuadamente la calidad de servicio de las conexiones ATM, tanto de señalización como de datos, dentro de la red de satélite.

Por otra parte, en una aproximación más integrada, los recursos de la red satélite se asignan de forma totalmente relacionada con los recursos de las redes terrestres, a través de la señalización ATM. En la Fig. 7 se detalla el modelo de referencia, en el que se puede ver como aspecto más importante que ahora el CCR también termina la señalización del protocolo NNI, con lo que dispone de la misma información que los nodos de la red ATM terrestre para asignar los recursos de la red satélite, y garantizar así la calidad de servicio de las conexiones. Los niveles de protocolos que aparecen punteados en la figura sólo existen en caso de que el satélite disponga de capacidades de conmutación ATM a bordo, como por otra parte es usual en las propuestas más recientes de sis-

temas de satélite de banda ancha: sistemas como *Spaceway*, *Astrolink* o *N-Star* emplean conmutación a bordo basada en celdas ATM.

Se debe observar que se ha incluido en los diagramas un nuevo nivel, S-ATM, o ATM específico de satélite. Dado que ahora la gestión de recursos de red la realiza directamente la señalización NNI, este nivel pretende cubrir el resto de funcionalidad de la entidad de Red, básicamente el direccionamiento de las ETs, y la funcionalidad del LLC, con el objetivo de reducir el *overhead* de los protocolos. El hecho de no emplear directamente el formato de las celdas ATM estándar para este nivel se debe a la necesidad de contemplar las características particulares del medio de satélite, como se pondrá de manifiesto en la siguiente sección. Un ejemplo de nivel diseñado para que celdas ATM accedan a un medio compartido lo constituye el estándar IEEE 802.14 para la transmisión de datos sobre redes de cable, o redes híbridas de fibra y coaxial. Este estándar define un nivel MAC cuyo formato de paquete es muy similar al de las celdas ATM, con un *payload* de 48 bytes y una cabecera de 6 bytes, uno más que la cabecera estándar ATM. Dicha cabecera, como la de ATM, contiene un VCI (*Virtual Channel Identifier*) para multiplexar circuitos virtuales, y un DLI (*Destination station Local Identifier*) de 14 bits, en lugar del VPI (*Virtual Path Identifier*) de ATM, de 12 bits. A diferencia de éste, el DLI se emplea como verdadera dirección de un módem de usuario, en vez de como etiqueta. El estándar IEEE 802.14 también considera el envío de "fragmentos" de longitud variable, para evitar la ineficiencia que supone la transmisión de 6 bytes de cabecera por cada 48 de datos.

5. Consideraciones sobre el Método de Acceso al Satélite

Según se ha mencionado, el soporte del servicio *multicast* para las aplicaciones multimedia es una característica deseable de las redes de comunicaciones. De hecho, el Grupo de Estudio 13 de la ITU-T ha decidido someter a consideración el problema de la replicación de celdas *en el nivel físico* para aplicaciones de punto a multipunto en sistemas ATM, además de la replicación que puede tener lugar en el nivel ATM o superiores.

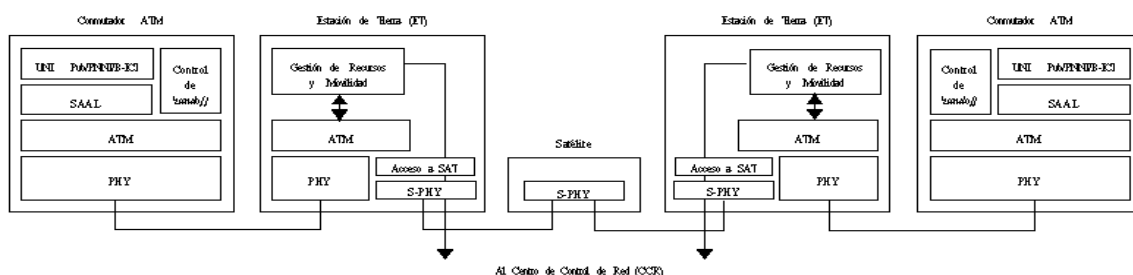


Fig. 5. Modelo de Referencia de Arquitectura SA-IM para Interconexión de Redes ATM con Soporte de Movilidad de Red

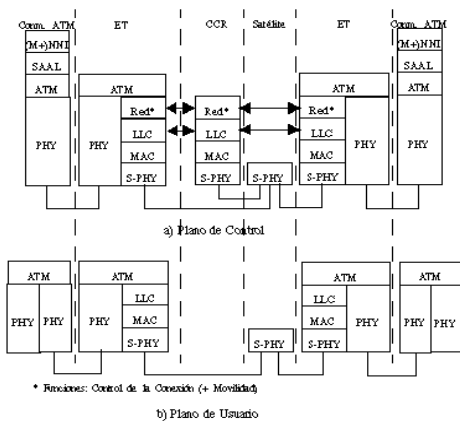


Fig. 6. Modelo de Referencia de Interconexión con Plataforma de Satélite No Orientada a ATM

Un medio compartido como el de satélite ofrece una oportunidad inmejorable para realizar la replicación de celdas en el nivel físico, dependiendo del algoritmo de acceso al medio (MAC) seleccionado. En efecto, el esquema de acceso debe ser capaz de asignar, de manera estática o dinámica, los recursos de la red satélite –canales- a las ETs de forma que grupos de estaciones puedan recibir las transmisiones de las demás.

Independientemente de esta capacidad, como se ha mencionado en la sección 4, la red de satélite puede incorporar un nivel de Red específico, con su propio esquema de direccionamiento, cuando la arquitectura SATM se diseña para transportar de manera transparente la señalización ATM, o sustituir dicho nivel de Red por un nivel S-ATM, cuyo formato de paquete se corresponde con el de la celda ATM estándar, con los mismos identificadores de conexión ligeramente modificados para asegurar su unicidad en el espacio de direcciones del medio compartido de satélite [11]. En éste último caso, si el protocolo de acceso al medio lo permite, es posible utilizar directamente el nivel ATM sobre el enlace de satélite, con lo que las estaciones de tierra se convierten en equipos más sencillos que sólo deben manejar el nivel físico de satélite. El conmutador ATM al que están conectadas recibirá por tanto todas las celdas que el mecanismo de acceso al medio decida entregarle.

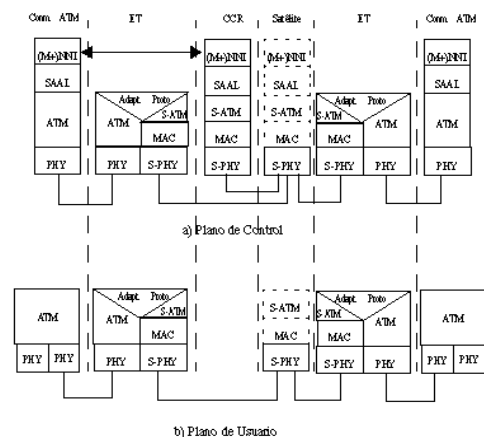


Fig. 7. Modelo de Referencia de Interconexión con Plataforma de Satélite con Señalización ATM

Dos esquemas de acceso al medio que ilustran esta posibilidad se muestran en la Fig. 8 [15]. La parte a) representa la solución basada en la opción multidesdoblado del estándar IDR (*Intermediate Data Rate*), que reparte el ancho de banda disponible asignando bandas de frecuencia diferentes a cada estación de tierra para la transmisión –FDMA, *Frequency Division Multiple Access*-. Todas las ETs, hasta un máximo de cinco, están equipadas con demoduladores para recibir las transmisiones de las demás. La opción que se representa en la parte b) está basada en TDMA (*Time Division Multiple Access*), de forma que cada estación recibe un número de ranuras, o *slots*, de tiempo para transmitir, mientras todas ellas reciben la trama completa en el enlace descendente. Los *buffers* a la salida de los demoduladores y a la entrada del modulador son necesarios para adaptar el flujo de datos de la red ATM a la recepción/transmisión a ráfagas del enlace de satélite debidas a la espera de los *slots* de tiempo asignados.

Estas configuraciones, y en general todas las que se fundamentan en mecanismos de acceso que no permiten el direccionamiento de las ETs dentro de la red de satélite, presentan el problema de que los conmutadores ATM conectados a las estaciones van a recibir posiblemente celdas no destinadas a ellos, es decir, con unos identificadores de trayecto y circuito virtual (VPI, VCI) que desconocen. En ese caso los conmutadores van a descartar las celdas, pero aparte de verse sobrecargados innecesariamente, los contadores de celdas mal encaminadas de sus bases de información se van a incrementar hasta el punto de que pueden generar las correspondientes alarmas. Para evitar esta situación se pueden direccionar las ETs dentro de la red de satélite mediante las entidades de Red, en el caso de redes SATM que encapsulan el protocolo ATM, o mediante el nivel S-ATM específico, en el caso de redes integradas con la señalización ATM, y diseñar las estaciones de tierra de modo que descarten las celdas no destinadas a ellas que reciban del protocolo de acceso, antes de entregárselas al conmutador.

Por último es necesario considerar la problemática de los efectos que provoca un entorno de errores a ráfagas como el de los enlaces de satélite, en especial en la banda Ka, sobre las tramas transmitidas por el protocolo de acceso. Se han desarrollado equipos que llevan a cabo técnicas de codificación de canal eficientes que permiten alcanzar probabilidades de error (BER) comparables a las de los enlaces de fibra óptica, y se han probado sobre enlaces de satélite fijos, del tipo *relay*, para formatos de transmisión digital terrestres como las jerarquías digitales plesiócrona (PDH, *Plesiochronous Digital Hierarchy*) y síncrona (SDH, *Synchronous Digital Hierarchy*), o el protocolo PLCP (*Physical Layer Convergence Protocol*). En estos casos el resultado en términos de BER ha sido satisfactorio [14][3].

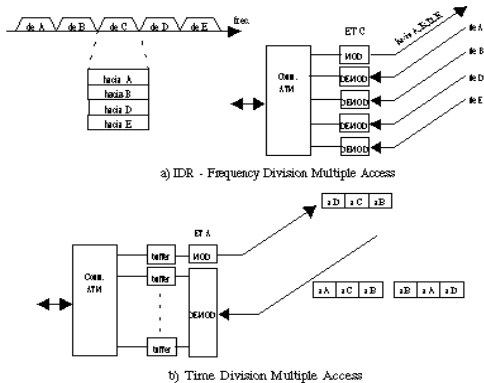


Fig. 8. Métodos de Acceso sin Direccionamiento de Estaciones de Tierra

No obstante, cuando se emplean protocolos de acceso al enlace de satélite con asignación dinámica de ancho de banda (DAMA, Demand Assignment Multiple Access), basados en FDMA, TDMA o MF-TDMA (Multiple Frequency TDMA), lo normal es utilizar un formato de trama acorde con el protocolo de acceso, que consiga un equilibrio entre las necesarias guardas, la ineficiencia introducida por el *overhead* de la trama, y el retardo de empaquetamiento. También en este caso es necesario asegurar que la probabilidad de error no alcanza niveles inaceptables en condiciones de desvanecimiento del enlace. En este sentido hay sistemas ([6]) que proponen contrarrestar dicho desvanecimiento adaptando la energía por bit en función del estado del enlace, compensando la atenuación mediante variaciones de la potencia transmitida y de las tasas binaria y de codificación. Los módems empleados pueden variar dinámicamente la tasa binaria, a cambio de introducir un breve preámbulo entre subráfagas, adicional al largo preámbulo de sincronización entre ráfagas.

6. Conclusiones

En el presente artículo se ha mostrado la necesidad de integrar las redes de satélite en el entorno de convergencia hacia redes globales que soportan múltiples contenidos al que tienden las nuevas redes de comunicaciones. A la luz de los recientes trabajos de estandarización llevados a cabo por diversos organismos (TIA, ITU, ATM Forum), se han presentado las posibles arquitecturas² para integrar los sistemas de satélite actuales y futuros con los sistemas terrestres basados en ATM, y se han apuntado los rangos de oportunidad de mercado de que pueden disponer estas arquitecturas. El breve espacio de tiempo transcurrido desde la aparición de estas líneas de trabajo, junto con el carácter genérico de las definiciones y la falta de certidumbre real acerca de la aplicabilidad comercial de las configuraciones, hace pensar que es necesario

²Los resultados de los análisis llevados a cabo para validar las arquitecturas presentadas se pueden encontrar en las referencias correspondientes.

dedicar aún mucho esfuerzo para concretar las propuestas realizadas.

Una conclusión importante que se extrae de la observación del escenario de integración de redes satélite y terrestres ATM es la constatación del elevado número de posibles configuraciones de red, de la gran cantidad de variables involucradas en la definición de una solución particular, y de la naturaleza rápidamente cambiante de las mismas, lo que dificulta el mantenimiento de los circuitos virtuales ATM si no se diseñan nuevos algoritmos que aprovechen lo predecibles que puedan ser los cambios. Por este motivo, y con el objetivo de reducir el campo de estudio para profundizar en la problemática, se ha seleccionado para su análisis una de éstas configuraciones, la interconexión de redes ATM, y se han identificado tres arquitecturas con características comunes: la interconexión de redes ATM fijas, la interconexión con soporte de movilidad de redes, y la interconexión empleando satélites de tipo OBP. La aproximación adoptada por los grupos de estudio para resolver el problema de la movilidad de terminales pasa por la extensión modular de los protocolos existentes, lo que ratifica la necesidad de adoptar soluciones simples para un escenario de por sí muy complejo.

El análisis de las arquitecturas ha identificado a su vez dos escenarios de integración de la red satélite con las redes terrestres: uno en el que los protocolos ATM se transportan transparentemente sobre la red satélite, y otro en el que la gestión de recursos de la red satélite se integra con la señalización de la red ATM. Se han presentado los modelos de referencia de ambos escenarios, y se ha concluido que la segunda aproximación reduce el *overhead* de niveles de protocolos y permite la extensión de las garantías de calidad de servicio ATM a los enlaces de satélite. La obtención de garantías de calidad de servicio en el protocolo de acceso al medio de satélite debe asumir limitaciones como el retardo inherente a este tipo de enlaces, y su elevada probabilidad de error, y debe abordarse con técnicas integradas de asignación dinámica de ancho de banda y de protección contra errores. En este sentido, el uso de satélites de tipo LEO/ MEO, y en especial con capacidades de procesamiento y conmutación a bordo, de acuerdo con la tendencia de las nuevas propuestas, pueden favorecer dichas garantías.

Referencias

- [1] E.Crawley, et al. "A Framework for Integrated Services and RSVP over ATM". IETF RFC 2382, Agosto 1998.
- [2] S.Blake, et al. "An Architecture for Differentiated Services". IETF RFC 2475, Diciembre 1998.
- [3] D.M.Chitre, et al. "Asynchronous Transfer Mode (ATM) Operation via Satellite: Issues,

- Challenges and Resolutions*". Int'l Journal of Satellite Communications, vol. 12, pp.211-222 (1994).
- [4] "Charter, Scope and Work Plan for Proposed Wireless ATM Working Group". ATM Forum, doc. 96-0530, Abril 1996.
- [5] A.Viswanathan, N.Feldman, Z.Wang, R.Callon. "Evolution of Multiprotocol Label Switching". IEEE Communications Magazine, vol. 36, no. 5, pp. 165-173. Mayo 1998.
- [6] N.Celandroni, E.Ferro, F.Potortì. "Experimental Results of a Demand-Assignment Thin Route TDMA System". Int'l Journal of Satellite Communications, vol. 14, pp.113-126 (1996).
- [7] J.Montgomery. "Fiber in the Sky". BYTE Magazine, pp. 58-72. Noviembre 1997.
- [8] P.Chitre, F.Yegenoglu. "Next Generation Satellite Networks: Architectures and Implementations". IEEE Communications Magazine, vol. 37, no. 3, pp. 30-36. Marzo 1999.
- [9] "Performance for B-ISDN Asynchronous Transfer Mode (ATM) via Satellite". Working Party 4B, doc. 4B/TEMP/25, as part of Draft New Recommendation ITU-R S.[DOC. 4/44]. Diciembre 1998.
- [10] "Primitives for Location Management and Handoff in Mobile ATM Networks". ATM Forum, doc. 96-1121, Agosto 1996.
- [11] I.Mertzanis, G.Sfikas, R.Tafazolli, B.G.Evans. "Protocol Architectures for Satellite ATM Broadband Networks". IEEE Communications Magazine, vol. 37, no. 3, pp. 46-54. Marzo 1999.
- [12] "Satellite ATM Networks: Architectures and Guidelines". TIA/EIA Telecommunications Systems Bulletin (TSB 91), Mayo 1998.
- [13] C-K.Toh, V.O.K.Li. "Satellite ATM Network Architectures: An Overview". IEEE Network, vol. 12, no. 5, pp. 61-71. Septiembre/Octubre 1998.
- [14] I.Akyildiz, S-H.Jeong. "Satellite ATM Networks: A Survey". IEEE Communications Magazine, vol. 35, no. 7, pp. 30-43. Julio 1997.
- [15] B.Perrin, R.Donadio. "Satellite full mesh ATM LAN: interconnection, satellite access scheme, signaling and performance". Proceedings of Broadband Communications '96. IFIP/Chapman &Hall, 1996, pp. 346-357. ISBN:0-412-75970-5.
- [16] "Satellites in the B-ISDN, General Aspects". RACE Specification D751. IBC Common Functional Specification, Issue D. Diciembre 1993.
- [17] <http://www.hns.com/spaceway/spaceway.htm>
- [18] <http://www.astrolink.com>
- [19] <http://www.cyberstar.com>
- [20] <http://www.teledesic.com>
- [21] <http://www.skybridgesatellite.com>

Gestión Integral de la Red de Señalización de Telefónica: Sistema SERES

Antonio Arias*, Juan Carlos Luengo**

* Tfno: 91-3379827, Fax: 91-3374222, E-mail: ariasd@tid.es

** Tfno: 91-3379306, Fax: 91-3374602, E-mail: luengo@tid.es

Telefónica I+D, C/ Emilio Vargas 6, 28043 MADRID

Abstract

Since signalling network is the nervous system of today's telecommunication networks and services, its management is an issue of major importance. This paper presents the management approach taken by Telefónica in order to control and to obtain a good performance from its Number 7 signalling network, that is, in order to give a high quality in all kind of services (Plain Telephony, Intelligent Network Services, Mobile Services, etc.). The different layers of SS#7 architecture are treated individually, considering the existing recommendations and/or the approach taken in the implementation of the management function. Finally, a general view of SERES (Management System of Signalling Network) and its functioning is presented.

1. Introducción

La gestión de la red de señalización es, hoy en día, uno de los aspectos más importantes de la gestión de las redes de telecomunicación. Desde el momento en que todos los servicios actuales de telecomunicaciones tienen como base las capacidades de la red de señalización, es de vital importancia tener un control adecuado de la misma. Surge por ello la necesidad de realizar una Gestión Integral del SS#7 (Sistema de Señalización N° 7) de Telefónica.

El SS#7 estandarizado por la UIT-T [1], ampliamente conocido y utilizado a nivel internacional, está todavía siendo desarrollado en sus niveles superiores, aquellos equivalentes a los niveles superiores de la arquitectura OSI, en algunos aspectos.

Estos aspectos están relacionados con, por ejemplo, la aparición de nuevos servicios y funcionalidades que son necesarias considerar a la hora de implementar las capacidades de los elementos de la red de señalización. La continua evolución de los mismos provoca continuos cambios en las recomendaciones asociadas.

1.1. Gestión de las partes de la red de señalización

Otro de los puntos no totalmente desarrollados es el que se trata en esta contribución: el de la gestión de la red. Las recomendaciones existentes cubren este aspecto únicamente para las capas que componen la PSR (Parte de Servicio de Red), suma de la PTM (Parte de Transferencia de Mensajes) y la PCCS (Parte de Control de las Capacidades de Señalización). Para las capas superiores, partes de usuario, no se ha desarrollado este mecanismo, aunque existe la denominada POMA (Parte de Operación, Mantenimiento y Administración).

En una primera aproximación, puede no parecer necesario ni conveniente que la gestión de las partes de usuario se relacione con la gestión de la PSR, puesto que las primeras podrían funcionar sobre cualquier arquitectura que cubriera los

requerimientos de las tres capas inferiores del esquema de referencia OSI.

Ahora bien, considerando la estructura final y la evolución del SS#7, la relación entre ambas partes es muy importante. Y dada esta relación, es necesario desarrollar algún mecanismo que permita determinar la influencia que, en ambos sentidos, se da entre las partes de usuario y la PSR.

Surge entonces la necesidad de determinar como la introducción de esas nuevas partes de usuario influye en el funcionamiento general de la red y en su capacidad para, soportando los nuevos servicios con la calidad necesaria, mantener el número y la calidad de los servicios previamente existentes.

Así mismo, hay que determinar de qué forma el funcionamiento de estos nuevos servicios puede verse afectado por fallos o mal funcionamiento de las partes inferiores, que dan soporte a las aplicaciones superiores.

Por lo tanto, es necesario realizar un estudio detallado de cada uno de los niveles que conforman la pila del protocolo de señalización. Este estudio deberá determinar, desde un punto de vista general, cuáles son los parámetros que deben ser monitorizados para mantener el control de la red. También deberá considerar las particularidades de cada uno de los niveles, tanto desde el punto de vista de la información a manejar (diferente según la parte considerada), como desde el punto de vista de la obtención de dicha información. Así, la interrelación con los sistemas de gestión propios de cada una de las estructuras de las partes de usuario consideradas, puede ofrecer importante información para la gestión del SS#7.

Todos estos aspectos son cubiertos por SERES (Sistema de Explotación de la RED de Señalización) proporcionando un mecanismo integrado de gestión de la red y que, en su evolución, deberá tener en cuenta las nuevas opciones que las partes de usuario ofrecen para su gestión.

2. Gestión de la PSR

El primer objetivo de un sistema que pretende gestionar la red de señalización debe ser el soporte

del intercambio de mensajes. Es decir, las primeras capas, aquellas que solucionan los problemas de envío seguro, sin duplicidad y sin pérdida desde un nodo a otro: PTM y PCCS

2.1. PTM

El modelo de red para la gestión de PTM está suficientemente detallado en la recomendación Q.751.1 de la ITU-T.

Por lo tanto, aquí se da un breve repaso a los conceptos (sobre todo a nivel de red), haciendo hincapié en los diferentes enfoques que suponen la supervisión y la planificación de la red.

2.1.1. Modelo de red

Para una descripción formal de los objetos que componen el modelo es necesario considerar la recomendación citada anteriormente. Aquí se presenta una idea topológica de ellos para poder después utilizarlos en el resto del artículo:

Punto de Señalización: Cada nodo en la red de señalización. Hay tres tipos, dependiendo de su función: PES (Punto Extremo de Señalización, que contendrá sólo funcionalidad de extremo), PTS (Punto de Transferencia de Señalización), PETS (Punto Extremo y de Transferencia de Señalización).

Enlace de Señalización: Transporta mensajes de señalización de un Punto de Señalización a otro.

Conjunto de Enlaces de Señalización: Es un grupo establecido de uno o más Enlaces de Señalización entre dos Puntos de Señalización.

Ruta de Señalización: Cada uno de los posibles caminos físicos para llevar mensajes de señalización desde su origen (OPC) hasta su destino (DPC).

Conjunto de Rutas de Señalización: Grupo de Rutas de Señalización existentes entre origen (OPC) y destino (DPC).

2.1.2. Actividades de gestión

En general, un sistema de gestión tiene como principal objetivo resolver los problemas de supervisión y planificación de la red.

Supervisión

El conocimiento de la topología de la red es un requisito imprescindible para realizar una buena supervisión. Para ello, actualmente, todas las tecnologías de central utilizadas en la red de Telefónica permiten la consulta desde un sistema remoto de gestión de su configuración SS7.

Es este conocimiento de la topología lo que permite interpretar las situaciones anómalas que se puedan detectar a través de alarmas y medidas de uso o de calidad de la red. Las situaciones más importantes son:

- Fallo de enlaces de señalización. Unido a la configuración, informará de la disponibilidad del conjunto de enlaces.
- Fallo de conjunto de enlaces. Unido al conocimiento de las tablas de encaminamiento

permite conocer a qué relaciones de señalización afecta el fallo y si existen alternativas.

- Fallo de conjunto de rutas. Indica la imposibilidad de intercambio de mensajes de señalización entre dos nodos de la red (y por lo tanto, el fallo del servicio, o su degradación)
- Sobrecarga en enlaces de señalización. Permitirá detectar con suficiente antelación situaciones de congestión de los enlaces, tomando entonces las medidas oportunas (alta de nuevos enlaces o reencaminamiento del tráfico).
- Medidas de calidad de la red: Parámetros como % de retransmisiones o retardo en la recepción de ACK, nos permitirá detectar situaciones de degradación de la red.

Una vez cubierta la detección de las situaciones anómalas queda por cubrir la realización de modificaciones en la red (reencaminamiento de tráfico) para solventarlas. Para ello resulta de nuevo imprescindible el conocimiento de la topología en tiempo real para evitar la creación de bucles, falta de bidireccionalidad, etc

Planificación

El dimensionado de la red de señalización, durante mucho tiempo, se ha basado en el número de llamadas que debían intercambiar los nodos. Actualmente, las aplicaciones de servicios móviles y red inteligente hacen que aparezca señalización no asociada a llamada, quedando así obsoleto el mecanismo de dimensionado.

Para solucionarlo, es necesario que el sistema de gestión sea capaz de elaborar matrices de tráfico origen-destino para cada pareja de nodos que exista en la red. De esta forma, una vez conocidos estos datos y junto con la topología actual de la red, se podrán detectar futuros cuellos de botella y plantear soluciones de evolución, que pueden ser:

- Optimización de los encaminamientos de tráfico
- Aumento de recursos en la red (alta de enlaces de señalización)

2.2. PCCS

A diferencia de PTM, en el caso de PCCS sólo existe una recomendación para el modelo de gestión a nivel de elemento de red (Q.751.2 del ITU-T). Tras repasar brevemente los conceptos que se manejan en ella, posteriormente se presentará el modelo de nivel de red que se ha elaborado y utilizado en Telefónica.

Por último, se mencionan algunos puntos que diferencian la gestión de PCCS de la de PTM

2.2.1. Modelo de elemento de red

Se presenta a continuación el modelo de gestión que aparece en la recomendación Q.751.2 de la ITU-T.

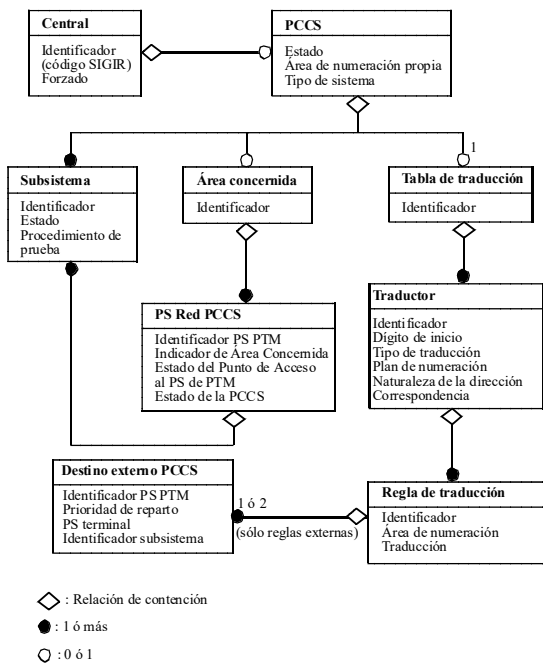


Figura 1.- Objetos del modelo de elemento de red

Central: Es el nodo de telecomunicaciones que contiene la PCCS.

PCCS: Es el contenedor de los objetos gestionados de la PCCS. Contiene tres grandes partes: los subsistemas, el área concernida del nodo PCCS y la tabla de traducción.

Subsistema: Cada uno de los usuarios de PCCS que existen en el nodo.

Área concernida: Esta clase de objeto contiene la lista de nodos PCCS remotos que deben ser informados del estado de la PCCS local y de sus subsistemas asociados.

Tabla de traducción: Esta clase de objeto contiene la información de gestión concerniente al encaminamiento PCCS. Está formado por un conjunto de Traductores (cada tabla de traducción) y de Reglas de Traducción.

2.2.2. Modelo de red

Una versión simplificada del modelo de red que se maneja en Telefónica es el siguiente:

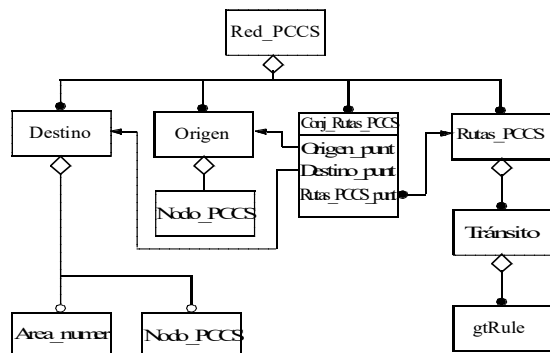


Figura 2.- Objetos del modelo de red simplificado

Red PCCS: Está formada por Nodos PCCS (origen o destino) y relaciones de señalización entre ellos (Conjunto de Rutas PCCS)

Origen: Es un nodo PCCS cuyos subsistemas generan tráfico.

Destino: Si el nodo PCCS al que van destinados los mensajes está gestionado, se conoce la numeración de títulos globales que maneja como propia y, por lo tanto, se sabe a qué destino real apunta una regla de traducción del nodo origen.

Sin embargo, si no está gestionado el nodo PCCS destino, sólo se conoce un área de numeración, hacia la que se encaminarán los mensajes (según la regla de traducción del nodo origen y de los posibles tránsitos PCCS a través de nodos “relay”).

Conjunto de Rutas PCCS: Está formado por todos los posibles caminos que puede seguir un mensaje desde un origen hasta su destino.

Ruta PCCS: Es cada uno de los posibles caminos que puede seguir un mensaje desde el origen hasta su destino. Está formada por diferentes saltos entre nodos que realizan traducciones PCCS. A cada uno de estos saltos se le denomina Tránsito y se corresponde con un Conjunto de Rutas de PTM.

2.2.3. Actividades de gestión

Al igual que en el caso de PTM, el sistema de gestión debe facilitar el desempeño de dos actividades básicas: supervisión y planificación.

Supervisión

En la gestión de PCCS también tiene un papel primordial el conocimiento de la configuración de las centrales y la topología de la red. Por lo tanto, el primer nivel que debe resolver un sistema de gestión es el de conocer esa topología.

A partir de aquí se pueden detectar situaciones de fallo o degradación de la red, analizar el impacto que puedan tener en el tráfico cursado y tomar decisiones para superar estas situaciones (fundamentalmente con reencaminamiento de tráfico). Las situaciones anómalas más importantes que se deben detectar son:

- Indisponibilidad en Conjuntos de Rutas PTM que afectará al tráfico PCCS que, desde ese origen, utilice como siguiente nodo de traducción el destino del CRTA de PTM.
- Imposibilidad de realizar traducciones de direcciones PCCS en un nodo. Esto indicará que las tablas de traducción presentan deficiencias o que algún nodo en la red está enviando tráfico con direcciones equivocadas.
- Problemas con el segmentado / reensamblado de mensajes PCCS.
- Indisponibilidad o congestión en la PCCS de una central.
- Indisponibilidad o congestión en los subsistemas de un nodo PCCS.
- Paso de nodo principal a nodo de “back-up”. Esto supone un cambio en el encaminamiento del tráfico que debe ser especialmente vigilado,

ya que puede provocar sobrecargas en la red o congestión en el nodo de “back-up”.

Planificación

Ya se ha comentado que el dato fundamental en la planificación de la red a nivel PTM es el tráfico por OPC-DPC. Sin embargo, cuando tenemos tráfico PCCS, esta medida no está asociada realmente a la idea de origen y destino, ya que los campos OPC y DPC de los mensajes cambian con cada traducción PCCS.

La medida que se debe utilizar en este caso es el tráfico desglosado por los siguientes campos claves:

- Código de punto de señalización del nodo PCCS que genera el tráfico.
- Rango de títulos globales que aparecen en la dirección llamada PCCS.

Puesto que el sistema de gestión conoce la configuración, sabrá asociar cada uno de los rangos de títulos globales a un nodo PCCS de la red.

Una vez conseguidas las matrices de tráfico PCCS como se ha descrito, se “cruzarán” con los datos de encaminamiento para detectar posibles congestiones futuras en la red.

Por último, del procedimiento de planificación deben salir propuestas de evolución de la red como:

- Modificaciones de los encaminamientos PCCS
- Alta de nuevos recursos (enlaces de señalización, nodos de traducción PCCS, etc)

3. Gestión de las Partes de Usuario

Las partes de usuario que se presentan a continuación son las que mayor impacto tienen en la actualidad. La introducción de nuevos servicios (como elemento diferenciador o simplemente como obligación de los organismos regulatorios) y el auge de la telefonía móvil obligan a tener una consideración especial a la PARI (Parte de Aplicación de la Red Inteligente) y a la PAM (Parte de Aplicación de Móviles).

Sin olvidar a la PUSI (Parte de Usuario de la RDSI), que es la encargada del manejo de las más comunes comunicaciones de telefonía, es necesario desarrollar mecanismos que permitan la gestión de la red de señalización en las nuevas condiciones que, bajo estos protocolos, aparecen.

Mientras que la PUSI presenta un protocolo orientado a la llamada, al establecimiento de la comunicación (con unas características u otras, pero relacionada con el establecimiento de la comunicación al fin y al cabo), las nuevas partes de usuario manejan una gran cantidad de información no directamente relacionada con el establecimiento de la comunicación.

Para todo este proceso se utiliza la PACT (Parte de Aplicación de las Capacidades de Transacción), capa del protocolo que se presenta a continuación y que se encuentra indisolublemente ligada a las partes de usuario que se explican posteriormente.

3.1. PACT

La PACT proporciona funciones y protocolos a una gran variedad de aplicaciones distribuidas entre centrales y centros especializados (por ejemplo, bases de datos) de las redes de comunicaciones.

Básicamente, la PACT es un conjunto de facilidades de comunicación que actúa de interfaz entre las aplicaciones y el servicio de la capa de red y, como tal, sólo tiene sentido como parte integrante de una de las aplicaciones que la requieran.

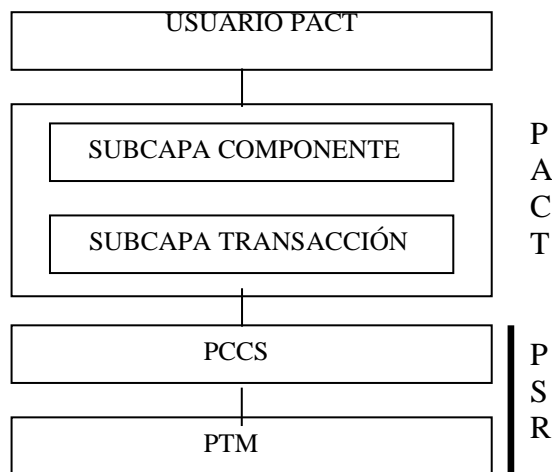


Figura 3.- Estructura de la PACT

Se presentan a continuación los conceptos básicos relacionados con el funcionamiento de la PACT.

Componente: Un componente es el medio por el cual la PACT transporta una petición de operación o una respuesta. La invocación de una operación se identifica mediante un identificador de invocación, lo que permite tener activas simultáneamente varias invocaciones de la misma o de varias operaciones.

Diálogo: Un diálogo es el intercambio sucesivo de componentes con el objetivo de llevar a cabo una aplicación. Un diálogo puede ser estructurado o no estructurado.

La PACT está dividida en dos partes (Figura 1).

Subcapa Transacción: La subcapa transacción proporciona la capacidad de intercambio de mensajes que contienen componentes. Los paquetes que trata la subcapa de transacciones son:

Mensajes BEGIN (establece un nuevo diálogo), *CONTINUE* (lleva componentes dentro de un diálogo), *END* (acaba un diálogo), *ABORT* (acaba prematuramente un diálogo, debido a algún error).

Subcapa Componente: La subcapa componente acepta una operación de la parte de usuario y la formatea como un componente PACT. Hay varios tipos de componentes PACT:

Componentes: *INVOKE* (lleva una operación que debe ser ejecutada), *RETURN RESULT* (lleva el resultado de una operación), *RETURN ERROR* (lleva información sobre un error en una operación), *REJECT* (enviar información sobre un error en un componente).

3.3. PAM

Al igual que ocurría en el caso de la PARI, la gestión de la Red Móvil en sí misma es algo que se encuentra fuera del proceso de gestión del SS#7. Y en este caso surge la dificultad añadida de que se desea controlar una red sobre cuyo sistema propio de gestión, a diferencia con la RI Normalizada, está fuera del control de Telefónica, puesto que pertenece a Telefónica Servicios Móviles.

Por lo tanto, las medidas de utilización a obtener (por servicio o en función de lo que se desee controlar) deberán serlo a partir de la observación (por cualquiera de los medios disponibles) de los mensajes PAM que circulan por la red.

Así podrán controlarse cuestiones como la cantidad y tipo de tráfico que la misma soporta que, junto con el resto de informaciones disponibles (considerando las operaciones PAM entre operadoras y la gestión desde diferentes capas del protocolo, a continuación), permitirán la obtención de una serie de parámetros que ofrecerán información sobre la utilización de la red por servicio.

Gestión desde las capas inferiores del protocolo

Esta línea de gestión utiliza las capacidades de gestión de las diferentes capas inferiores del protocolo SS#7 para llevar a cabo la gestión de la PAM. Esta misma aproximación puede ser utilizada para la gestión de la PARI, aunque es ese caso, y puesto que está disponible, se ha hecho hincapié en la gestión considerando el SMS.

Este método pretende recopilar la información que las capas inferiores ofrecen en relación a la Parte de Usuario considerada, pudiendo así, al menos, informar de su funcionamiento.

Gestión desde la PACT: No es aplicable una gestión de la PAM desde esta capa, ya que en este caso concreto, está integrada en los elementos que componen la red de móviles, y no realmente en la red de Telefónica. Por lo tanto, desde el punto de vista de SERES, no tiene sentido tratar de controlar el éxito o fracaso de las operaciones que se realizan entre nodos que no pertenecen a la red a explotar,

Por ello es más adecuado un enfoque de gestión en el que se determine la utilización de los recursos propios, discriminando ésta por servicios. La gestión de la PAM en sí misma queda entonces en manos de Telefónica Móviles.

Gestión desde la PCCS: La gestión desde la PCCS tiene un funcionamiento diferente al de cualquier otra parte de usuario (como, por ejemplo, la PARI). El hecho histórico de que se consideren varios subsistemas diferentes para cada uno de los elementos de la red, la estructura claramente diferenciada de la misma (controlada por Telefónica Móviles) y, por lo tanto, la gestión externa que de los elementos implicados se realiza, dificulta la gestión de esta capa desde la PCCS.

Así, la gestión en este caso se basará en utilizar aquella información, extraída de las capacidades de Gestión de la PCCS (comentadas en el apartado 2.2 de esta contribución), que pueden dar una información general relacionada con la PAM,

existiendo diferencias entre la información que se puede obtener en función del equipo utilizado.

4. Sistema SERES

SERES (Sistema de Explotación de la Red de Señalización) es la solución del grupo Telefónica para gestionar la red de señalización. Cubre las siguientes funcionalidades:

- **Gestión de configuración:** Comprende actividades como la auditoría periódica de la configuración de las centrales, herramientas de consulta gráfica y textual de ésta, planificación de modificaciones en la red, implantación de estas modificaciones y herramientas de validación de configuración
- **Gestión de alarmas:** SERES recoge las alarmas de señalización que generan las centrales, las correlaciona y analiza el impacto que suponen en la red.
- **Gestión de prestaciones:** Con un periodo de 5 minutos se recogen medidas de la red y se generan excepciones al detectar sobrecargas, retransmisiones excesivas, etc.

4.1. Gestión de configuración

De forma automática, o bajo petición de un operador, SERES realiza auditorías de la configuración de las centrales para conocer:

- Configuración PTM:
 - Enlaces y conjuntos de enlaces de señalización.
 - Tablas de encaminamiento PTM
 - Parámetros de nivel 3 del punto de señalización.
 - Parámetros de nivel 2 del punto de señalización.
- Configuración PCCS:
 - Tablas de traducción de títulos globales.
 - Subsistemas del nodo
 - Lista de puntos de señalización del área concernida

Así, lo único que hace falta configurar de forma manual en el sistema son los puntos de señalización que hay en la red. A partir de ahí, el sistema se encarga, de forma automática, de mantener actualizada el resto de la información topológica (enlaces, conjuntos de enlaces, tablas de encaminamiento, etc.). Para ello utiliza conexiones con las centrales a través de una red X.25 y envía, periódicamente o bajo demanda, los comandos necesarios para conocer toda la configuración dada de alta en cada central.

Todos estos datos se almacena en una base de datos que será el "corazón" del sistema y es utilizada por herramientas textuales de consulta o herramientas gráficas de navegación:

- Enlaces que componen un conjunto de enlaces.
- Las diferentes rutas que se utilizan para llegar a un destino.
- Qué rutas se verían afectadas por el fallo de un conjunto de enlaces.
- etc.

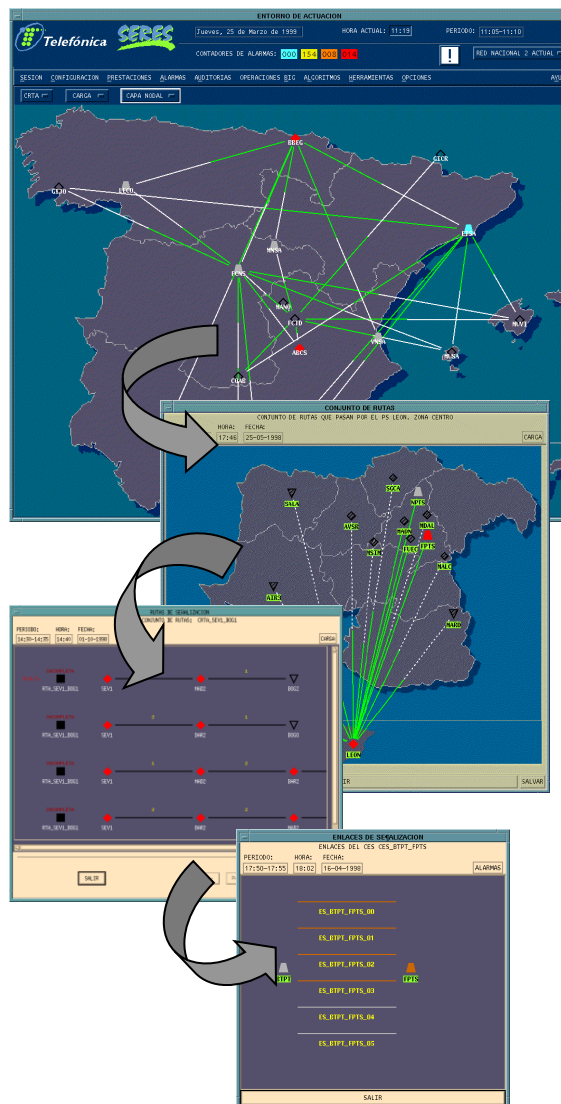
Estas herramientas permiten conocer el impacto que tienen los fallos o sobrecargas que se detecten con las otras funcionalidades del sistema.

La Figura 6 presenta un ejemplo de navegación que permitirá a un operador conocer cualquier detalle de la red y, de esa forma analizar cualquier problema que pudiera tener. Los diferentes pasos que se muestran son:

1. Ventana principal del SERES. En estos mapas se representan las posibles incidencias que presente la red.
2. Conocer las conexiones de cualquiera de los puntos de señalización hacia determinada zona.
3. Preguntar por los diferentes caminos (rutas) que se utilizan para mandar la señalización entre dos nodos.
4. Para uno de los conjuntos de enlaces que aparecen en la ventana anterior, conocer sus enlaces y en qué estado se encuentran.

Figura 6.- Pantallas de navegación de SERES

SERES también tiene perfiles de usuario que



permiten realizar una planificación completa de la red:

1. Detección de congestiones, falta de redundancia, etc.

2. Propuesta de evolución de la red. Se realizan validaciones sobre esta nueva configuración para detectar bucles, rutas demasiado largas, malos comportamientos ante fallos simples y graves, etc.
3. Generación de cambios necesarios.
4. Secuenciación de esos cambios. Esto es necesario para evitar que las diferentes situaciones intermedias por las que se pase puedan provocar problemas en la red.
5. Implantación de cambios. Aislando al operador de los comandos particulares de cada tecnología de central, se van realizando las modificaciones previstas en los pasos anteriores (alta y baja de enlaces, modificación de las tablas de encaminamiento, etc)

4.2. Gestión de alarmas

El sistema SERES realiza una gestión de fallos en tiempo real. Recibe alarmas de las centrales y analiza el impacto que puedan tener en la red. Un simple fallo de un señalizador es analizado por el SERES para averiguar si quedan más enlaces disponibles o si hay encaminamientos alternativos. Se correlacionarán las alarmas y, cruzándolas con la información de configuración, se presentará al operador aquellas que tengan mayor gravedad. Las alarmas de señalización se presentan en los mapas presentados en el apartado anterior o en listas ordenadas por gravedad. Un ejemplo de esto último aparece en la Figura 7.

ID. ALARMA	ID. OBJETO	T. DETECCIÓN	CAUSA	CORRELACION
66199	CRTA_SEVI_HONU	17/11/97 02:06:46	EL EVENTO DEGRADADO	PRINCIPAL
60977	CRTA_BILB_SEVI	12/11/97 02:08:22	EL EVENTO DEGRADADO	PRINCIPAL
60978	CRTA_SSEB_SEVI	12/11/97 02:08:22	EL EVENTO DEGRADADO	PRINCIPAL
24663	CRTA_SEVI_MSIM	28/10/97 13:21:40	EL EVENTO DEGRADADO	PRINCIPAL

Figura 7.- Ejemplo gráfico de la gestión de alarmas

Esta herramienta permite preguntar por las causas de los fallos y el sistema proporcionará un análisis de los distintos problemas de la red que han dado lugar a esta situación y su evolución hasta llegar al momento actual.

4.3. Gestión de prestaciones

Cada 5 minutos, el SERES analiza medidas de la red de señalización y elabora los parámetros que aparecen en la recomendación Q.752. Algunos de ellos son: carga, retransmisiones y fallos en los enlaces, tiempo de inaccesibilidad de destinos, distribución de tráfico por servicio, etc. Estos parámetros son sometidos a umbrales ajustables por el operador, tal y como muestra la Figura 8, y de esta comparación aparecen excepciones (alarmas) que serán presentadas al operador como problemas de la red.



Figura 8.- Ejemplo gráfico de la gestión de prestaciones.

Los datos que se han ido recibiendo cada periodo, son almacenados en una base de datos que permitirá realizar informes de tráfico, calidad y fallos de la red:

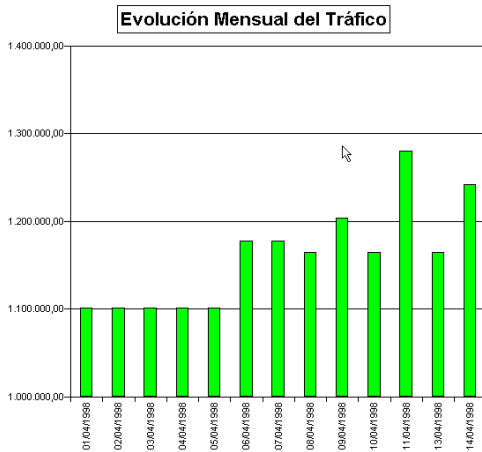


Figura 9.- Ejemplo de informes de resultados.

Cada día se calculan informes de periodo cargado que darán lugar a matrices de tráfico Origen-Destino, imprescindibles para la posterior tarea de planificación.

5. Conclusiones

Este artículo presenta la aproximación de Telefónica a la Gestión Integral de la Red de Señalización SS#7. En un primer momento, se ha presentado, para aquellos niveles del protocolo de señalización que lo poseen, el mecanismo de gestión existente en las recomendaciones asociadas. Posteriormente, se ha presentado, de forma genérica, las diferentes maneras de implementar la gestión para cada una de las Partes de Usuario, considerando sus particularidades.

Como ha podido verse a lo largo de este artículo, las posibilidades de acceso a la información relativa a los elementos de red determina de alguna forma el modelo final de gestión a considerar. Aunque existe la opción de utilizar las posibilidades de gestión que ofrecen las capas inferiores del protocolo, éstas son, hoy por hoy, bastante reducidas.

Se presenta brevemente la herramienta SERES y se explica su funcionamiento, indicando de forma genérica la información de entrada requerida y la de salida proporcionada. Y, para cada una de las partes de usuario mencionadas, y antes de entrar en detalle en la revisión de la herramienta, se presentan las

opciones de gestión junto con los sistemas propios de esas redes (Inteligente y Móviles).

Así, mientras que en el caso de la PARI es posible un control más exhaustivo debido a la posibilidad de interconexión y de intercambio de información entre el SERES y el SMS, en el caso de la PAM esto no es factible en un primer momento, lo que limita de alguna manera las posibilidades de actuación.

Por ello es necesario buscar vías alternativas que permitan la obtención de la información relativa a los servicios móviles, que hacen un uso intensivo de la red de señalización. En esta nueva aproximación, entra en juego la posibilidad de utilizar nuevos sistemas que faciliten la obtención de esos datos, como es el IAS (Interfaz de Acceso a Sondas).

El IAS es un sistema de mediación que, mediante un sistema de sondas no intrusivas, extrae y almacena información de señalización de la red. Posteriormente, será necesario desarrollar una aplicación superior que, a partir de los datos almacenados, extraiga los parámetros que se deseen manejar.

Agradecimientos

Al grupo de desarrollo SERES de Telefónica I+D por su profesionalidad y buen hacer en la realización de un sistema que completa las necesidades de gestión del grupo Telefónica.

Al grupo de Inteligencia y Señalización, por su colaboración en los trabajos relacionados con SERES.

Referencias

- [1] ITU-T, Recomendaciones de la Serie Q.7XY.
- [2] Telefónica I+D, "Estudio sobre la Gestión de la PCCS ". Edición 2.0, Diciembre 1998, Código: 901342-185-4002.
- [3] Telefónica I+D, "Estudio sobre la Gestión de la PARI ". Edición 1.0, Diciembre 1998, Código: 901342-185-4001.
- [4] Telefónica I+D, "Estudio sobre la Gestión de la PAM ". Edición 1.0, Diciembre 1998, Código: 901342-185-4000.

Control de tráfico mediante Class-Based Queuing gestionado por RSVP usando Linux como router con capacidad QoS

A. Gómez Skarmeta, A. López Toledo
Departamento de Informática, Inteligencia Artificial y Electrónica, Universidad de Murcia,
Campus de Espinardo s/n 30008 (MURCIA)
Telf: 968-364607, Fax: 968-364151
E-mail: {skarmeta, alberto}@dif.um.es
12 de Abril de 1999

Abstract

This paper describes the use of the Resource ReSerVation Protocol (RSVP) in combination with Class-Based Queuing (CBQ) as an example of Quality of Service (QoS) resource mangement. The platform used for this study is Linux OS, as it offers an open environment to monitorizing and managing the behaviour of both techniches. The rsvpd 4.2a4 version is described in conjunction with the QoS managment of Linux kernel 2.2.5. Much of the capabilities of CBQ are commented and used in this study. Finally a comparison of resource reservation in Linux against CISCO RSVP with Weighted Fair Queueing (WFQ) implemetation is included.

1. Introducción

El protocolo RSVP[1] se está convirtiendo en un estándar ampliamente aceptado por la comunidad de Internet para el manejo de QoS sobre IP.

RSVP no proporciona los mecanismos necesarios para mantener la calidad de servicio, sino que simplemente se encarga de dar la infraestructura de comunicaciones necesarias y especifica un mecanismo de solicitud, propagación, mantenimiento y eliminación de reservas.

La calidad de servicio para un determinado flujo de datos se implementa mediante mecanismos colectivamente denominados *traffic control* (control de tráfico). La mayoría de las implementaciones de control de tráfico actuales se encuentran en versiones beta de los sistemas operativos instalados en los routers. Estas versiones normalmente no proporcionan un buen soporte para el control de QoS con RSVP, e incluso algunas tan sólo soportan la señalización pero no mecanismos detallados para su funcionamiento. Incluso en el mejor de los casos carecen de los instrumentos necesarios para una monitorización adecuada por lo que los datos deben ser recabados mediante la experiencia. Estas experiencias suelen consistir en la saturación de algún interfaz del router con varios flujos para comparar la salida de ellos tanto con reservas como sin ellas. Dado el carácter cerrado de estas implementaciones comerciales, el router se comporta como una caja negra, y la relación existente entre RSVP y el control de tráfico aparece oculta al usuario.

El sistema operativo Linux ofrece una implementación abierta de control de tráfico que se integra a la perfección con RSVP. Linux ofrece una plataforma que va adoptando continuamente

las innovaciones tecnológicas introducidas en este campo. Ofrece también las herramientas necesarias para monitorizar el comportamiento del control de tráfico y además, nos ofrece la posibilidad de inspeccionar el código que realiza dicho control, junto con los interfaces de comunicación con protocolos de reserva como RSVP.

Este documento muestra el funcionamiento conjunto de RSVP con el control de tráfico implementado en Linux, en particular con el mecanismo de capa de enlace denominado Class-Based Queuing, y muestra detalladamente los pasos seguidos desde que un host solicita una determinada calidad de servicio mediante RSVP hasta que esa QoS es proporcionada, observando el estado del router en cada uno de ellos.

2. RSVP y CBQ.

En este apartado se describe de una forma genérica el funcionamiento tanto de CBQ como de RSVP, y como ambos pueden ser usados en conjunción para cumplir las especificaciones del IETF *Integrated Services* [2]. Para el caso podremos considerar que todas las reservas son *fixed filter* a direcciones unicast, el resto de opciones se extraen lógicamente de las conclusiones aquí descritas.

2.1. RSVP.

RSVP define un modelo de asignación de QoS según el cual cada receptor es responsable de elegir su propio nivel de reserva de recursos, iniciando dicha reserva y manteniéndola activa tanto tiempo como lo desee. Es un protocolo de control Internet, cuya misión consiste en establecer y mantener las reservas de recursos en un árbol de distribución, con independencia de cómo se hayan creado.

Para tomar las decisiones de calidad de servicio asociadas a los paquetes de una aplicación, RSVP

interactúa con cuatro entidades: el módulo de control de admisión (*admission control*) encargado de determinar si la QoS requerida puede ser concedida en términos de recursos disponibles, el módulo de control de política (*policy control*), encargado de determinar si se tienen los permisos necesarios para conseguir dicha QoS, un clasificador de paquetes (*packet classifier*) y un planificador de paquetes (*packet scheduler*) instaladas en cada host.

El *packet classifier* determina la ruta del paquete y a qué flujo pertenece, el *scheduler* toma las decisiones de envío para alcanzar la QoS deseada. En caso de que el nivel de enlace del host tenga su propia capacidad de gestión de QoS (como ATM), entonces el *packet scheduler* negocia con él para obtener la calidad de servicio solicitada por RSVP, normalmente realizando un mapeado de los servicios siempre que sea posible. En caso contrario, por ejemplo, cuando el host está utilizando una línea alquilada o bien está sobre una LAN, el *scheduler* asigna por sí mismo la capacidad de transmisión de paquetes, entre otros recursos, como tiempos de CPU y *buffers*.

El objeto principal con el que interactúan las aplicaciones es la llamada *flowspec* o especificación de flujo, que describe tanto la corriente de tráfico enviada por el emisor como la solicitada por el receptor. Esta *flowspec* especifica en realidad la QoS deseada, y establece los parámetros en el *packet scheduler* del nodo. Junto a éstos aparecen los *filter spec*, que junto con la especificación de sesión, definen los paquetes de datos que han de recibir la QoS definida por los *flowspec*. Los *filter spec* consisten básicamente en un conjunto de filtros que pueden ser usados para seleccionar arbitrariamente subconjuntos de paquetes en una sesión dada, utilizando para ello cualquier campo de las cabeceras de protocolo.

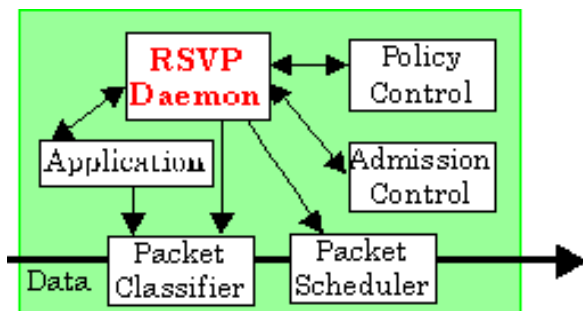


Fig 1: Funcionamiento de RSVP

2.2. CBQ.

Class-Based Queuing (CBQ) [5] es un mecanismo de control de tráfico basado en la compartición del ancho de banda de un enlace para proporcionar un determinado mantenimiento de recursos. Como característica principal, CBQ introduce la noción

de compartición de enlace, permitiendo que múltiples agentes, familias de protocolos o tipos de tráfico compartan el ancho de banda de un determinado enlace. Además, y con el mismo conjunto de mecanismos, CBQ implementa las capacidades de control de tráfico requeridas para implementar los modelos de Servicios Integrados del IETF.

CBQ tiene una estructura jerárquica asociada con un determinado enlace de la red, donde cada *clase* en la estructura corresponde a algún tipo de agregación de tráfico. Esta estructura de compartición de enlace especifica la política deseada en términos de división de ancho de banda para un determinado enlace durante periodos de congestión. Dada una división de un enlace en clases de tráfico se le asocia un determinado ancho de banda a cada una de las divisiones. Esta 'reserva' puede ser estática (asignada por el administrador) o dinámica (con el uso de RSVP). El objetivo principal de CBQ es que una clase con mucha demanda reciba al menos todo el ancho de banda que se le asoció incluso en situaciones de congestión. Esto puede significar que una clase puede ver su ancho de banda limitado al que se le reservó si hay congestión. Si una de las clases no tiene ancho de banda reservado entonces CBQ no garantiza ningún ancho de banda para esa clase durante congestión, y el que se le asocia depende de los mecanismos de planificación de paquetes del router.

Otro concepto clave en CBQ es el préstamo de ancho de banda. Si una clase no está usando todo el ancho de banda asignado, la distribución del exceso de ancho de banda no es arbitrario, sino que sigue ciertas reglas. En realidad, en una estructura compleja de CBQ, todos los paquetes que pasan por el router se asignan a alguna de las hojas del árbol, mientras que los nodos intermedios no 'transmiten', sino que sólo dan una cierta indicación de cómo el ancho de banda sobrante será asociado. En la Fig. 2, si el flujo asociado a la clase D estuviese superando el ancho de banda que le corresponde, el sobrante pasaría a la clase E mientras que el padre A tuviese ancho de banda de sobra. Llegado el caso, el ancho de banda sobrante de A se repartirá atendiendo a los porcentajes de las clases B y C.

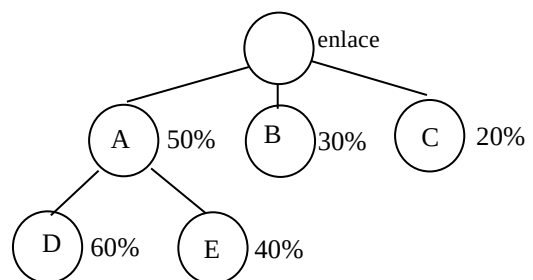


Fig 2: Ejemplo de configuración CBQ

Cada una de las clases definidas tiene asociada su propia cola en el router. Para cada clase se definen dos *packet schedulers*, uno denominado *genérico* y otro de *compartición de enlace*. El primero se encarga de transmitir los paquetes cuando el ancho de banda de la clase no es superado. Es decir, se utilizará un planificador adecuado para que se cumpla el ancho de banda requerido en ese momento por el flujo (en caso de tráfico en tiempo real, que sea capaz de enrutar paquetes sin superar el retardo máximo). El planificador de compartición de enlace es el encargado de transmitir los paquetes cuando se supera el ancho de banda. Éste debe tener las mismas capacidades que el general, pero además debe ser capaz de limitar el ancho de banda del flujo para que se acomode al asociado a su clase, e incorporar alguna estrategia de descarte de paquetes por si fuera preciso. Notar que no necesariamente deben existir dos planificadores, ya que puede haber uno sólo que cambie su estado en el momento que una clase supera su ancho de banda.

Los planificadores de paquetes usan los conceptos de *selector*, que decide en cada instante qué clase tiene permiso para enviar el próximo paquete (p. ej. empezar por la clase de mayor prioridad y hacer un round-robin en las clases con la misma prioridad – por eso se dice que CBQ es un *scheduler* del tipo *Weighted Round Robin -WRR*), y un *delayer* (retrasador), que calcula el momento en el cual una clase que está superando el ancho de banda reservado podrá enviar el próximo paquete. En este sentido, una clase que esté siendo *retrasada* está viendo limitado su ratio al ancho de banda asociado en la jerarquía.

CBQ necesita además de otros mecanismos para permitir la compartición efectiva: un *clasificador de paquetes* y un *estimador del ancho de banda*. El clasificador se encarga de clasificar los paquetes que le llegan al router a la clase apropiada del enlace de salida. En este aspecto CBQ no especifica ningún requisito especial al clasificador más que su existencia. El estimador se encarga de estimar el ancho de banda usado por cada clase en un determinado intervalo de tiempo, para determinar si una clase está recibiendo o no el ancho de banda asociado. El intervalo de tiempo fijado es un parámetro fundamental que determina la precisión con la que el router hará cumplir las asociaciones de compartición del enlace.

Además de estos comportamientos puede definirse que una clase sea de tipo *exempt* (exenta), cuando no se desea ninguna restricción de ancho de banda, *bounded* (delimitada) cuando no se le permite a esa clase coger algo del ancho de banda de sus ancestros o *isolated* (aislada), cuando no permite que sus descendientes cojan algo de su ancho de banda prestado.

2.3. Uso de RSVP y CBQ con Integrated Services.

CBQ y RSVP son ortogonales. CBQ es una implementación de un control de tráfico que especifica la manera en que un router manejará los paquetes que le van llegando. Por definición CBQ no necesita RSVP o cualquier otro protocolo de reserva de recursos. Sin embargo, la potencia y versatilidad que ofrece CBQ lo hace idóneo para actuar en conjunción con este protocolo de reserva. Los flujos podrían solicitar reservas dinámicamente para obtener una mayor prioridad en la jerarquía CBQ. Teniendo en cuenta que RSVP tiene como objetivo proporcionar los Servicios Integrados del IETF, una posible implantación conjunta de ambos podría ser la siguiente:

Controlled Load: Se podría tener una sola clase de prioridad alta para tráfico Controlled Load que use el protocolo RSVP como reserva. Los routers utilizarían protocolos de control de admisión para esa clase y multiplexarían estadísticamente el tráfico Controlled Load en dicha clase, usando un encolamiento del tipo FIFO dentro de la clase. [3]

Guaranteed Services: Se podría crear una nueva clase CBQ en el router cada vez que se le concediera a un flujo un servicio garantizado. Todas estas clases serían hijas de una reservada para el tráfico en tiempo real reservado vía RSVP, que quizás compartiera un ancestro superior con otros tráficos en tiempo real de alta prioridad. [4]

Como se puede observar, CBQ dispone de todos los elementos necesarios con los que RSVP interactúa en cada nodo:

El *control de admisión* y de *políticas* es necesario para ambos y debe ser proporcionado por el propio router.

El *clasificador de paquetes* lo proporciona CBQ, asignando los paquetes que llegan a un determinada clase (aquella a la que pertenece el flujo para una determinada reserva). Este clasificador es configurado atendiendo al objeto *filterspec* proporcionado.

El *planificador de paquetes* debe ser capaz de proporcionar el servicio solicitado por una reserva si esta ha sido admitida. Los servicios se determinan de acuerdo a los *flowspec* solicitados. En el caso de *Controlled Load* el servicio se utiliza un *packet scheduler* normal en una clase con un determinado ancho de banda, y todas las peticiones de este servicio se asignan a esta clase. El servicio *Controlled Load* es proporcionado implícitamente por el control de admisión, que negaría las reservas de este servicio si el número de flujos actuales que se multiplexan en esa clase es tal que comenzaría a empobrecer el servicio y por lo tanto no cumplir el objetivo de que ‘parezca’ que en el enlace hay una carga ligera. Todos los flujos en esta clase no tienen que preocuparse en caso de

congestión si usan todo el ancho de banda reservado, ya que todos ellos tienen asegurado el ancho de banda, proporcionado por CBQ. En el caso de *Guaranteed Services* el caso es más simple. Tendremos una clase 'hoja' por cada solicitud de reserva de recursos para este servicio. El ancho de banda queda asegurado por CBQ al tratarse de clases separadas. El *jittering* puede controlarse si la clase padre de todas estas reservas de *Guaranteed Services* tiene una prioridad alta para el *WRR* del CBQ. El propio *packet scheduler* general de esa clase es el encargado de proporcionar el ancho de banda requerido y que se cumplan los parámetros de máximo retraso permitido para esa clase.

Integrated Services no especifica con exactitud qué debe hacerse con los paquetes que superan los servicios reservados. Indica que al menos deben tratarse como paquetes del tipo *best effort*. En CBQ no hay una clase que obtenga un tratamiento peor que ese. Recordemos que las clases con ancho de banda asignado de 0 son tratadas con esta política, y por lo tanto la especificación de los *Integrated Services* también se cumple en este caso. Además, como veremos en el siguiente apartado, la implementación de Linux que se comenta incluye la posibilidad de definir a qué clase irán a parar los paquetes que excedan el ancho de banda solicitado, pudiéndose crear entonces una clase para *best effort* donde irían todos los paquetes sobrantes, aunque es conveniente que los paquetes que sobrepasen la reserva no pierdan demasiada prioridad siempre y cuando queden recursos libres.

3. RSVP y CBQ en Linux.

Linux soporta control de tráfico con RSVP desde Abril de 1998. Todas las implementaciones 'oficiales' desde entonces han sido llevadas a cabo por Alexey Kuznetsov, que es el encargado de realizar los parches de la versión de RSVP de ISI a Linux, incluir las funcionalidades en el kernel y la implementación de las herramientas a nivel de usuario para administrar el control de tráfico (herramientas denominadas *iproute2+tc*) entre otras cosas. Todas estas herramientas y parches pueden encontrarse en [7], mientras que la distribución de RSVP del IETF puede encontrarse en <ftp://ftp.isi.edu/rsvp>. Actualmente la última versión de RSVP de la que se dispone parche para Linux es la 4.2a4, y el soporte para el control de tráfico funciona para versiones del kernel superiores a la 2.1.108 y todas las del 2.2.X. Las comentadas en este documento corresponden a la versión del kernel 2.2.5. Oficial en este caso se refiere que es la que proporciona por defecto el kernel de Linux, ya que existen otras implementaciones de CBQ y RSVP como las de FreeBSD (que usa ALTQ y RSVP 4.1a6), RedHat 5.1 (con RSVP 4.2a3 y sin *traffic control*) y Slackware (con CBQ y RSVP 4.1a6).

3.1. Instalación y configuración previa.

Para hacer funcionar el control de tráfico en Linux es necesario activar una serie de opciones en el kernel. La primera de ellas es los sockets *netlink*. Esto es debido a que la comunicación entre el kernel y el demonio RSVP se realiza mediante estos sockets. Esto activa, además todas las funciones del servicio RT, que requieren el control *Sysioctl* (otra opción del kernel). Posteriormente debe activarse todas las casillas del apartado QoS para que el kernel le de soporte a todas estas funciones.

La compilación de la versión correspondiente de RSVP requiere de una serie de parches que se pueden encontrar en [7].

3.2. Control de tráfico.

El control de tráfico en Linux es bastante completo, y se configura mediante la herramienta *tc* incluida en el paquete *iproute2+tc*. Esta herramienta proporciona cuatro servicios principalmente:

qdisc (*queueing disciplines* – disciplinas de encolamiento): Este objeto es el equivalente al *packet scheduler* comentado a lo largo del documento, aunque se trata de un concepto más amplio. Se asocia a un nodo de la jerarquía y se refiere a cualquier mecanismo de control de tráfico que se pueda implementar. Normalmente un *qdisc* es una 'caja negra', que es capaz de encolar paquetes y desencolarlos (cuando el dispositivo está en condiciones de enviar algo) en el orden y en el momento determinado por el algoritmo escondido en él. Los *qdisc* se dividen en dos categorías: *queues* (colas), que no tienen una estructura visible desde el exterior y *schedulers* (planificadores), que asignan los paquetes a las clases de tráfico usando los clasificadores. Normalmente cada clase tiene algún *qdisc* como hijo (reglas, colas ,etc ...). Todo el trabajo inteligente de control de tráfico se realiza en los módulos *qdisc*.

Los tipos de colas que podemos encontrar son *noop*, que no hace nada más que enviar paquetes cuando puede, *fifo* queue, que es la cola fifo normal que conocemos, y *3-band FIFO queue*, también denominado *fifo_fast_enqueue*, que puede configurarse como un caso de cola FIFO con alta prioridad.

Existen numerosos *schedulers* implementados. Los que nos interesan son los siguientes:

- *CBQ* (*Class-Based Queueing algorithm*): El comentado en el apartado anterior. Usa una versión avanzada del algoritmo *Weighted Round Robin* tal y como se definió el CBQ en su diseño. Dada su estructura jerárquica, este

- qdisc* puede tener otros *qdiscs* asociados a sus nodos hoja.
- *SFQ* (*Stochastic Fairness Queuing algorithm*), que es similar al algoritmo *WFQ*, aunque no utiliza el mecanismo de *time-stamp*, sino que usa un mecanismo de procesamiento de colas en *round-robin*. Esto provoca que no sea apropiado para ser usado en CBQ si se va a trabajar con tráfico en tiempo real. Es debido a que la planificación en *round-robin* y los objetivos de compartición de enlace de CBQ son aparentemente contradictorios. Como su nombre indica no es del todo ‘justo’, ya que en periodos de alta congestión trata todos los flujos como uno solo.

class (clases): Son las clases de tráfico. Se asocian a un interfaz con una cierta prioridad, un handle de la forma X:Y y opcionalmente un *qdisc*, del cual pueden configurarse sus parámetros. Todos los conceptos de clases vistos anteriormente como *bounded*, *isolated*, etc. pueden ser configurados.

filter (filtros): Son los *packet classifiers* comentados en el apartado anterior. Los filtros se asocian a un dispositivo y se les puede asignar una determinada prioridad, el protocolo que van a filtrar, los parámetros del estimador (intervalo y constante de tiempo) y se les indica el handle de la clase a la cual van a ser asignados los paquetes que filtre. Hay cuatro tipos de filtros, por rutas, rsvp/rsvp6, fw (basado en las ipchains) u u32 (Universal 32bit key Packet Classifier). Dependiendo del filtro se pueden asociar políticas, como descartar paquetes o enviar los paquetes que superen el ancho de banda a otras clases.

estimator (estimadores): están implementados como un módulo externo a los *qdisc*. Sus valores de frecuencia y constante de tiempo se pueden especificar para cada clase. De esta manera se puede determinar cada cuánto tiempo queremos que el kernel compruebe el cumplimiento de las restricciones de CBQ a esa clase, y por consiguiente fijar su precisión.

3.2.1. Integración de RSVP con el control de tráfico.

La implementación de RSVP para Linux sólo trata el tráfico de tiempo real. Para el correcto funcionamiento de la reserva de recursos es necesario configurar CBQ en el interfaz por donde se vayan a realizar las reservas. El demonio rsvpd detecta automáticamente cuando eso es así, y en caso contrario cualquier reserva será rechazada con un mensaje *Traffic control error* debido a que no existe el mecanismo subyacente necesario para su gestión.

3.2.2. Configuración de CBQ en el interfaz.

El primer paso es instalar CBQ en la raíz del interfaz indicado. Esto se hace instalando un *qdisc*

e indicándole que será el raíz de una jerarquía de CBQ en ese interfaz (su handle será 1:0). Todas las clases que no tengan su propio *qdisc* serán gestionados por este *scheduler*. Como parámetros de ancho de banda en el raíz debe incluirse el verdadero del interfaz, ya que no es posible limitar el ancho de banda del interfaz indicándole un valor menor al real. Esto es así debido a que para el kernel el ancho de banda del interfaz es desconocido e irrelevante, y trata de la misma manera un interfaz de 10Mbits o uno de 100Mbit. El valor indicado en la configuración es importante porque los estimadores utilizarán este valor para estimar los ratios de transmisión y los porcentajes que deben compartirse entre los distintos nodos del árbol. También se pueden indicar parámetros que ayudarán al kernel en su gestión como es el tamaño medio de los paquetes que recibirá así como cual será el tamaño mínimo.

El siguiente paso es instalar una clase raíz. El identificador de esta clase será 1:1. Esta clase se configura con el mismo ancho de banda y ratio que da el propio interfaz, y será la clase padre del resto de clases que podrán coger algo de ancho de banda prestado. No hay que olvidar configurar los parámetros del estimador en esta ni en ninguna otra clase para que los resultados sean precisos.

Debajo de esta clase lo normal es configurar la estructura CBQ deseada en función de las necesidades de cada institución para ese enlace. Para el funcionamiento de RSVP debe configurarse una clase que contenga todo el tráfico de tiempo real que vaya a ser configurado por RSVP. Esa clase debe tener el handle *:7FFE. RSVP busca la clase con ese identificador y asocia las reservas a esa clase. El ratio que se le debe asignar a esta clase es la máxima que se le desee asignar al tráfico en tiempo real.

La clase *:7FFF es también una clase especial, donde cae todo el tráfico en tiempo real que excede el ancho de banda reservado. Es necesario por tanto configurar esta clase adecuadamente.

El resultado es el que se puede ver en la Fig. 3. Es la única configuración ‘manual’ necesaria para hacer funcionar RSVP con CBQ. En las otras clases se puede configurar el tráfico ftp, telnet o de correo, asignando diferentes *qdisc* a cada una y diferente ancho de banda, asignando clasificadores por rutas, protocolos, etc ...

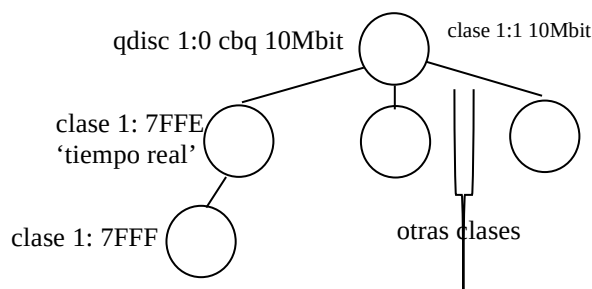


Fig 3: Configuración base de CBQ para trabajar con RSVP

3.2.3. Interacción entre RSVP y CBQ al realizar reservas.

Dada la configuración CBQ de la figura al arrancar el demonio de RSVP, éste configura automáticamente dos filtros, uno con el protocolo ip que filtra todos los paquetes rsvp y otro de protocolo ipv6 si existiese y rsvpd hubiese sido compilado con es opción. Así manera el kernel le pasa al demonio los paquetes del protocolo RSVP.

Una vez realizada la reserva lo que hace el demonio es, tras pasar el control de admisión si existiese (en este caso que exista ancho de banda para ser asignado) crear una nueva clase hija de *:7FFE que se denomina *:8001 (la siguiente se llamará *:8002, etc ...) con los parámetros del *flowspec* solicitado. Por otro lado instala en la raíz de CBQ un nuevo filtro que filtra los paquetes correspondientes a la sesión que se haya configurado en la reserva (normalmente dirección-puerto origen, dirección-puerto destino) que asocia los paquetes que la cumplan a la nueva clase creada. La configuración tras una reserva sería la que muestra la Fig. 4.

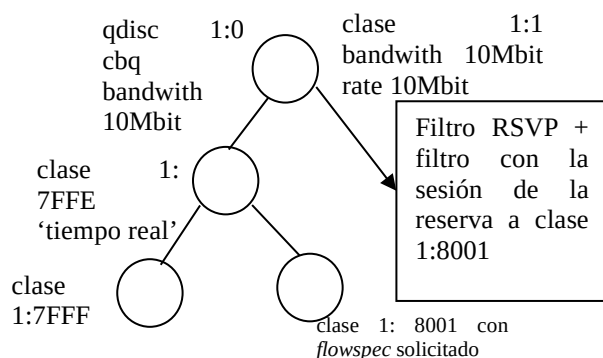


Fig. 4: Estado de CBQ en el enlace tras una reserva RSVP

Una vez realizada la reserva e instaladas las nuevas clases y los filtros, CBQ es el encargado de que se cumplan los servicios solicitados, atendiendo a las normas comentadas en el apartado 2.

Los *Integrated Services* se gestionan tal y como Floyd sugiere en [5]. Para *Controlled Load* sólo se utiliza una clase, que aumenta su ancho de banda a medida que se van recibiendo solicitudes de reserva de este tipo de servicio. Por cada solicitud

se instala un filtro que asigna los paquetes a esa clase y los que sobrepasan el ancho de banda (de la reserva en este caso, no de la clase ya que todos los flujos *Controlled Load* comparten la misma clase) es llevado a *best effort*. Esta política no se instala en la propia clase, sino en los filtros. Éstos son los encargados de comprobar si los paquetes cumplen o no las reservas.

En el caso de *Guaranteed Services* se crea una clase por cada reserva que contiene el ancho de banda solicitado, y un filtro que asocia los paquetes de la sesión a esa clase. En este caso la política para los paquetes que superan el ancho de banda surge directamente de superar el ancho de banda de la clase (sólo hay una clase por reserva).

4. Pruebas realizadas.

Se han realizado dos pruebas para observar el comportamiento de la reserva de recursos con CBQ y RSVP en un router Linux. La primera muestra el comportamiento de la implementación del control de tráfico en un router CISCO, la segunda muestra el control de tráfico en una máquina Linux.

4.1. Prueba con router CISCO 4500.

La configuración para esta prueba se muestra en la Fig. 5. Todas las redes son Ethernet a 10Mbits. Se pretendía realizar una reserva desde labredes16 en la subred 94 hasta odisea en la subred 95, ambas máquinas PC-linux. Para introducir ruido se utilizaron las máquinas caprichosa (PC-Linux) y aries (Sun-Solaris) introduciendo tráfico hacia labredes11 (O2). Todas las emisiones se realizaron con el programa de libre distribución mgen. Se necesita utilizar tres redes para poder saturar el interfaz de salida del router. Por otro lado la máquina que realiza la reserva está sola en una subred para evitar las colisiones y tener la seguridad de que consigue más de 5Mbits que es lo que le reservaría el router con WFQ.

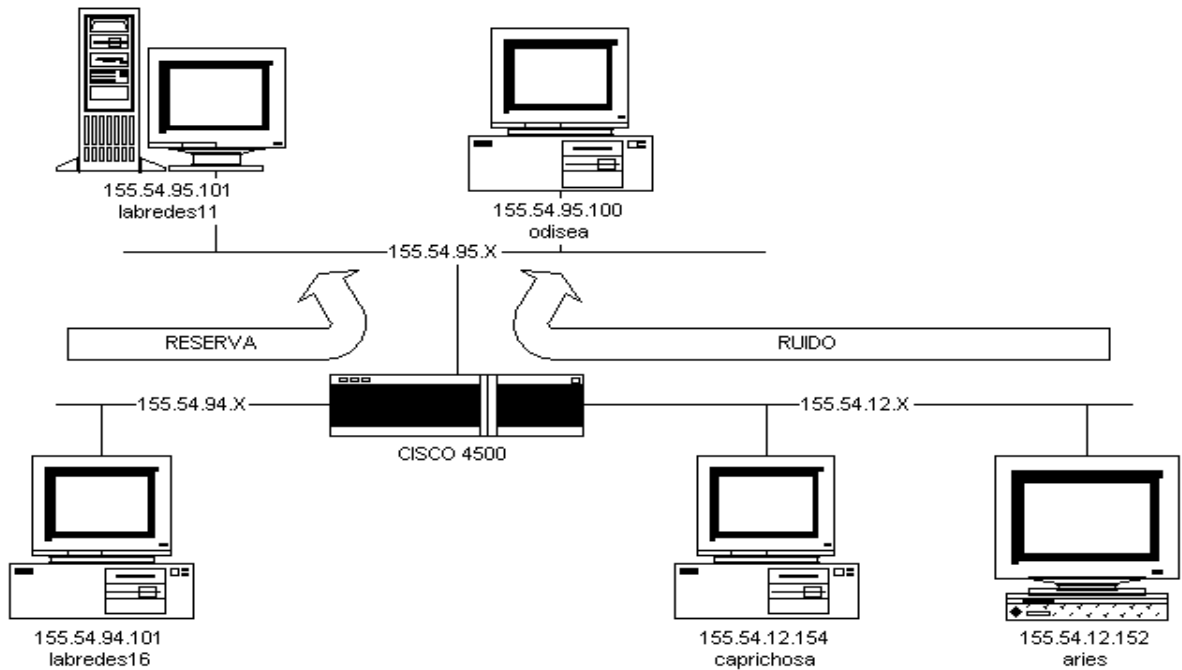


Fig. 5: Configuración de las pruebas con el router CISCO

Los resultados se muestran en la Fig. 6. Hay tres flujos: el de la reserva (rsvp) desde labredes16 a odisea, y los otros dos sin reserva desde aries y caprichosa hacia labredes11. Las emisiones comienzan sin ninguna reserva instalada. El router aplica WFQ y le da 5Mbps a cada interfaz. Como los dos flujos que vienen por la red 12 tienen que compartirlo, como media se le asignan unos 2.5Mbps. Los picos que se observan son debidos a que el receptor de los dos flujos nrsvp es la misma máquina, los dos procesos recogen sus

paquetes a ráfagas y por turnos, no simultáneamente.

En el punto 76830 se realizó una reserva dinámicamente de 7.5Mbps. El flujo de la reserva aumenta hasta algo más de 8Mbps. Esto es debido a que el resto de ancho de banda del interfaz de salida (unos 9.2Mbps, que es lo que consigue el router) es también repartido entre los dos interfaces de entrada con WFQ, por lo que a rsvp le queda 7.5Mbps + 0.8Mbps. En el instante 76915 se redujo la reserva a 5Mbps y finalmente se eliminó.

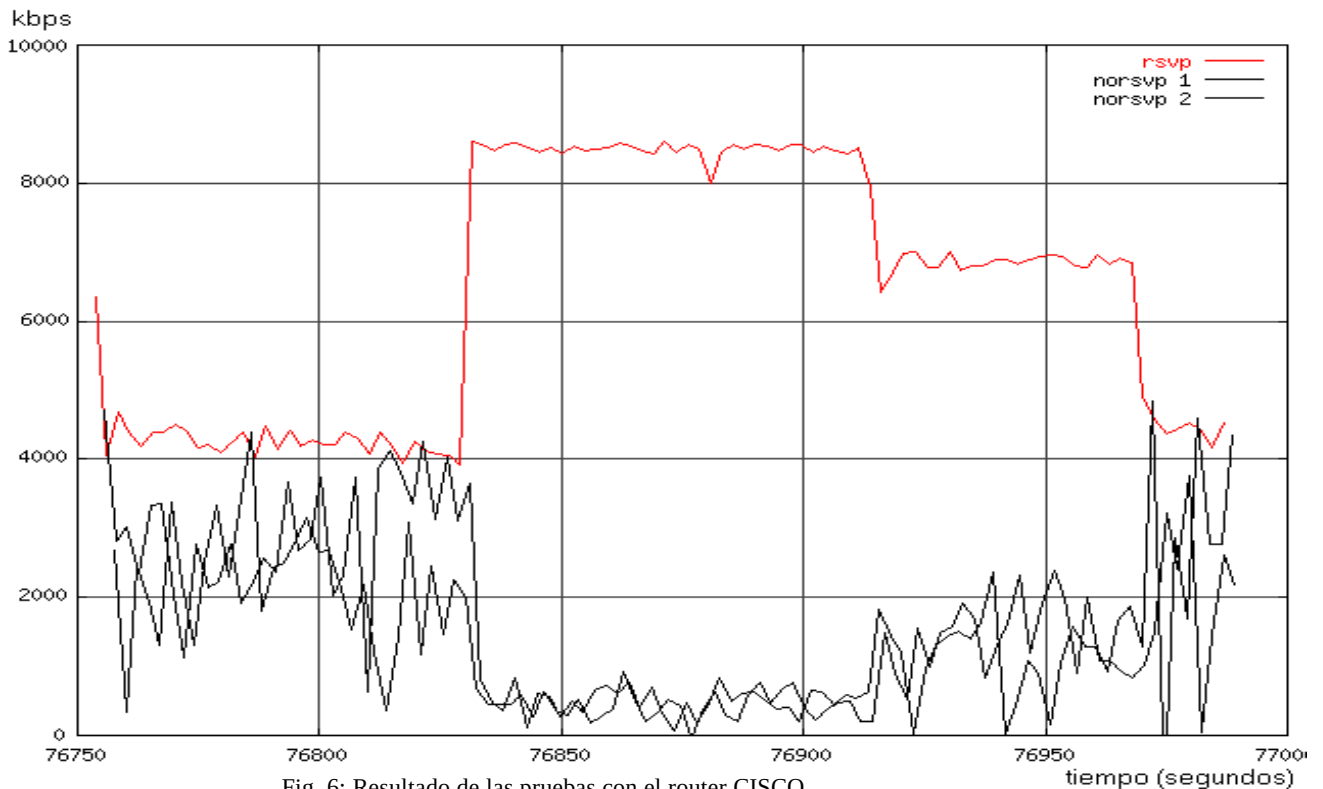


Fig. 6: Resultado de las pruebas con el router CISCO

4.2. Prueba con Linux.

La Fig. 8 muestra la configuración. Esta prueba se realizó en un entorno totalmente aislado entre tres equipos con Linux y kernel 2.2.5. El router tenía dos interfaces, uno de 10Mbps y otro de 100Mbps. Los dos flujos de entrada procedían de la red 93 con 100Mbps para poder saturar la salida de

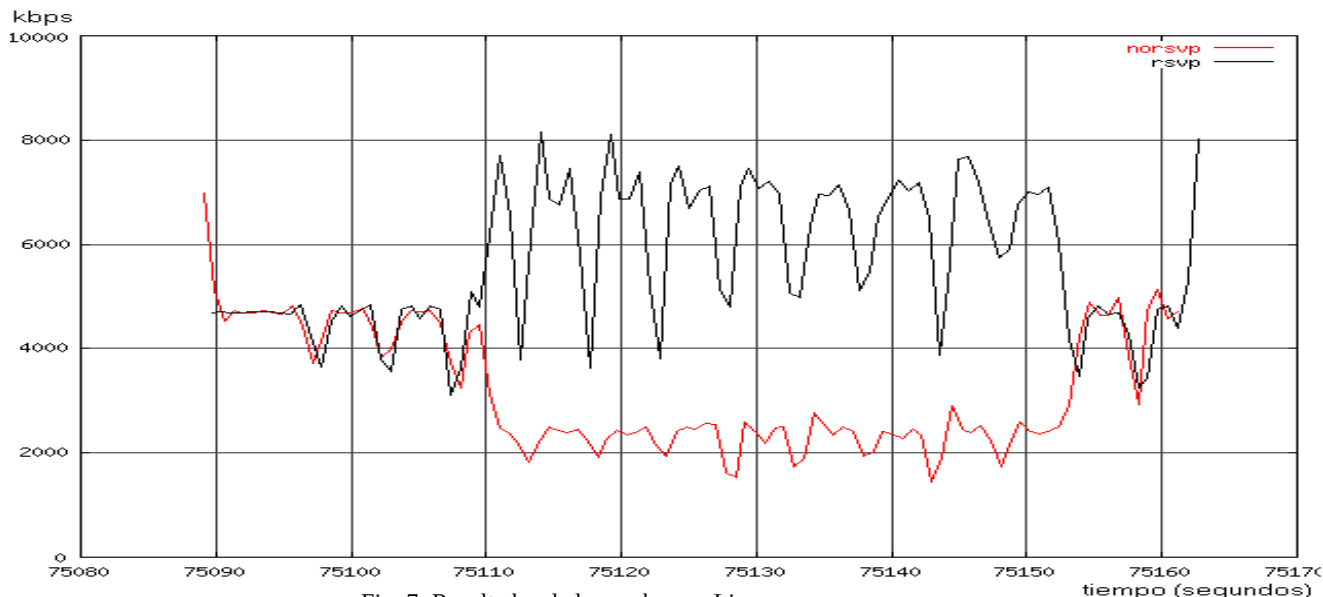


Fig. 7: Resultados de la prueba con Linux

10Mbps. Se ejecutaron dos procesos que emitían cada uno a 10Mbps. En la Fig. 7 se puede observar el resultado. Inicialmente ambos procesos emitían sin reserva. En el instante 75110 se solicita una reserva de 7Mbps que se elimina en el instante 75153. Como se observa, el flujo sin reserva ve limitado su ancho de banda a favor del reservado.

Las fluctuaciones del ancho de banda se deben a que el receptor de los dos flujos es un solo equipo, recogiendo cada proceso los paquetes por turnos.

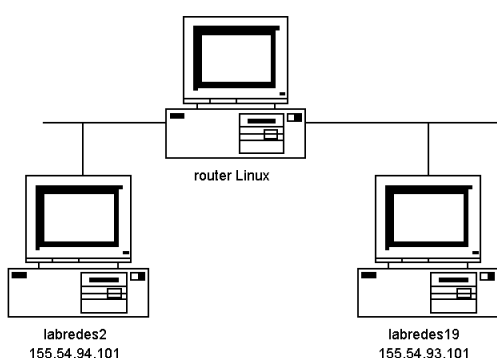


Fig. 8: Configuración de la prueba con Linux

5. Conclusiones y vías futuras.

En este estudio se ha mostrado la utilidad de Linux en un entorno distribuido para el desarrollo de routers con RSVP. También se ha mostrado cómo se usa CBQ para implementar de una manera eficaz los Servicios Integrados del IETF y cómo se configura dinámicamente a través de RSVP. Se ha

visto el funcionamiento de dicha administración en Linux y se ha comparado con una implementación comercial en un router CISCO modelo 4500.

Como futuras vías de investigación se pretende observar el comportamiento de las reservas multicast e IPv6, realizando estudios similares.

6. Agradecimientos.

Nuestro mas sincero agradecimiento a todas las personas que han colaborado en este estudio. En primer lugar a Alexey Kuznetsov, por haber desarrollado el control de tráfico para Linux, por su paciencia, por habernos dedicado horas de trabajo y aclaraciones, y por habernos enviado parches de los problemas que hemos encontrado prácticamente en 'tiempo real'.

En particular a David Larrabeiti de la Universidad Carlos III de Madrid, por ayudarnos con las primeras implementaciones de RSVP para Linux, y en proporcionarnos los parches que él mismo había diseñado. A Xavier Martínez de la Universitat Politècnica de Catalunya, por ayudarnos a reproducir las pruebas de RSVP que ellos habían realizado con su router de CISCO y ayudarnos con la configuración del WFQ en el router.

Por último agradecer al CICYT por haber soportado parcialmente este trabajo a través del proyecto TEL97-2001-E.

7. Referencias.

- [1] B. Braden, B., Ed., et. Al., "Resource Reservation Protocol (RSVP - Ver. 1 functional Specification)", RFC 2205, Sep. 1997.
- [2] J. Wroclawski, "The Use of RSVP with IETF Integrated Services", RFC 2210, Sep. 1997.

- [3] J. Wroclawski, “*Specification of the Controlled Load Quality of Service*” RFC 2211, Sep. 1997.
- [4] S. Shenker, C. Patridge, C., y R. Guerin, “*Specification of Guaranteed Quality of Service*”, RFC 2212, Sep. 1997.
- [5] S. Floyd y V. Jacobson, “*Link Sharing and Resource Management Models for Packet Networks*”, IEEE/ACM Transaction on Networking, Vol. 3, No. 4, pp 365-386, Agosto 1995. URL: <ftp://ftp.ee.lbl.gov/papers/link.ps.Z>.
- [6] S. Floyd, página WWW de CBQ, URL <http://www-nrg.ee.lbl.gov/floyd/cbq>.
- [7] A. Kuznetsov, Sitio FTP de los fuentes de RSVP y del paquete de iproute2+tc, URL <ftp://ftp.inr.ac.ru/ip-routing>.
- [8] J. Crowcroft, M. Handley, I. Wakeman, “*Internetworking Multimedia*” UCL Press, Nov. 1998

Capítulo 1B: Sistemas Móviles

Protocolo de handover y sincronización para comunicaciones móviles multimedia.
V. Martínez Barres, V. Casares Giner (U. Politécnica de Valencia)

Red ATM Experimental para Comunicaciones Móviles Multimedia.
J. Martínez, J.R. Vidal y L. Guijarro (U. Politécnica de Valencia)

Macro-Diversidad con Combinación Pos-Decodificación.
M. Gorricho, J. Paradells (U. Politécnica de Cataluña)

Análisis de Throughput y Retardo en un Sistema CDMA Celular de Paquetes con Control de Potencia Imperfecto.
J. Romero Jerez, M. Ruiz García, A. Díaz Estrella (U. Málaga)

Protocolos de Control de Acceso al Medio para Tráfico Multimedia en Redes de comunicaciones Móviles.
L. Carrasco, G. Femenias, I.F. Akyildiz, J. McNair Broadband (U. Islas Baleares)

Análisis de controles de admisión en un sistema CDMA para tráfico heterogéneo.
R. Pérez Sánchez, M. Ruíz García (U. Málaga)

Protocolo de handover y sincronización para comunicaciones móviles multimedia

Vicente Martínez Barres, Vicente Casares Giner
Departamento de Comunicaciones Universidad Politécnica de Valencia
Camino de Vera, s/n 46022 Valencia
Email: vimarba0@teleco.upv.es

Abstract

The increase in the demand of mobile communications and multimedia applications has created an enormous interest in the extension of the ATM concept to wireless access networks. The present paper studies the handover and synchronization problem in mobile multimedia communications. We describe different methods proposed in the literature to solve the access and fixed networks handover problem. We present a mobile communications system implementation whose objective is to evaluate the impact of handovers in an ongoing multimedia communication.

1 Introducción

El incremento de la demanda en el campo de las comunicaciones móviles lleva a investigadores y fabricantes a dirigir sus esfuerzos al desarrollo de nuevas tecnologías en este sector que permitan aumentar la cantidad y calidad de los servicios ofrecidos. Junto con las comunicaciones móviles el sector de la multimedia también está experimentando un elevado auge, por este motivo se está dedicando gran esfuerzo a compatibilizar ambos sectores y poder ofrecer al público productos de calidad que reúnan ambos aspectos. El tráfico multimedia se caracteriza por su heterogeneidad, sus elevados requisitos de ancho de banda y sus requisitos de tiempo real. Una de las tecnologías que mejor se adaptan a esta definición es *ATM*, por ello se ha pensado en ella para soportar estos nuevos productos. Ésta necesitará de nuevas aportaciones que solucionen la problemática intrínseca de las redes móviles: localización, *handover*, acceso al medio radio, etc. A esta nueva tecnología resultante se la ha denominado *wireless-ATM*.

Este documento forma parte de un proyecto mayor que culminó con el desarrollo de una red experimental *ATM* para comunicaciones móviles multimedia¹, en [9] se describe el marco del proyecto, sus objetivos, las diferentes partes de la red experimental y las relaciones entre ellas. El presente documento se centra en la problemática del *handover*, tanto a nivel radio como a nivel de red fija. Se describirán las diferentes opciones que aparecen en la literatura y se presentarán las soluciones adoptadas en la red experimental, junto con una evaluación de las mismas.

2 Handover. Definición y necesidad

Aunque se podría disponer de una red móvil formada por una sola estación base² (EB) que diera servi-

cio a todos sus posibles usuarios, ésta no pasaría de ser una red privada con un número muy limitado de abonados. Para lograr implantar redes móviles en amplios territorios y capaces de dar servicio a un gran número de abonados es necesario dividirlo en partes más pequeñas, llamadas células, principalmente por dos motivos: (i) potencias de transmisión; y, (ii) reutilización de canales. El recurso del espectro es uno de los más limitantes, fraccionando el territorio en células, varias pueden reutilizar la misma zona del espectro si el nivel de interferencia es lo suficientemente bajo. Esta división del territorio genera un problema cuando un usuario, que mantiene una comunicación en curso, se desplaza de una célula a la vecina. Deberá existir algún mecanismo automático que le provea de un canal radio en la nueva célula y traslade su comunicación en curso hacia el nuevo radiocanal. Este proceso se denomina *HandOver (HO)*.

3 Detección y ejecución del handover. Nivel radio

Para iniciar un *HO* deben considerarse dos cuestiones: (i) cómo se detecta la necesidad de *HO*; (ii) quién inicia el proceso de *HO*.

La ejecución de un *HO* es cara, por lo que su realización innecesaria debe ser minimizada. Un criterio demasiado laxo puede provocar que se realicen demasiados *HOs* innecesarios, con el consiguiente aumento de la carga de señalización. Un criterio demasiado estricto puede provocar la pérdida de la llamada antes de que se solicite o termine la realización del *HO*.

3.1 Detección del handover

A continuación se enumeran algunos de los algoritmos que se utilizan para decidir la realización del *HO*:

Usemos como referencia la figura 1 donde el terminal móvil (TM) se mueve desde la EB 1, EB antigua, hacia la EB 2, EB candidata o nueva. El nivel medio de la señal recibida por el TM desde la EB 1 dismi-

¹ Este trabajo ha sido financiado por el Ministerio de Educación y Ciencia a través de la "Comisión Interministerial de Ciencia y Tecnología (CICYT)" en los proyectos TIC96-0680 y TIC98-0495-C02-02.

² Se entenderá por EB al conjunto formado por un transceptor radio más su soporte de operación e inteligencia asociada. Así, en un entorno de comunicaciones móviles de segunda generación una EB estaría formada por un transceptor radio más la parte de BSC que se encarga de su control. En un entorno *wireless-ATM* estaría formada por un transceptor radio o radiopuerto más la

parte de access point (AP) y/o EMAS-E [9] que se encarga de su control.

nuye a medida que se aleja de la misma, aumentando el nivel medio de la señal que se recibe de la EB 2.

- Potencia de la señal recibida. La decisión del *HO* se fundamenta en la mayor potencia promedia recibida. En la figura 1, el *HO* ocurriría en la posición A.
- Potencia de la señal recibida con umbral. El *HO* únicamente se lleva a cabo si la señal actual es suficientemente débil (por debajo de un umbral) y la otra es la mayor de las dos. Si el umbral es T2, el *HO* se ejecutará en el punto B.
- Potencia de la señal recibida con histéresis. El *HO* únicamente se ejecuta si la nueva señal está por encima de la actual en un margen de histéresis, h en la figura 1. En este caso, el *HO* ocurrirá en el punto C. Esta técnica prevé del llamado efecto “ping-pong”.
- Potencia de la señal recibida con histéresis y umbral. La ejecución del *HO* se lleva a cabo únicamente si la señal actual cae por debajo de un umbral y la posible candidata está por encima de la actual en un margen de histéresis. En la figura 1, el *HO* se ejecutará en el punto C si el umbral es T1 y T2, y ocurrirá en el punto D si el umbral es T3.

3.2 Métodos de conmutación del radiocanal

Existen dos procedimientos para la conmutación del canal radio [1] entre la EB antigua y la EB candidata:

- *Hard-handover*. El móvil únicamente puede recibir tráfico de usuario de una EB al mismo tiempo. Para establecer el nuevo radiocanal debe primero liberar el antiguo. Se produce normalmente un pequeño hueco en la conversación durante el traspaso del enlace.
- *Soft-handover*. El móvil puede recibir señales de ambas EEBB simultáneamente y ambas EEBB pueden recibir al móvil. La red debe combinar las señales de la dos de alguna forma.

3.3 Ejecución y control del handover

Tres son los métodos que se manejan para la realización del *HO* [1]:

(i) *MCHO* (“*Mobile-Controlled Handover*”). En este método el TM monitoriza constantemente la potencia

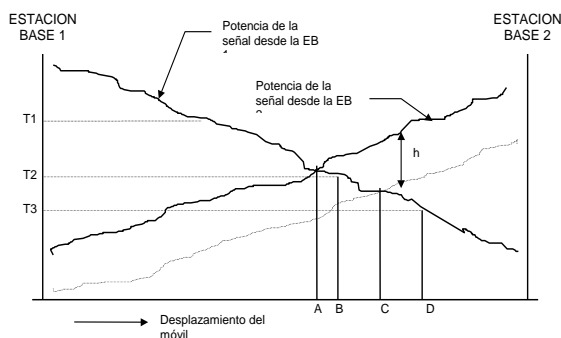


Figura 1: Evolución de la potencia recibida por el móvil. Umbrales de decisión.

cia de la señal recibida de su EB y de varias candidatas para el *HO*. Cuando se cumple algún criterio, el móvil busca el mejor candidato y envía una petición de *HO*. Se utilizada en los sistemas *DECT* y *PACS*.

(ii) *NCHO* (“*Network-Controlled Handover*”). En este método la EB monitoriza la potencia y calidad de la señal del TM y cuando ésta se deteriora por debajo de un umbral la red inicia el *HO* con otra EB. Se emplea en los sistemas *CT-2 Plus* y *AMPS*.

(iii) *MAHO* (“*Mobile-Assisted Handover*”). Es una variante de *NCHO*, en la que la red pregunta al móvil acerca de la medida que obtiene de la señal recibida de las EB que tiene alrededor. Esta información la retorna a la red y es ésta la que decide si es necesario el *HO* y con qué EB candidata. Esta estrategia de *HO* es empleada en *GSM*.

3.4 Métodos de dialogo durante el handover

Otra forma para clasificar los diferentes métodos de *HO* se basa en observar con qué EB se realizada principalmente el dialogo de control durante el mismo:

- *Back Handover*. Se identifica como “*backward handover*” aquel cuyo dialogo entre TM y red fija se efectúa mayoritariamente vía la EB antigua.
- *Forward Handover*. Se identifica como “*forward handover*” aquel cuyo dialogo entre TM y red fija se efectúa mayoritariamente vía la EB candidata o nueva.

Si el *HO* se efectúa debido a que el enlace existente se deteriora, el éxito del *back handover* no queda garantizado. En el *backward handover*, el TM puede informar a la EB antigua sobre un colectivo de posibles candidatos (más de uno) como EB nueva, de esta manera, la EB antigua puede verificar y coordinar, de entre todas las posibles, que EB ofrece mejores posibilidades; sin embargo, con el *forward handover* el TM ya toma partido por una.

4 Handover de red fija

En la figura 2 se muestra el concepto de *HO* de red. El móvil sale de una célula cuya EB recibe la información de usuario que entrega al móvil de un nodo de red al que está conectada. Al traspasar el control a la EB vecina ésta debe recibir la información proveniente de la red fija para entregarla al móvil. Es necesario que la red provea algún mecanismo para realizar un re-enrutamiento de la conexión en curso. Éstos pueden ser muy diversos en cuanto a su planteamiento y muy diferentes desde el punto de vista de la eficiencia.

Asociada al *HO* aparece la posible pérdida de la secuencia en las unidades de datos de protocolo (*PDU*s) que se entregan al móvil. Dado que el *HO* provoca un cambio en la ruta de la información de usuario manteniendo el origen y el destino, durante este cambio se pueden perder o solapar paquetes. Será necesario proveer ciertos mecanismos para

eliminar o minimizar este problema según el caso. Los llamaremos, mecanismos de sincronización.

Podemos clasificar los métodos de *HO* de red en cuatro tipos [2]: (i) establecimiento total; (ii) reencaminamiento incremental; (iii) extensión de la conexión; (iv) establecimiento *multicast*. A continuación se exponen ejemplos concretos de cada uno de estos tipos.

4.1 Establecimiento total

El establecimiento total de la conexión requiere la creación de una nueva conexión entre puntos terminales. Es el método que supone menor impacto en la red fija ya que no precisa añadir funcionalidad complementaria a los conmutadores. Sin embargo, involucra al terminal de red fija en el *HO* y su lentitud de ejecución puede provocar que se llegue al cierre abrupto de la conexión.

4.2 Reenrutamiento en el nodo común más cercano

El método *NCNR* (“*Nearest Common Node Rerouting*”) [3] se encuadra dentro del grupo de reencaminamiento incremental y su principal característica es la búsqueda dinámica del nodo desde el cual se alcanzará la nueva EB a la que debe conectarse el TM.

Si suponemos establecida una conexión entre un terminal fijo (origen) y un TM, se define *NCN* (“*Nearest Common Node*”) como el primer nodo común en las rutas óptimas: origen – EB antigua y origen – EB candidata, empezando por las EEBB. Al usar este nodo para re-enrutar la conexión en curso, *NCNR* minimiza los recursos utilizados por la red al realizar los sucesivos *HOs* en los que se puede ver involucrada una conexión. La búsqueda del *NCN* se realiza mediante un algoritmo [3] que es válido para cualquier topología de red y por lo tanto, es independiente de ésta. Cuando se encuentra el *NCN*, éste es el encargado de establecer la parte del circuito virtual (*VC*) que falta entre éste y la EB candidata especificando la calidad de servicio (*QoS*) de la conexión. Cuando todo el establecimiento está completo se informa al móvil y éste comienza a realizar el *HO* radio. El *NCN* se encarga de replicar por ambas ramas las células que le llegan desde el origen, esta situación multipunto se mantiene hasta que se estabiliza el nuevo canal radio. La realización del *HO* de red por este método distingue según se realice sobre una conexión que soporta tráfico sensible al tiempo o tráfico insensible al tiempo. Se pretende con esto minimizar el *jitter* en el primer caso y minimizar las células perdidas en el segundo caso [3].

4.3 Árbol virtual de conexiones

El método *VCT* (“*Virtual connection tree*”) [4] se encuadra dentro de los denominados de establecimiento *multicast*. En el establecimiento de la conexión, se crea de forma estática un grupo de *VCs* punto a punto que van desde el nodo raíz de la conexión hasta cada una de las EEBB del área de cobertura. De esta forma, cuando un TM se mueve de un punto de acceso a otro, no es necesario establecer

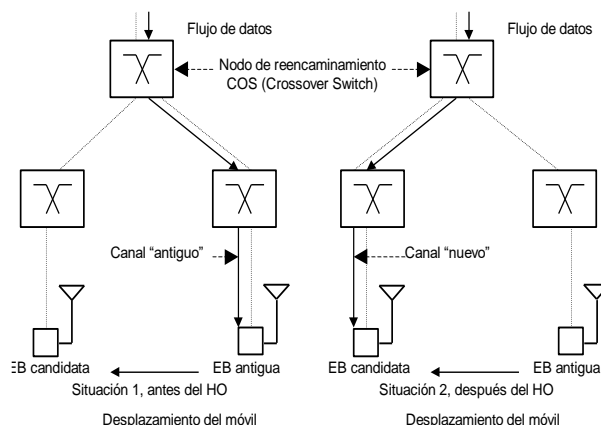


Figura 2: Handover de red.

nuevas conexiones en la parte de red fija. El reencaminamiento en la red fija es realizado simplemente utilizando otra rama del árbol virtual. Al cambiar de EB, el TM cambiará de *VC* y transmitirá por el que llega a la nueva EB. El conmutador al que se conecta la raíz del árbol detectará que el móvil ha realizado un *HO* porque las células le llegan por otro *VC* del árbol virtual, cuando detecte este hecho, encaminará las células del sentido descendente por el nuevo *VC*. Este planteamiento tiene varios inconvenientes: (i) provoca un consumo elevado de recursos [5]; (ii) el conmutador al que se conectada la raíz precisa de una funcionalidad especial para realizar el *HO*; (ii) la región de movilidad está limitada al área de cobertura de todas las EEBB que forman el árbol de conexiones.

4.4 Leaf Initiated Join Handover (LIJ)

También dentro de los métodos de establecimiento *multicast* encontramos el *LIJ* [5]. La *ITU-T*, al igual que el *ATM Forum* han estandarizado las conexiones punto-multipunto. Dos capacidades principales se requieren para este esquema de *HO* de red: (i) conexiones punto-multipunto; y, (ii) asociación autónoma de un terminal (hoja) a una conexión punto-multipunto (árbol). Actualmente sólo está estandarizado por el *ATM Forum* como opcional en la *UNI 4.0*. Cuando se determina la necesidad de un *HO*, la EB candidata solicita su inclusión al árbol que soporta la conexión entre origen y EB antigua. Mediante la capacidad definida en la *UNI 4.0*, la EB puede hacerlo sin incorporar protocolos de *HO* adicionales. Una vez añadida la hoja, la red envía el flujo de datos del origen hacia las dos EEBB. La situación con dos ramas en la conexión punto-multipunto durante el *HO* no representa una condición muy costosa. El principal inconveniente de este método es que sólo es capaz de soportar conexiones unidireccionales hacia el móvil con los *AAL* actuales.

4.5 Extensión de camino

El método de extensión de camino se basa en la idea de ir encadenando las diferentes EEBB a medida que avanza el móvil. Más concretamente la idea es que, el conmutador que dispone del radiopuerto que funciona como EB antigua cree un circuito virtual entre

él y la EB candidata, y cambie sus tablas de encaminamiento para enviar por esta extensión de VC las células que antes entregaba al radio puerto. Este método es rápido ya que no hay búsqueda de ningún nodo específico que realice la conmutación. Como inconvenientes principales encontramos: (i) la extensión del VC introduce la utilización de rutas no óptimas, incluida la posible aparición de bucles; (ii) aumento del retardo a medida que se va extendiendo el camino, lo que puede traer problemas en el cumplimiento del *cell delay transfer (CTD)*.

5 Hacia la implementación de los algoritmos de *handover*: Señalización

La gran mayoría de los métodos que se plantean para incorporar capacidad de movilidad a las redes ATM pasan por añadir capacidades a los conmutadores que antes formaban la red fija. Cualquier método de los que se han planteado, se soporta sobre mensajes que intercambian los diferentes elementos de red, y no sólo éstos son importantes, sino que también qué canales se utilizarán. Existen tres métodos para soportar la señalización:

(i) Incorporar la señalización necesaria para el HO sobre el canal de señalización ATM. Esto supone revisar la señalización UNI y PNNI y añadir ahí la señalización necesaria para el HO [6] [10].

(ii) Utilización de otra conexión ATM diferente para la señalización del HO. Se pueden interconectar los conmutadores que intervengan en el HO mediante conexiones punto a punto (sobre un identificador bien conocido VPI/VCI) [6] [10].

(iii) Señalización dentro de banda. Los mensajes de señalización fluyen en la misma conexión de usuario transportados en células especiales. Estas células serían transparentes para los conmutadores que no incluyan capacidad de movilidad e interpretadas por los que sí la incluyen. Además permite marcar el flujo de datos sin necesidad de numerar las células, lo cual es muy útil para la sincronización. Existen ya propuestas para este tipo de señalización como la que se plantea en [2].

El *ATM Forum* plantea en [10] cualquiera de las opciones (i) y (ii) como posibles.

6 Hacia un *handover* sin interrupción. Sincronización

Un problema que surge asociado al HO es la posible pérdida de la secuencia de correcta de paquetes con la información de usuario que se entrega al móvil. Dado que el HO implica un cambio en la ruta de la información de usuario, manteniendo origen y destino, durante este cambio es posible perder o duplicar células. Será necesario proveer ciertos mecanismos para eliminar o minimizar este problema. Los llamaremos, mecanismos de sincronización. Concretamente, la pérdida de la secuencia de entrega de células puede ocurrir por:

(a) Ejecución del re-encaminamiento en la red fija, es decir, cómo, un conmutador (COS) modifica sus

tablas de encaminamiento para llevar a cabo el reencuentro de una conexión. Según el método que se utilice, pueden aparecer períodos de vulnerabilidad en los que las células se descarten o dupliquen en el conmutador. Los tres métodos que se pueden utilizar son [7]: (i) *break-make*, elimina la conexión antigua antes de establecer la nueva, se pueden perder células durante el cambio; (ii) *make-break*, antes de eliminar la conexión antigua se crea la nueva, aparece un período en el que se envían células por las dos ramas, es decir, un funcionamiento *multicast* que puede provocar duplicados; (iii) *atomic*, el conmutador debe realizar las operaciones anteriores de manera atómica sin manejar ninguna célula entre los dos pasos, de esta forma se asegura que en éste no se producirán ni pérdidas ni duplicados.

(b) Si se utiliza un método del tipo (i), además de las posibles pérdidas que introduce intrínsecamente, pueden producirse más errores. Debe asegurarse que tanto los conmutadores de la rama antigua como el móvil conozcan cuál es la última célula que va a circular por esta rama y no cerrarla hasta que todas sean recogidas en el móvil. Además de asegurar que el móvil recoja hasta la última célula por el canal antiguo las células no deben perderse en el nuevo canal mientras el móvil aún no ha conmutado hacia éste. Este sería el caso, por ejemplo, de un *hard-handover* en el que la EB candidata debe encolar todas las células mientras aún no se haya establecido el canal radio.

(c) Si se utiliza el método (ii), además de evitar ninguna pérdida de células, se debe tener en cuenta que no existe, a priori, una última célula por el canal antiguo. Ésta debe negociarse entre EEBB o en el móvil para evitar solapes o huecos, a esta célula negociada la llamaremos punto de conmutación. Nuestra red experimental ATM implementa un algoritmo para negociar el punto de conmutación, ver punto 7.4.

7 Implementación de una red experimental ATM para comunicaciones móviles multimedia.

Se ha llevado a cabo la implementación de una red experimental ATM para comunicaciones móviles multimedia. En [9] se presenta con detalle el marco del proyecto, sus objetivos, una descripción de las diferentes partes de la red experimental y sus relaciones. Los objetivos fundamentales fueron evaluar el efecto, tanto de la conexión radio como, más concretamente, del HO, sobre el tráfico multimedia. La red experimental consta de dos partes: (i) red fija ATM real, formada por dos EEBB, un servidor y un conmutador ATM; y, (ii) red radio simulada, que permite la interconexión entre EEBB y móviles. En la figura 3 se muestra tanto la estructura física como la arquitectura de protocolos de la red experimental, se ha escogido de esta forma con el fin de crear un escenario lo más próximo posible al entorno *wireless-ATM* [9].

Este artículo se centra en el tema de la sincronización y el *HO*. En las secciones anteriores se ha descrito la problemática y soluciones que aparecen en la literatura. En esta sección justificaremos las decisiones de implementación tomadas. Concretamente nos centraremos en: (i) el método de *HO* de red fija implementado; (ii) el tipo de *HO* radio; (iii) el tipo de señalización; (iv) el método de sincronización. Estos problemas se resuelven en la capa de sincronización y *HO* (*SHO*) de la arquitectura de protocolos, figura 3. A la hora de llevar los métodos o ideas teóricas a la práctica nos encontramos con limitaciones que nos hacen llegar a soluciones simplificadas o a elegir opciones subóptimas. A continuación se discute cuáles fueron y por qué las soluciones adoptadas.

7.1 Handover de red fija. Discusión del método adoptado

Se dispone de una red *ATM* compuesta por un conmutador convencional y tres estaciones de trabajo, con sendas tarjetas *ATM*, la cual no incorpora ninguna funcionalidad específica para el soporte de movilidad. La mayoría de los métodos descritos para la implementación del *HO* de red requieren que el conmutador incorpore nuevos módulos de señalización que se ejecuten durante el *HO*. El conmutador que se utilizó no permite la programación externa y funcionaba como una caja negra a la que sólo era posible acceder mediante la señalización *UNI* a través de los puertos *ATM*, o mediante comandos de gestión a través de un puerto *Ethernet* o *RS-232*. Por lo tanto se descartó la posibilidad de incluir rutinas específicas para el *HO* en el conmutador. La siguiente opción fue buscar métodos que pudiéramos implementar trasladando el procesamiento necesario a las estaciones de trabajo que hacen de servidor y *EEBB*.

El método de establecimiento total requiere la participación de la fuente en el *HO*, ello no es adecuado para ser usado en redes públicas. El método *NCNR*, que quizá sea el más evolucionado, queda descarta-

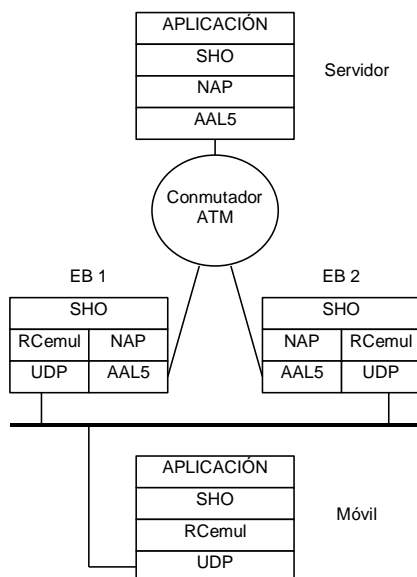


Figura 3: Arquitectura de la red experimental ATM.

do ya que necesita de rutinas en los nodos de la red [3]. El método extensión de camino puede ser implementado si es posible acceder a las tablas de encaminamiento del conmutador y crear conexiones entre el conmutador y un punto final. Esto no era posible mediante señalización *ATM*, por lo que también quedó descartado. Quizá, utilizando los comandos de gestión sería posible, pero esta opción no es adecuada.

Únicamente quedaban por considerar dos métodos: *VCT* y *LIJ*. El método *VCT* podía implementarse con facilidad ya que las operaciones que conlleva pueden perfectamente realizarse desde los puntos finales, aunque en realidad deberían realizarse en los conmutadores. El problema de éste es que es ineficiente en la reserva de recursos y limita la movilidad a pequeñas redes privadas. El método *LIJ* también permitía su implementación. Aunque la red soporta *UNI* 3.1, que no permite la asociación autónoma de hojas, es posible mediante software alojado únicamente en las estaciones de trabajo, proporcionar esta funcionalidad. Ésta es la misión de la capa *NAP* que aparece en la figura 3. El principal problema de este método es que no permite comunicaciones bidireccionales, este problema no afecta al objetivo de la red experimental *ATM* ya que éste es evaluar el efecto del *HO* sobre el tráfico multimedia, de un servidor hacia un usuario. Éste fue, por tanto, el método que se eligió.

7.2 Handover radio. Discusión del método adoptado

Una vez escogido el método *LIJ* buscamos el tipo de *HO* radio que mejor se adaptaba a este método. En este caso no encontramos limitaciones ya que la red experimental *ATM* emula el canal radio sobre una *Ethernet* (capa *RC_emul* de la figura 3) y podíamos implementar cualquier tipo de *HO* radio.

El *HO* es controlado por el móvil para simplificar la emulación, es decir, un *HO* del tipo *MCHO*. El móvil detecta la necesidad de *HO* y propone una *EB* candidata, esta información se genera en la capa *RC_emul* del móvil y la comunica a la capa *SHO*. Ésta última se encargará de realizar el traspaso del radiocanal.

Debido a que el re-encaminamiento de la conexión en la red fija se realiza desde la *EB* candidata mediante una primitiva *LEAFCONNECT.req*, de la capa *SHO* a la capa *NAP*, para añadirse como hoja al árbol que soporta la conexión de aplicación, y que la *EB* antigua no interviene en este proceso, se presta perfectamente a realizar un *forward-handover*. Si se hubiera realizado un *back-handover*, debería haberse establecido un canal entre *EEBB* por el que la *EB* antigua le comunicara a la *EB* candidata la información que recibió del móvil. Como se puede apreciar, un *forward-handover* simplificaba el proceso.

En el entorno de comunicaciones móviles multimedia, en el que se enmarca la red experimental, interesa reducir al máximo posible el lapso de tiempo sin tráfico que puede introducir el *HO* en el flujo multimedia. Por esta razón optamos por un traspaso del

radiocanal del tipo *soft-handover*. Utilizando esta técnica el móvil dispone simultáneamente de los dos flujos, puede decidir, él mismo, el punto de conmutación y la única pérdida de la cadencia de entrega será producida por la diferencia de retardos de ambos caminos. En un traspaso de la forma *hard-handover* siempre existe el lapso de tiempo que transcurre entre que se cierra el antiguo canal radio y se abre el nuevo en el que no se recibe información de usuario, situación que puede alterar el perfil de tráfico que se entrega a la aplicación.

7.3 Protocolo de *handover*

La figura 4 muestra el protocolo de *handover* utilizado. Las flechas de trazo grueso corresponden a mensajes de señalización entre entidades de nivel SHO y las flechas de trazo fino a primitivas de diálogo entre capas. Se pueden apreciar todas las características hasta ahora comentadas. La capa RC_emul solicita el HO mediante ALERT_HO.ind e informa de la EB candidata. La capa SHO se provee de un canal bidireccional (dos unidireccionales, ya que la capa RC_emul ofrece únicamente canales de este tipo) de control con la EB candidata, sobre el que se realizará el intercambio de mensajes entre las entidades del nivel SHO. El móvil solicita a la EB candidata que se añada como hoja al árbol que soporta la conexión de aplicación (ADDLEAF). La EB establece el canal radio de bajada para el soporte del tráfico de usuario y posteriormente solicita a la capa NAP (LEAFCONNECT.req) la inclusión como hoja a la conexión, una vez completada la operación, la confirma al móvil (OK_LEAF). El intercambio de señales desde ADDLEAF hasta OK_LEAF se repite para cada conexión de nivel SHO. Una vez reencaminadas todas las conexiones se indica a la capa inferior la finalización del HO (ALERT_HO.resp).

7.4 Método de sincronización implementado

Debido a las limitaciones expuestas en el punto 7.1 no era posible trabajar en red ATM a nivel de célula sino que únicamente a nivel de usuario de AAL5. Es por esto que no podíamos plantear mecanismos de sincronización a nivel de célula. La sincronización se implementó a nivel SHO, consistió en la entrega de las PDU's de la capa SHO en la secuencia adecuada a la capa de aplicación del móvil ocultando los efectos del HO. Los mecanismos a utilizar son los mismos que a nivel de célula, tal y como se expone en la sección 6 pero con mayor facilidad ya que ahora sí que es posible añadir números de secuencia a las PDU's.

El método LIJ implementado se puede considerar como del tipo *make-break* o *multicast* por tanto exige de una negociación del punto de conmutación entre el flujo de la EB antigua y la nueva. Básicamente el algoritmo de negociación que se implementó fue el siguiente: Se dispone de dos entidades, una atendiendo al flujo antiguo y otra al nuevo, situadas en el móvil o en las EEBB antigua y candidata respectivamente, según el tipo de HO radio. Una vez comenzado el HO, la entidad que atiende al nuevo

flujo comunica a la otra cuál ha sido el número de secuencia de la primera PDU que ha recibido (señal CONMUTAR de la figura 5), llamémosle Nsec₁, y almacenará todas las PDU's que reciba a partir de ese momento. Al recibir este mensaje la entidad que atiende al flujo antiguo calculará el punto de conmutación (Nsec₂), éste será: (i) si ya ha entregado la PDU con número de secuencia Nsec₁, Nsec₂ será igual al número de la siguiente PDU que debe entregar (para evitar solapes), en ese momento lo comunica a su homóloga (señal ARRANCAR de la figura 5) y cierra el canal radio con la EB antigua; (ii) si aún no ha entregado la PDU Nsec₁, esperará a entregar hasta la anterior (para evitar huecos) y cerrará el canal comunicando a su homóloga que continúe en Nsec₁. En nuestra red experimental ATM, este intercambio de señales transcurre en el móvil, es decir, tanto la entidad que atiende a la conexión antigua como la que atiende a la nueva conexión residen en el móvil. Esto es posible gracias al *soft-handover* que permite al móvil disponer de ambos flujos simultáneamente. En caso de realizar un *hard-handover* habría que trasladar estas máquinas de estados a las EEBB y habilitar un canal entre éstas para el diálogo.

7.5 Resultados

Aunque la capa de aplicación de la red experimental ATM (figura 3) incluye aplicaciones de vídeo, el tipo de tráfico que éstas generan no es el más adecuado para evaluar cuantitativamente las características de la red experimental. Este tráfico está compuesto por paquetes de longitud variable generados a tasa variable dependiendo de la imagen y la codificación. Con este tipo de tráfico es más complejo, realizar medidas relativas al *jitter* de paquetes tanto en funcionamiento normal como durante un HO. Por este motivo se utilizó un generador de tráfico sobre la capa SHO (figura 3) del servidor que generaba pa-

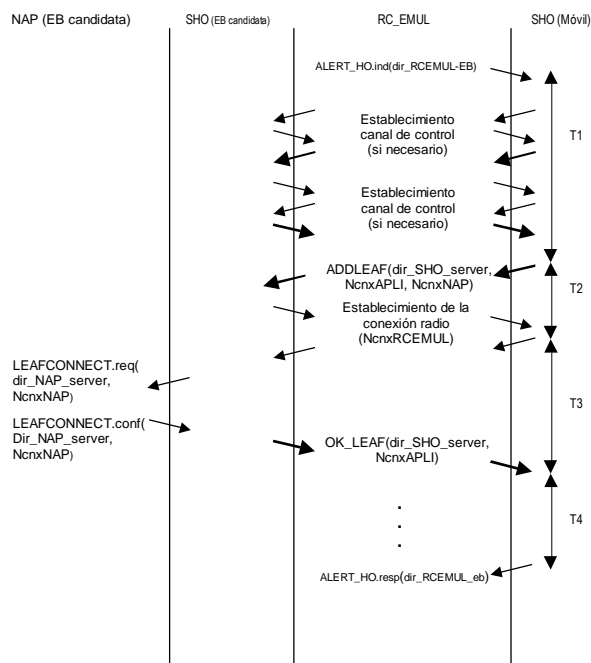


Figura 4: Protocolo de *handover*.

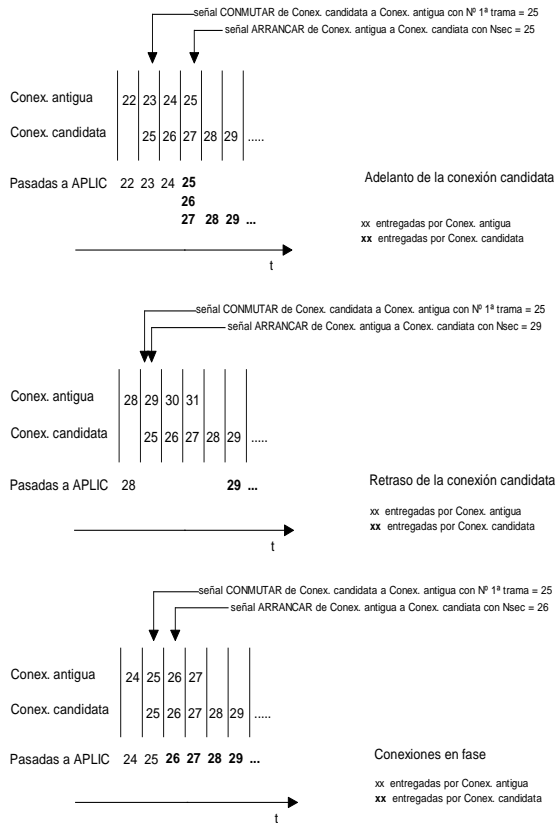


Figura 5: Evolución del algoritmo de sincronización de flujos

quetes de longitud constante a cadencia constante.

El escenario definido para la realización de las medidas fue el siguiente: Se establece una conexión a nivel SHO entre servidor y móvil por la que se introduce el tráfico sintetizado por el generador. En el móvil, mediante llamadas al sistema operativo, se mide el tiempo transcurrido entre eventos relevantes.

La figura 6 representa el tiempo entre cada par de paquetes que la capa SHO, en el móvil, entrega a la capa de aplicación. Es una medida del *jitter* que introduce la red. Los paquetes se generan en el servidor con una cadencia constante de 20 ms y un tamaño de 3000 bytes. Esto supone un flujo de 1.2 Mbps que se puede comparar al que podría emitir una aplicación común de vídeo. Las líneas discontinuas verticales indican los momentos en los que se ha realizado un *HO*. El primer paquete válido por el nuevo canal tras un *HO* no sufre un retardo especialmente elevado; concretamente, si la cadencia media es de 20 ms, estos paquetes se reciben, en media, 23.9 ms después del último paquete válido recibido por el canal antiguo, debido a la diferencia de retardos entre ambos canales y al tiempo de procesado necesario para realizar la sincronización y conmutación de los canales en el móvil. Con este retardo, estos paquetes pertenecen a la franja central, entorno a los 20 ms, que concentra la gran mayoría del tráfico. Se pueden apreciar dos franjas a 32 y 8 ms respectivamente, éstas corresponden a los paquetes retransmitidos por la capa RC_emu1, a 32 ms llegan los paquetes que han sufrido una retransmisión y a 8 ms el siguiente al retransmitido. Los pun-

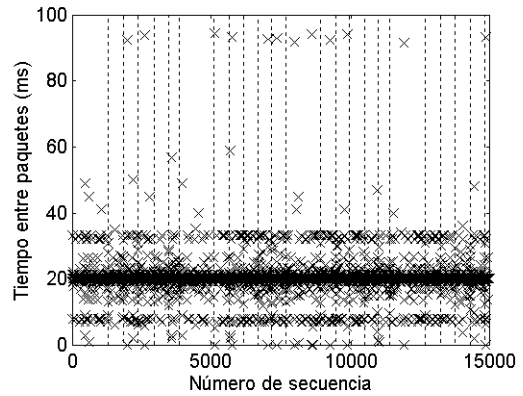


Figura 6: Jitter de la transmisión.

tos dispersos que se observan por encima de los 40 ms y por debajo de los 6 ms son debidos al sistema operativo, cuando no concede la UCP a los procesos que la necesitan para realizar operaciones propias como: escribir en fichero, realizar swaping,... provocando que el tráfico se frene en las colas. Se ha probado la red experimental ATM con aplicaciones de vídeo reales y soportan perfectamente, sin ninguna disminución de la calidad, el *jitter* que introduce el sistema operativo (los lapsos de casi 100 ms sin paquetes) mediante *buffers* que lo absorben, por lo tanto sobradamente soportan el *HO*, el cual introduce una variación de unos 4 ms en la cadencia de entrega.

La figura 7 muestra el funcionamiento del algoritmo de sincronización. Representa el deslizamiento, medido en paquetes, que percibe el móvil, entre el flujo nuevo y el antiguo. Recordar que *Nsec CONMUTAR* es el número de secuencia del primer paquete que llega por el nuevo canal y *Nsec ARRANCAR* el del primer paquete que se entregará a la aplicación por el nuevo canal tras un *HO*, por lo tanto, su diferencia indica cuantos paquetes va retrasado el canal nuevo con respecto al antiguo. Esta gráfica se ha tomado con un tráfico diferente al anterior, formado por paquetes de 100 bytes cada 5 ms, de esta forma se aprecia mucho mejor el efecto del desfase. En la gráfica se observa que para este tráfico, el deslizamiento oscila entre dos y tres paquetes. Esta medida, realizada con un tráfico como el utilizado en la figura 6, produce deslizamientos de a lo sumo un paquete y en raras ocasiones. Es lógico, con paque-

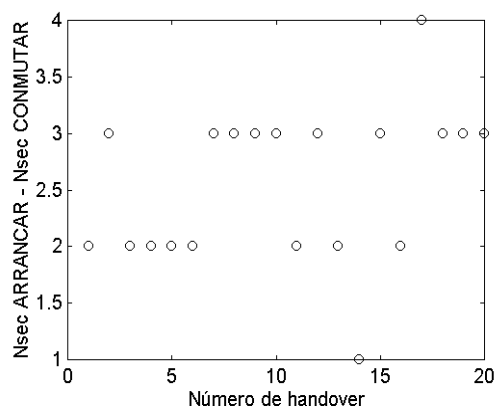


Figura 7: Desfase entre flujos.

tes más grandes y más espaciados debe de haber mucho desfase entre los canales para deslizar más de un paquete. Se observa por tanto la necesidad de la sincronización ya que las aplicaciones multimedia igual generan paquetes cortos y muy poco espaciados, que largos y más separados, dependiendo del tipo de codificación.

La figura 8 muestra los tiempos indicados en la figura 4, representados de forma acumulada. T1 corresponde al tiempo de establecimiento del canal de control; T2, al tiempo de establecimiento del canal radio; T3, a la duración del HO de red fija, vista desde el móvil; y T4, al tiempo necesario para la sincronización. El tiempo total del HO es, en media, de 213 ms, comparable al del sistema DECT (100-500ms) y bastante menor que el de GSM (1s) [1]. La medida está realizada para una sola conexión en el móvil. En el caso de soportar más, este tiempo se incrementa de la forma $(T2+T3)*N^{\circ}_{de_conexiones}$ ya que estas operaciones se realizan secuencialmente para cada conexión. La sincronización se realiza en paralelo por lo que T4 no aumenta apreciablemente por el hecho de soportar más conexiones. En cuanto a T1, como el canal de control es compartido, es independiente del número de conexiones. T3 consume el 42% del tiempo del HO por lo que debería ser el primer paso en un proceso de refinamiento si interesase reducir este tiempo. El tiempo de duración del HO no tiene relación con el jitter en la entrega de paquetes, en cambio, es importante para el diseño de las zonas de solape entre células. Atendiendo a la velocidad media a la que se espera que atraviesen los móviles de una célula a otra, el móvil debe permanecer en la zona de solapamiento de las dos EEBB al menos mientras dura el HO.

8 Conclusiones

Se han presentado los problemas y soluciones para el HO y sincronización en redes wireless-ATM. Para que éstas se conviertan en una realidad comercial deberá pasarse por una estandarización de algún método. Como siempre ésta pasará por un compromiso entre prestaciones y complejidad. En este caso, la complejidad incluye la cantidad de cambios a realizar sobre los diseños ya existentes de conmutadores que no soportan movilidad.

Mediante la red experimental ATM diseñada se ha podido verificar que uno de los métodos de HO de red que menor impacto supone para la red fija (LJ), tiene buenas prestaciones temporales. También se ha podido observar que debe ser implementado algún mecanismo de sincronización para minimizar la tasa de error de paquetes. Como conclusión más importante hemos podido determinar que la ejecución de un HO, implementado con los procedimientos descritos, es totalmente transparente a cualquier aplicación común para la transmisión de vídeo, no suponiendo ninguna merma en su calidad de reproducción.

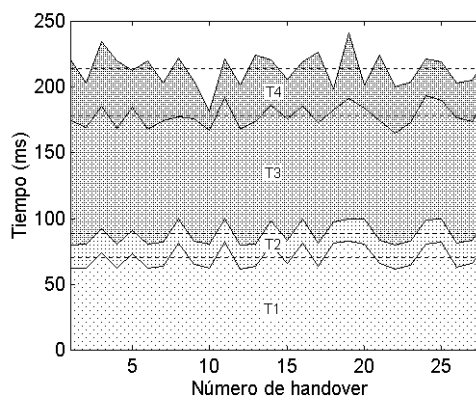


Figura 8: Duración del handover.

Referencias

- [1] A. Noerpel, Yi-Bing Lin, "Handover Management for a PCS Network", IEEE Personal Communications, pp. 18-24, December 1997.
- [2] M.A. Marsan, et al., "Local and Global Handovers for Mobility Management in Wireless ATN Networks", IEEE Personal Communications, pp. 16-24, October 1997.
- [3] A. Akyol, D.C. Cox, "Rerouting for Handoff in a Wireless ATM Network", IEEE Personal Communications, pp. 26-33, October 1996
- [4] A.S. Acampora, "An Architecture and Methodology for Mobile-Executed handoff in Cellular ATM Networks", IEEE J. Select. Areas Commun., Vol. 12, No 8, pp. 1365-1374, Oct. 1994.
- [5] M. Teughels, E.V. Lil, A.V. de Capelle, "Hand-over in Wireless ATM: signalling and simulation results", The 10th International Conference on Wireless Communications Proceeding, pp. 526-538, July 1998.
- [6] J. Naylor, et al., "Low-Latency handover in a Wireless ATM LAN", IEEE J. Select. Areas Commun., Vol 16, No 6, pp. 909-921, August 1998.
- [7] R. Ramjee, J. Kurose, "Performance Evaluation of Connection Rerouting Schemes for ATM-Based Wireless Networs", IEEE Trans. on Networking, Vol. 6, No 3, pp. 249-261, June 1998.
- [8] V. Martínez, "Diseño e implementación de la función de sincronización y handover para un demostrador de comunicaciones móviles multimedia", Documento técnico TIC96-0680, Noviembre 1998.
- [9] J. Martínez, J.R. Vidal, L. Guijarro, "Red ATM Experimental para Comunicaciones Móviles Multimedia", submitted to JITEL99, April 1999.
- [10] R.R. Bhat, "Draft baseline test for Wireless ATM Capability Set 1 Specification", ATM Forum/BTD-WATM-01.1., December 1998.

Red ATM Experimental para Comunicaciones Móviles Multimedia*

J. Martínez, J.R. Vidal y L. Guijarro
Área de Ingeniería Telemática.
Departamento de Comunicaciones. Universidad Politécnica de Valencia.
Camino de Vera s/n. 46022-Valencia.
E-mail: jmartinez@upvnet.upv.es

Abstract

This paper describes an ATM network with wireless access for multimedia communications. The network has been specifically designed to serve as a testbed on which evaluate the impact that different handovers techniques have on multimedia traffic. The access network has been completely emulated, where the behaviour of the radio channel is emulated by software and its transmission functions supported by an Ethernet LAN. The paper discusses the design decisions that have been taken and provides an example showing how a handover takes place.

1 Introducción

El Modo de Transferencia Asíncrono (ATM) junto con la señalización definida para la RDSI de banda ancha se están convirtiendo el común denominador de todo tipo de redes y servicios. La tecnología ATM permite integrar en una misma infraestructura de red tanto servicios de tiempo real como servicios de datos, además sus prestaciones se mantienen prácticamente invariantes a medida que aumenta el tamaño de la red. Aunque inicialmente ATM se diseñó con el objetivo de soportar la red troncal, en la actualidad se ha extendido el concepto a entornos de red de acceso inalámbrico, permitiendo extender los servicios de red, ya suficientemente consolidados, a terminales móviles.

En 1996 el Departamento de Comunicaciones de la Universidad Politécnica de Valencia comenzó a trabajar en la definición de una red ATM experimental que permitiese avanzar en el estudio de los problemas que plantea el soporte de comunicaciones móviles multimedia. Este artículo describe el marco en el que se circunscribió dicha red experimental, su arquitectura funcional y su arquitectura de protocolos.

El resto del artículo se ha estructurado de la siguiente forma. En la sección 2, se describen las tecnologías de acceso inalámbricas existentes en el momento en el que se propone el concepto *Wireless ATM* (WATM), así como los elementos funcionales típicos y la arquitectura de protocolos definida en WATM. En la sección 3, se analizan los aspectos más relevantes en el diseño de una red WATM como son, por ejemplo, el *handover* y la gestión de la movilidad. En la sección 4, se describe la red ATM experimental diseñada, su objetivo y los elementos que la componen. En la sección 5, se describe la arquitectura de protocolos diseñada para la red experimental y se analiza su relación con la arquitectura definida en WATM. En la sección 6, se relacionan las medidas realizadas. Finalmente, en la sección 7

se presentan las conclusiones y futuras líneas de trabajo.

2 Tecnologías en las Redes de Acceso Inalámbricas

La búsqueda de soluciones para las redes de acceso inalámbricas no es nuevo, pero se había centrado en soluciones para el soporte de tráfico de datos. De ellas, sólo son relevantes las que ofrecen una arquitectura celular basada en la reutilización de frecuencias, puesto que son las únicas que permiten crecer con el número de abonados. En la Tabla 1 se describen las características de aquellas que se han considerado con más interés comercial.

Tabla 1. Tecnologías de Acceso Inalámbricas

Tecnología	Tasa Binaria
GSM(GPRS)	<144 Kbit/s
UMTS y WLANs	0.3-2 Mbit/s
HIPERLAN	20 Mbit/s

Las redes WLANs (*Wireless LANs*) se han diseñado para entornos de oficina, haciendo uso de enlaces radio de corta distancia o bien enlaces ópticos de infrarrojos, para ofrecer una alternativa flexible y competitiva al cableado convencional. Desgraciadamente, estas ventajas no han sido suficientes para el éxito generalizado de esta tecnología ya que, en general, es necesario realizar un cableado para la red telefónica y el coste incremental para el cableado de la red de datos es pequeño. Adicionalmente, las redes WLANs ofrecen capacidades muy limitadas de movilidad, por lo que no son adecuadas para muchas aplicaciones.

En cambio, sistemas de red pública de segunda generación como GSM, o de tercera generación como UMTS, permiten ofrecer capacidades de movilidad en vastas zonas geográficas. Desgraciadamente, el soporte que ofrecen a las comunicaciones multimedia es muy limitado. En el caso de GSM (GPRS) por el reducido ancho de banda de que dispone y porque

* Este trabajo ha sido financiado por la Comisión Interministerial de Ciencia y Tecnología (CICYT) a través de los proyectos TIC96-0680 y TIC98-0495-C02-02.

está pensado únicamente para el soporte de tráfico de datos. En el caso de UMTS, aunque se contempla

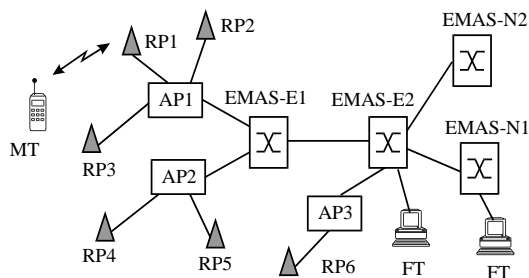


Figura 1. Arquitectura funcional definida para WATM.

ofrecer anchos de banda de hasta 2 Mbit/s, es previsible que se optimice para el soporte de servicios de voz y se mantenga la compatibilidad en los servicios de datos, sacrificando de esta forma una verdadera integración de servicios.

En este escenario surge WATM como una tecnología de acceso inalámbrico a las redes ATM sin discontinuidades*. Cuando el terminal es móvil, WATM permite, en principio, preservar las garantías de QoS extremo a extremo de las clases de servicio ya definidas para la red fija y además ofrece capacidad ilimitada de movilidad a los usuarios.

En Diciembre de 1995 el ATM Forum comenzó sus trabajos para definir la arquitectura WATM. Estos trabajos se han realizado en colaboración con el *European Telecommunications Standards Institute* (ETSI), cuyo grupo de estudio RES10 se ha centrado básicamente en la definición del interfaz aire. Para ello, el ETSI ha aprovechado el trabajo realizado para el HIPERLAN II (*High Performance Radio LAN*) que definía un interfaz en la banda de los 5.2 Ghz, con un ancho de banda de unos 20 Mbit/s y un diámetro máximo de célula de 50 m.

Se han contemplado dos posibles escenarios de interconexión a través de interfaz aire: la de terminales fijos, es decir sin soporte de movilidad, y la de terminales móviles. En la Fig. 1 se relacionan las diferentes agrupaciones funcionales definidas para el segundo escenario. Como puede observarse, se han definido unos conmutadores específicos denominados *End-user Mobility Supporting ATM Switch* (EMAS) para el soporte de la movilidad. Los EMAS pueden ser de dos tipos: los EMAS-E (*Edge*) que poseen Radio Puertos (RP) y los EMAS-N (*Network*) que no los poseen. Los EMAS-E pueden utilizar unos concentradores de tráfico exteriores denominados *Access Points* (APs), a donde se conectan los RPs. Como puede también verse, la interconexión entre los EMAS se realiza por medio de enlaces

cableados. Los terminales móviles (*Mobile Terminals*, MTs) acceden a la red a través de los RPs, mientras que los terminales fijos (*Fix Terminals*, FTs) se conectan a los EMAS-N.

En la Fig.2 se muestra la arquitectura de protocolos definida, tanto para el plano de usuario como para el de señalización. En ambos planos la capa WAL (*Wireless Access Layer*) permite el acceso al interfaz aire e incluye las capas PHY, MAC y DLC. La capa MAC permite gestionar el acceso al medio de forma que pueda garantizarse el QoS solicitado por el usuario. El subsistema clave de la capa MAC es el planificador (*scheduler*) que permite asignar recursos disponibles a las conexiones en curso [1]. La capa DLC permite ofrecer un servicio libre de errores para el tráfico sensible a las pérdidas.

En el plano de usuario, por encima de la capa WAL, se mantienen las mismas capas que en una red ATM fija.

En el plano de señalización existen dos aproximaciones: o bien añadir un protocolo independiente que

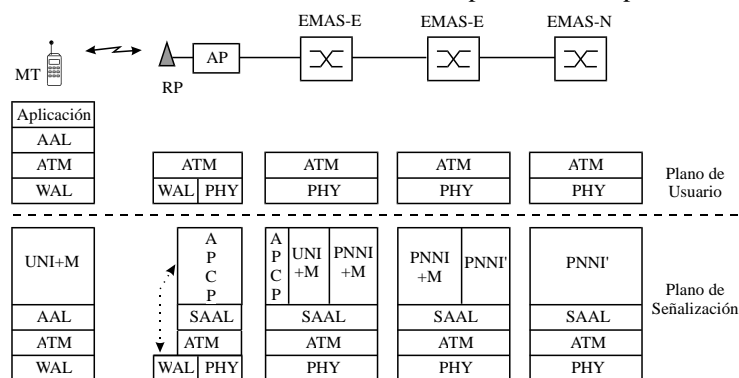


Figura 2. Arquitectura de protocolos definida para WATM

interaccione con los protocolos de señalización convencionales, o bien diseñar un protocolo nuevo de señalización que incorpore las funciones de movilidad. Por el momento, ambas aproximaciones son todavía objeto de un intenso debate [2]. Así, los protocolos UNI+M y PNNI+M podrían acabar definiéndose por cualquiera de las dos opciones. El protocolo PNNI' soporta las funciones características definidas para el interfaz PNNI pero, como novedad, además soporta el transporte de la información de movilidad [3].

Aparecen también nuevos protocolos, como el *Access Point Control Protocol* (APCP), que permite al EMAS-E tener acceso al estado de los recursos radio.

3 Aspectos Relevantes en el Diseño de una Red WATM

Direccionamiento

La solución actual está basada en la asociación de un nombre y una dirección a cada terminal. Para los terminales fijos, nombre y dirección coinciden. Para los terminales móviles el nombre no cambia, pero la dirección es temporal y cambia al cambiar el AP al

* Del inglés *seamless*.

que se encuentra asociado el MT. Es necesario pues definir una asociación entre ambas [4] y un mecanismo de registro de los MTs en los EMAS-E.

En muchos escenarios es deseable percibir a los APs y a los RPs como elementos de red y no como simples tarjetas *hardware*, por ello, también se propone dotarles de dirección ATM. Esto puede ser útil, por ejemplo, cuando se desea identificar el RP destino de un *handover*.

Multiplexación en los APs

En general, dado que varios MTs pueden compartir el mismo RP se pueden producir duplicaciones en los VPI/VCI utilizados por los MTs. Esto puede ocurrir, por ejemplo, con el canal de señalización utilizado por los terminales en el interfaz UNI. Como es bien conocido, este canal debe ser el VPI=0, VCI=5.

Es pues el AP el encargado de realizar la asociación (Direc. MAC, VPI/VCI) de entrada con el VPI/VCI de salida hacia el EMAS-E. Además, cada vez que se produce el registro de un MT, el EMAS-E crea un "UNI virtual" asociado al nuevo MT. Este UNI virtual permite al conmutador comunicarse con el MT como si fuera un terminal fijo conectado a un UNI físico [3].

Handover

Diferentes escenarios de *handover* son posibles: entre RPs del mismo AP, entre RPs de diferentes APs pero pertenecientes al mismo EMAS-E o entre RPs de diferentes EMAS-E. En general, sólo el primero puede resolverse sin activar los mecanismos de *handover* a nivel ATM, los cuales producen la modificación de los VPI/VCIs de ciertos segmentos de la conexión.

El *handover* es iniciado por el MT y puede ser *backward* o *forward*, dependiendo de que el EMAS-E con el que se intercambia la señalización sea el actual o el candidato. En [3] se reserva la denominación *handover forward* para aquellos casos en los que el terminal realiza una disociación abrupta con su RP actual y solicita la asociación con un nuevo RP. Además, en [3] sólo se contemplan *hard handovers*, es decir, aquellos *handovers* en los que se recibe información desde el nuevo RP tras haber dejado de recibir desde el RP antiguo, y por tanto no se recibe nunca simultáneamente desde ambos RPs.

El MT es capaz de identificar diferentes RPs y proveer durante el *handover* una lista priorizada de los RPs destino deseados. Aunque la re-negociación de parámetros de tráfico y de QoS puede ser necesaria, ésta no se contempla en [3]. La información que se intercambian durante el *handover* el MT y el EMAS-E, y el EMAS-E actual y el EMAS-E candidato, se transporta por unos canales de control bien conocidos.

Se han sugerido diferentes aproximaciones para hacer posible el re-encaminamiento de conexiones durante un *handover*. En general, el re-

encaminamiento consiste en liberar la parte de la conexión definida entre el EMAS-E actual y un EMAS-N intermedio (que actúa como nodo raíz) y el establecimiento de una nueva rama de la conexión entre el EMAS-E candidato y el EMAS-N intermedio. A este EMAS-N se le denomina *Crossover Switch* (COS) [5].

Se pueden presentar diferentes escenarios de re-encaminamiento dependiendo de la localización del COS. En el caso de un *handover* intra-EMAS-E, es decir el RP destino pertenece al mismo EMAS-E, entonces el COS y el EMAS-E son el mismo equipo y por tanto no es necesario realizar re-encaminamiento. En el caso de un *handover* inter-EMAS-E, existen diferentes posibilidades:

1. El EMS-E actual y el COS son el mismo equipo, en cuyo caso la conexión se extiende hacia el EMAS-E candidato. Esto es equivalente a la aproximación *Path Extension* utilizada en [6].
2. El COS es conocido por el EMAS-E actual y por tanto no necesita ser descubierto dinámicamente. Este conocimiento es posible bien por configuración o bien por topología. A esta aproximación se la denomina *Anchor Rerouting*.
3. El EMAS-E actual necesita descubrir el COS dinámicamente para lo cual se han propuesto diferentes alternativas [7].
4. El EMAS-E candidato necesita descubrir el COS dinámicamente para lo cual se han propuesto diferentes alternativas [7].

Las aproximaciones *Path Extension* y *Anchor Rerouting* pueden ser ineficientes en ciertas circunstancias, al dar lugar a rutas sub-óptimas que consumen demasiados recursos de red. Por ello, la propuesta del ATM Forum [3] soporta mecanismos para descubrir el COS dinámicamente.

Gestión de la Movilidad

Como ya se ha comentado, para el soporte de la movilidad existen dos aproximaciones: o bien añadir un protocolo independiente que interaccione con los protocolos de señalización convencionales, o bien diseñar un protocolo nuevo de señalización que incorpore las funciones de movilidad. La primera aproximación suele dar lugar a protocolos de señalización demasiado complejos, cuyo rango completo de funciones no son necesarias en las redes ATM fijas.

El diseño de protocolos para el soporte de la movilidad tiene que tener en cuenta el re-encaminamiento, tanto de la conexión del plano de usuario como la del plano de señalización. Además, las acciones implementadas para proveer tanto canales de señalización como de datos al MT durante el *handover*, deben ejecutarse tan rápido como sea posible. Ello es especialmente importante cuando la conexión se pierde abruptamente.

4 Descripción de la Red ATM Experimental

Objetivo

El objetivo de la red ATM experimental es estudiar la problemática asociada al soporte de comunicaciones multimedia en un escenario de terminales móviles. Una de las características diferenciales de este tipo de escenarios, respecto al caso en el que los terminales son fijos, es la necesidad de realizar *handovers*, durante los cuales la comunicación puede llegar incluso a interrumpirse durante cortos intervalos de tiempo. De forma más específica, el objetivo de la red ATM experimental es evaluar cuantitativamente el impacto que el proceso de *handover* tiene sobre las comunicaciones sensibles al retardo.

La configuración definida para la red experimental es un compromiso entre los objetivos que se persiguen y la capacidad tecnológica a la que se tiene acceso. Esta arquitectura puede observarse en la Fig.3 Como vemos, la parte de red de acceso se emula en su totalidad. Las funciones de transmisión se realizan sobre una LAN Ethernet, mientras que el comportamiento de los canales radio se emula mediante software. Los APs se soportan sobre dos WS Sun dedicadas a tal efecto, donde se ha supuesto que los APs disponen de un sólo RP. El MT se emula mediante un PC con sistema operativo Linux.

Hay que resaltar que la funcionalidad definida para los APs de la red experimental se ha simplificado muy considerablemente respecto a la funcionalidad definida para los APs en WATM. Esto es debido, por una parte, a que en la red experimental se utiliza un solo MT, y por otra, a que el interfaz del MT con la red es diferente al UNI definido en las redes ATM.

Las características del conmutador y de las tarjetas de red ATM utilizadas son las siguientes.

Conmutador

El conmutador utilizado es un conmutador ATM convencional que no soporta las funciones de movilidad. Concretamente se trata del modelo ASX-200BX, de Fore Systems y tiene las siguientes características:

- Tiene una capacidad agregada de 2.5 Gbit/s.
- La matriz de conmutación no produce bloqueo y dispone las colas a la salida.
- Dispone de 4 ranuras para módulos de red, dos de las cuales están ocupadas por módulos de 4 puertos con interfaz físico OC-3/STM-1 a 155 Mbit/s, con fibra multimodo y conectores SC.
- El retardo de tránsito del conmutador es de 10 microsegundos aproximadamente.
- La matriz de conmutación es del tipo *Time Division Multiplexing*.

- Cada módulo de red dispone de buffers de capacidad de 32768 células ATM; esta capacidad se pone a disposición en un *pool* común para todos los puertos de salida; esta disposición se denomina *distributed shared memory*.

En los puertos de salida se efectúa la gestión necesaria para garantizar el QoS contratado. La revisión de software utilizada implementa la señalización UNI 3.1 del ATM Forum.

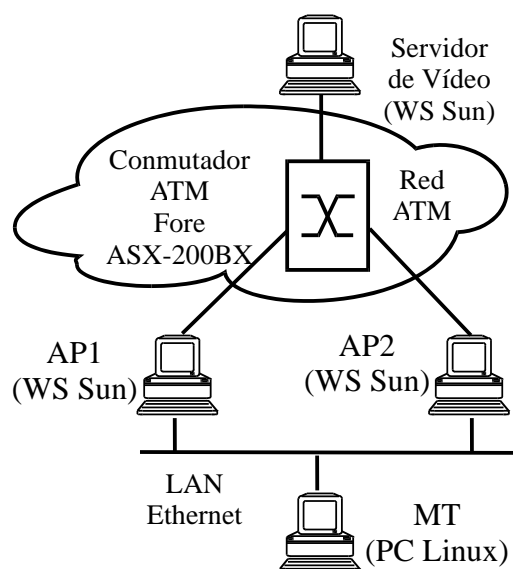


Figura 3. Configuración de la Red Experimental.

Tarjetas de Red

Las tarjetas de red ATM son del tipo ForeRunner SBA-200E. Incorporan el interfaz Sbus y soporte para SunOS 4.13 y Solaris 2.x Incorporan conector tipo SC para fibra multimodo y soportan 155 Mbit/s STM-1/OC-3. Soportan hasta 1024 VCIs.

La tarjeta cumple la especificación de señalización UNI 3.1 del ATM Forum y soporta un interfaz de programación *X/Open Transport Interface* (XTI) que sólo provee acceso al AAL5.

5 Arquitectura de Protocolos de la Red ATM Experimental

Se han propuesto diferentes alternativas para la realización de un *handover*, éstas se analizan en [8]. Para la red experimental se ha optado por implementar aquella basada en la capacidad denominada *Leaf Initiated Join* (LIJ), que permite que un terminal pueda asociarse autónomamente a una conexión punto a multipunto en curso. Esta decisión se ha tomado, por una parte, por ser la que ofrece un uso más racional de los recursos, y por otra, por limitaciones en la tecnología a la que se tiene acceso.

El conmutador utilizado sólo ofrece la versión 3.1 del UNI, por lo que la funcionalidad LIJ ha sido necesario emularla. De ello se encarga la capa *Network Access Protocol* (NAP). Observe su localización en la arquitectura de protocolos diseñada para la red experimental descrita en la Fig. 4.

Cuando se utiliza el AAL5, las comunicaciones punto a multipunto sólo pueden ser unidireccionales. Ello es debido a que la máquina de re-ensamblado en recepción, una vez ha comenzado a re-ensamblar una nueva AAL5-PDU, asocia todas las células que llegan con el mismo VPI/VCI a la misma AAL5-PDU hasta que se recibe la célula etiquetada con *End Of Message* (EOM). Por tanto, la recepción simultánea por la raíz de AAL5-PDUs provenientes de diferentes hojas imposibilitaría el re-ensamblado correcto de las mismas.

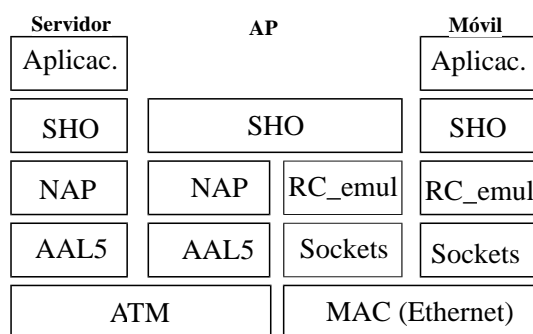


Figura 4. Arquitectura de protocolos diseñada para la Red Experimental.

El establecimiento de conexiones unidireccionales adicionales hacia la raíz, para soportar un flujo bidireccional, no ofrece tampoco una solución al problema. Si se desea respetar el escenario de direccionamiento definido en WATM, el MT debería estar siempre accesible a través del mismo VPI/VCI, incluso cuando cambia de AP. Para ello se requiere tener acceso dinámicamente a la tabla de encaminamiento del conmutador, lo cual no es posible con el conmutador disponible.

Estas limitaciones han restringido la funcionalidad de la red experimental, siendo sólo posible el soporte de comunicaciones multimedia de difusión (*multicast*). Por tanto, la red experimental emula un escenario en el que un MT deambula por una zona de cobertura dada mientras recibe un flujo multimedia unidireccional desde el servidor de vídeo. Durante el trayecto del MT se ejecutan múltiples *handovers*.

Como se puede observar en la arquitectura de protocolos descrita en la Fig. 4, no se han separado los planos de usuario y de control. No obstante, se han definido entes específicos de control en cada capa que se comunican con sus homólogos a través de conexiones de control.

La capa de Sincronización y *HandOver* (SHO) es la encargada de gestionar el proceso de *handover* y de asegurar, en el transcurso de éste, el adecuado secuenciamiento (sincronización) de las PDUs que son transferidas a la capa de Aplicación. Esto es diferente a lo que ocurre en un escenario WATM real, ya que la capa SHO realiza funciones tanto en el plano de usuario como en el de control.

El resto de las capas de la Fig. 4 tienen una funcionalidad bien conocida, como por ejemplo las capas ATM, AAL5 y MAC. La capa *Sockets* se ha preferi-

do denominarla así para explicitar que se usa la librería de funciones UNIX del mismo nombre. Las capas diseñadas se describen con detalle a continuación.

La definición del interfaz de cada una de las capas diseñadas para la red experimental sigue la recomendación UIT-T X.213 “Definición del servicio de red para la interconexión de sistemas abiertos para aplicaciones de la UIT-T”. Aquí, se definen grupos de primitivas para el establecimiento y liberación de conexiones y para la transferencia de datos.

Capa NAP

La capa NAP se encarga de soportar las conexiones del nivel SHO sobre conexiones ATM que ofrecen el QoS adecuado. Además, es la encargada de soportar el re-encaminamiento de parte de la conexión de usuario durante el *handover*. Los detalles de su implementación se describen en [9].

Como se comentó, para su realización se ha decidido implementar la funcionalidad *Leaf Initiated Join* (LIJ). Las conexiones entre un terminal de la red fija y un MT, se soportan sobre la red fija mediante una conexión punto a multipunto entre el terminal fijo y el AP al que está asociado el MT. Cuando el MT determina la necesidad de un *handover*, solicita al AP candidato que se asocie al árbol.

Una vez añadida la hoja, la red envía el flujo de datos desde el servidor hacia los dos APs, y éstos lo re-envían al móvil. Por tanto, éste recibe los flujos desde los dos APs simultáneamente hasta que decide cuando terminar el *handover*. En ese instante, el MT solicita al AP antiguo que se libere del árbol de conexiones.

El escenario de *handover* descrito, con dos ramas en la conexión punto a multipunto, supone un coste extra al requerir el uso por duplicado de recursos de red fija, radio y móvil. Este coste extra es sólo necesario durante el *handover* y representa un coste muy competitivo respecto al resto de alternativas estudiadas [8].

Conviene destacar que la funcionalidad LIJ, que es opcional en el UNI 4.0, no está disponible con el UNI 3.1. Cuando se disponga del UNI 4.0, se podrá, casi sin modificaciones, eliminar la capa NAP y situar la capa SHO directamente sobre la capa AAL5, mejorándose considerablemente las prestaciones la red experimental [9].

Capa RC_emul

El objetivo de la capa de emulación del canal radio (RC_emul) es la de ofrecer un servicio de transporte de información similar al que se obtendría en un escenario WATM real, en el que las comunicaciones entre los MTs y los APs se soportan sobre un canal radio de acceso múltiple. Esta capa emula todos aquellos aspectos cuyo efecto sobre el *handover* resulten de interés.

Como se observa en Fig. 4, la capa RC_emul utiliza el servicio MAC de una red Ethernet como medio

físico de comunicación entre el MT y los APs. La capacidad de esta red es suficiente como para considerar despreciable los retardos de acceso y las pérdidas.

La capa RC_emul sólo ofrece un tipo de servicio portador de tramas de longitud variable y por tanto no es capaz de distinguir entre conexiones de usuario y conexiones de control. El servicio portador ofrecido permite asegurar una transmisión libre de errores y garantiza el orden de entrega.

La capa RC_emul implementa un modelo que define el comportamiento del canal radio en un entorno de comunicaciones móviles terrestres. Este modelo calcula los efectos degradantes que introduce el interfaz radio y emula su efecto sobre la comunicación en curso, mientras el móvil se desplaza sobre un mapa bidimensional dividido en células, en cuyo centro se encuentra un AP. El modelo permite calcular el nivel de la señal en todo momento y obtener la probabilidad de error de bit que el MT puede esperar al comunicarse con cada uno de los APs. A partir de esta información se obtiene el retardo introducido por el interfaz radio. Éste viene determinado por los siguientes componentes [10]:

- Retardo de transmisión. Es básicamente la relación entre la longitud del mensaje y la tasa binaria, ya que el retardo de propagación es despreciable.
- Retardo variable: que depende de la probabilidad de pérdidas o error de tramas, y se modela mediante una función aleatoria que simula el retraso introducido por la retransmisión de las tramas perdidas. Esta componente se aplica tanto al tráfico de usuario como al de control.
- Retardo de acceso: añadido al tráfico que no tiene garantizada una tasa binaria en el interfaz radio, es decir, el de control. Esta componente permite simular el retraso por los procesos de contienda para acceso al canal común.

Aprovechando los parámetros calculados, el MT realiza un control de los niveles de señal recibida comparando la del AP actual con la recibida desde los APs candidatos. Esto le permite decidir sobre la necesidad de asociarse a un nuevo AP, utilizando para ello el protocolo de *handover*.

Capa SHO

La capa de Sincronización y *HandOver* (SHO) se encarga de realizar todas las funciones asociadas a la gestión del *handover*. Su implementación se describe en [8].

Para simplificar la complejidad del diseño se ha optado por que el MT controle el proceso de *handover*. El MT detecta la necesidad de *handover* y selecciona el AP candidato. Esta funcionalidad está localizada en la capa SHO, pero es la capa RC_emul la encargada de proveer la información sobre el estado del interfaz radio necesaria.

También por simplicidad, se ha optado por una técnica de *forward handover*. Así, cuando el MT decide el AP candidato, le solicita a éste que se asocie como hoja al árbol de la conexión que soporta el tráfico de usuario. Si se hubiese optado por la técnica de *backward handover*, hubiese sido necesario que el MT comunicase al AP actual el AP candidato y que el AP actual comunicase al candidato la información necesaria para realizar la asociación al árbol de la conexión de usuario en curso. Un procedimiento algo más complejo.

En un escenario de comunicaciones multimedia, interesa minimizar la longitud temporal de las interrupciones en el flujo de información de usuario percibida por el MT. Por ello se ha optado por la técnica de *soft handover*. Aquí, el móvil dispone simultáneamente de los dos flujos de información y puede decidir por sí mismo el instante en el que comienza a recibir desde el AP candidato. En una técnica *hard handover*, la longitud temporal de la interrupción en el flujo de información de usuario percibida por el MT es mayor puesto que se debe cerrar el canal con el AP actual y abrir el canal con el AP candidato.

La capa SHO se encarga además de las funciones de sincronización. Por sincronización nos referimos a aquellas funciones que permiten que la capa de Aplicación reciba la secuencia correcta de PDUs, es decir, sin duplicaciones ni pérdidas. Piénsese que durante un *soft handover* el MT recibe dos flujos de PDUs y éstos no necesariamente deben estar sincronizados en lo que se refiere a los números de secuencia.

Capa de Aplicación

La capa de Aplicación oculta los aspectos más específicos de las capas inferiores de la red experimental, permitiendo el soporte de cualquier tipo de aplicación multimedia convencional que funcione sobre *sockets*. Para ello, la aplicación multimedia escogida se lanza sobre un socket local con dirección *loopback* al que está asociado la capa de Aplicación.

Concretamente, las funciones de la capa de Aplicación son:

- Adecuar las tramas que llegan desde la aplicación multimedia a los requerimientos de otras capas de la red diseñada. Para ello se ha diseñado una subcapa de segmentación y reensamblado.
- Establecer y liberar conexiones tanto a petición del servidor como del móvil.

Ejemplo de Funcionamiento del Conjunto

Los pasos que se siguen para la ejecución de un *handover* típico son:

- El MT monitoriza el nivel de potencia de la señal recibida desde el AP actual y desde los APs candidatos.
- Cuando se cumplen unas condiciones determinadas, el MT establece una conexión de control con

el AP candidato solicitando el *handover* y transfiriéndole la información necesaria para su correcta ejecución.

- El AP candidato solicita su asociación como hoja al árbol *multicast* que soporta la conexión de usuario.
- Cuando la asociación ha terminado con éxito, el AP candidato comienza a transferir información al MT. En este instante, el MT recibe dos flujos multimedia, uno a través del AP actual y otro a través el AP candidato. También se activan en este instante las funciones de sincronización asegurando a la aplicación multimedia el correcto secuenciamiento de tramas de usuario, sin duplicaciones ni pérdidas.
- Cuando se alcanzada la correcta sincronización entre los dos flujos multimedia, el MT solicita al AP actual que se disocie del árbol *multicast*, pasando, por tanto, el AP candidato a ser el AP actual.
- El ahora AP antiguo inicia el proceso de disociación del árbol *multicast*. Cuando éste concluye, termina el *handover*.

6 Medidas Realizadas

Se han realizado dos tipos de medidas sobre la red ATM experimental. Por una parte, se ha evaluado el impacto que los *handovers* tienen sobre el tráfico multimedia midiendo el *jitter* de trama percibido por el MT. Esta medida se ha realizado mientras el móvil recorría una trayectoria pre-programada y realizaba diferentes *handovers*. Los resultados de éstas medidas se describen con detalle en [8].

También se ha evaluado el tiempo medio de duración del proceso de *handover*. Este tiempo es la suma de cuatro componentes:

- El tiempo de establecimiento de un canal de control entre el MT y el AP candidato.
- El tiempo de establecimiento de un canal para soportar el tráfico de usuario entre el MT y el AP candidato.
- El tiempo necesario para que el AP candidato se asocie al árbol de la conexión *multicast* de usuario.
- El tiempo necesario para alcanzar la sincronización entre los flujos que le llegan al MT desde el AP actual y el candidato.

Se ha medido que el tiempo de ejecución de un *handover* es en media de 213 ms., un tiempo comparable a los que se obtiene en un sistema DECT y considerablemente menor al que se obtiene en GSM. Los escenarios definidos y los procedimientos utilizados para la obtención de estas medidas se describen con detalle en [8].

7 Conclusiones y Trabajo Futuro

Se ha implementado una red experimental para evaluar el impacto que los *handovers* tienen sobre el tráfico multimedia en un entorno de red troncal ATM y red de acceso inalámbrico.

La solución diseñada para el *handover* en la parte de red fija está basada en el uso del procedimiento *Leaf Initiated Join* (LIJ), que permite que un AP pueda asociarse autónomamente a una conexión punto a multipunto en curso.

La solución diseñada para el *handover* en la parte de red de acceso se basa en el uso de las siguientes técnicas: proceso iniciado por el MT, *forward handover* y *soft handover*.

Las medidas realizadas para el tiempo entre tramas recibidas por el MT permiten asegurar que el *jitter* introducido por proceso de *handover* diseñado es despreciable. Además, la duración total del *handover* es comparable a la que se obtiene en otros sistemas reales.

Las líneas de trabajo futuras se centrarán en la implementación y evaluación de otras soluciones para el proceso de *handover*.

Referencias

- [1] P. Nieminem, et al., "Performance Analysis of MASCARA DLC Layer Implementation," Proceedings of ACTS summit 98, Rhodes, Greece, June 8-11, 1998.
- [2] A. Kaloxylos, et al., "Mobility Management and Control Protocol for Wireless ATM Networks," to appear in IEEE Network Special Issue on PCS Network Management.
- [3] R.R. Bhat, "Draft baseline test for Wireless ATM Capability Set 1 Specification," ATM Forum/BTD-WATM-01.1., December 1998.
- [4] G. Bautz, "Addressing in WATM Networks," ATM Forum/97-0322, April 1997.
- [5] R.R. Bhat, "Framework for dynamic COS discovery in Wireless ATM," ATM Forum contribution 98-0005, February 1998.
- [6] P. Agrawal, et al., "SWAN: A mobile multimedia wireless network," IEEE Personal Communications, April 1996.
- [7] C-K Toh, "Performance Evaluation of Crossover Switch Discovery Algorithms for Wireless ATM LANs", IEEE International Conference on Computer Communications, InfoCom'96, San Francisco, California, March 1996.
- [8] V. Martínez y V. Casares, "Protocolo de Handover y Sincronización para Comunicaciones Móviles Multimedia," JITEL99, September 1999.

- [9] F. Torrecilla, et al., “*Mecanismo de Red Fija ATM para el Soporte de Handover en Comunicaciones Móviles Multimedia*,” Technical Document TIC96-0680, November 1998.
- [10] J.R. Vidal, et al., “*Emulación de Radiocanal WATM en una Red Experimental*,” Technical Document TIC96-0680, November 1998.

Macro-Diversidad con Combinación Postdecodificación

Mónica Gorricho Moreno, Josep Paradells Aspas
Universitat Politècnica de Catalunya.
C/. Jordi Girona, 1-3. Módulo C3. Campus Nord. Barcelona 08034.
Telf: 93-4016013, Fax: 93-4015981
e-mail: telmgm@mat.upc.es

Abstract

In this paper, the authors analyse a macro-diversity scheme, which relies on post-detection combining, as opposed to traditional macro-diversity techniques, which are selection based. It uses the information from all base stations that form the diversity set, once it has passed the decoder, so the bits will enter the combiner and they will be weighted and summed to minimize the probability of an incorrect decision. The average BER improvement and the average residual BER as compared to the selection-based macro-diversity scheme are evaluated. The results of the proposed combining scheme are always between the results of the selection and an ideal maximal ratio combining schemes.

1. Introducción

Las comunicaciones móviles están sujetas a constantes variaciones en el nivel de señal recibido como consecuencia de la superposición de los desvanecimientos. La degradación de los niveles de señal se puede combatir utilizando técnicas de diversidad que requieren varios caminos de transmisión de señal llamados ramas de diversidad que transportan la misma información pero que experimentan desvanecimientos incorrelados, por lo que se pueden combinar o seleccionar la mejor de ellas con el fin de recuperar la información de modo más fiable que si ésta se recibe por un único camino. Existen dos tipos básicos de diversidad, (i) la micro-diversidad que consiste en la utilización de varias ramas de diversidad ubicadas en una misma estación base, y (ii) la macro-diversidad en la que las diferentes ramas de diversidad están en estaciones base distantes.

Para el caso de micro-diversidad, además de la posibilidad de seleccionar la rama con mejor calidad, existen varias técnicas de combinación, como son la combinación de igual ganancia y la combinación de ratio máximo que utilizan la información proveniente de todas las ramas de diversidad [1]. Para el caso de macro-diversidad, debido a la separación existente entre las diferentes estaciones base, es difícil utilizar las técnicas de combinación de ratio máximo o de igual ganancia que se aplican en micro-diversidad, ya que para poder llevar a cabo dichas combinaciones es necesario que todas las señales estén en fase, requerimiento difícil de conseguir en el caso de estaciones base distantes. Por ello, el mecanismo de macro-diversidad que se cita en la literatura tradicionalmente es la selección [2].

En este artículo se propone un mecanismo de combinación de las señales recibidas por las diferentes ramas de diversidad, que es aplicable a

esquemas de micro-diversidad, pero también a esquemas de macro-diversidad. El esquema de combinación propuesto está basado en la utilización de la información proveniente de las diferentes ramas (igual que sucede en los esquemas de combinación de micro-diversidad) una vez que ésta ya ha pasado por la etapa de decodificación, por tanto la información disponible será en forma de bits y no será necesario que las diferentes señales estén en fase para combinarlas.

El esquema propuesto minimiza la probabilidad de decidir erróneamente y para ello, se toman los bits que proporciona el algoritmo de decodificación para cada una de las ramas de diversidad y se combinan mediante una ponderación y un sumado de los mismos. Con el resultado de la combinación se decide si el bit transmitido en cada instante fue un *cero* o un *uno*. Para tomar una decisión es necesario disponer de un umbral. Si se quiere que dicho umbral sea independiente del número de ramas de diversidad de que se dispone así como de los factores de ponderación utilizados en cada rama, conviene realizar una transformación de la información a la salida del decodificador, de modo que si el bit recibido, b , toma valores *cero* o *uno*, éste se modifica como $a = 2 \cdot b - 1$ de modo que tomará valores *más o menos uno* y por tanto el umbral de decisión se situará en el *cero* y será independiente tanto del número de ramas de diversidad como de los factores de ponderación utilizados.

2. Combinación

Si se considera que se dispone de tres ramas de diversidad y el factor de ponderación de todas las ramas es el mismo, por ejemplo la unidad, a la salida del decodificador se dispone de

$$a = a_1 + a_2 + a_3 \quad (1)$$

donde a_i son los “bits transformados” de cada una de las tres ramas de diversidad y a es el resultado de la combinación.

Para decidir que se transmitió un *cero* es necesario que $a < 0$, y para ello como mínimo dos de los “bits transformados” recibidos deben ser negativos, mientras que si se recibieron dos o más valores positivos se decidirá un *uno*. Por tanto, se dispone de un mecanismo de decisión por mayoría. Ahora bien, la decisión por mayoría no siempre es la óptima. Si las tres ramas de diversidad tienen una calidad semejante convendrá decidir por mayoría, mientras que si la calidad de una de las ramas es mucho mejor que las otras dos, convendrá seleccionar la información procedente de aquella rama. Por tanto, es necesario buscar unos factores de ponderación diferentes de la unidad y convendrá que estén relacionados con la calidad de cada una de las ramas de diversidad.

2.1. Combinación Óptima

La idea es similar a la que propone Haas en [3], pero en el mecanismo de combinación postdecodificación no se establecen una serie de criterios y en base a estos se decide el bit a la salida, sino que mediante la ponderación por los factores adecuados y el posterior sumado de la información se minimiza automáticamente la probabilidad de decidir erróneamente.

Si se dispone de tres ramas de diversidad, el mecanismo propuesto realizará, como se verá a continuación, o bien una selección de la mejor de las ramas o una decisión por mayoría de los bits de la entrada.

Al igual que en el caso anterior los “bits transformados” que llegan por cada una de las ramas se denominan a_1 , a_2 y a_3 , y se ponderarán respectivamente por los factores k_1 , k_2 y k_3 . En este caso, se producirá un error en los supuestos que aparecen en la Tabla 1, en la que E significa que se ha producido un error y N que no se ha producido.

En algunas combinaciones de errores y factores de ponderación aparece una cierta indeterminación y por tanto se producirá un error en la mitad de los casos. Esos casos son los que están marcados con un asterisco en la Tabla 1.

Con esta información se puede calcular la probabilidad de error equivalente a la salida del combinador. Para ello se utilizará la nomenclatura de la Tabla 2 con el fin de simplificar la notación para calcular dicha probabilidad de error equivalente.

Mediante manipulación algebraica, se llega a la nueva probabilidad de error a la salida del combinador que es la siguiente

$$p_e = (1 - A + B - C + D - E + F) \cdot \left[(p_{e1}p_{e2}p_{e3}) - \frac{1}{2}(p_{e1}p_{e2} + p_{e1}p_{e3} + p_{e2}p_{e3}) \right] + \frac{1+F-E}{2}p_{e1} + \frac{1+D-C}{2}p_{e2} + \frac{1+B-A}{2}p_{e3} \quad (2)$$

Los posibles casos extremos que se pueden estudiar son aquellos en los que las relaciones entre las probabilidades de las diferentes ponderaciones reciben los valores que aparecen en la Tabla 3.

A continuación se estudiarán los valores de probabilidad de error equivalente para estos dos casos y se extraerán conclusiones, en función de los resultados obtenidos. En el primero de los casos, si se sustituye $A = C = E = 1$ en la fórmula (2) se obtiene

$$p_e = p_{e1}p_{e2} + p_{e1}p_{e3} + p_{e2}p_{e3} - 2p_{e1}p_{e2}p_{e3} \quad (3)$$

que es el mismo resultado que se obtendría si se realizase una decisión por mayoría con tres ramas de información.

Tabla 1: Errores a la salida del mecanismo de combinación de bits.

a_1	a_2	a_3	Condición	
E	E	E		
E	E	N	$k_1+k_2>k_3$	
E	N	E	$k_1+k_3>k_2$	
N	E	E	$k_2+k_3>k_1$	
E	N	N	$k_1>k_2+k_3$	
N	E	N	$k_2>k_1+k_3$	
N	N	E	$k_3>k_1+k_2$	
E	E	N	$k_1+k_2=k_3$	*
E	N	E	$k_1+k_3=k_2$	*
N	E	E	$k_2+k_3=k_1$	*
E	N	N	$k_1=k_2+k_3$	*
N	E	N	$k_2=k_1+k_3$	*
N	N	E	$k_3=k_1+k_2$	*

Tabla 2: Nomenclatura de las relaciones entre las probabilidades de los diferentes factores de ponderación.

$P(k_1 + k_2 > k_3)$	A
$P(k_3 > k_1 + k_2)$	B
$P(k_1 + k_3 > k_2)$	C
$P(k_2 > k_1 + k_3)$	D
$P(k_2 + k_3 > k_1)$	E
$P(k_1 > k_2 + k_3)$	F
$P(k_1 + k_2 = k_3)$	$I-A-B$
$P(k_1 + k_3 = k_2)$	$I-C-D$
$P(k_2 + k_3 = k_1)$	$I-E-F$

Tabla 3: Casos extremos de relaciones entre ponderaciones.

A	B	C	D	E	F
1	0	1	0	1	0
1	0	1	0	0	1

En el segundo caso, si se sustituyen en la fórmula (2) los valores $A = C = F = I$ la probabilidad de error equivalente es el resultado de tener una selección de la rama asociada con p_{e1} . Si se estudia más detalladamente este caso, se observa que, dado que se ponderará más aquella rama de diversidad con mejor calidad, y dado que para que se cumpla que $A = C = F = I$ la ponderación de la primera rama tiene que ser mucho mayor que la de las otras dos. Se puede concluir que en este segundo caso se está realizando una selección de la rama con mejor calidad. Por tanto, en los extremos, se estará decidiendo mediante una selección de la mejor de las ramas o bien mediante una decisión por mayoría, y la combinación óptima será aquella que permita realizar esta distinción.

2.2. Factores óptimos de ponderación

Una vez observados los casos extremos, el siguiente paso es conseguir minimizar la probabilidad de error. Es decir, si al seleccionar la mejor de las ramas de diversidad la probabilidad de error resultante es menor que la obtenida por decisión por mayoría, hay que tomar selección, en caso contrario, se decidirá por mayoría. Por consiguiente, se deben buscar los factores de ponderación óptimos que en este proceso siempre proporcionen el valor mínimo entre la probabilidad de error que se obtendría por selección y la que se obtendría por mayoría. La expresión matemática equivalente será

$$p_e = \min \left\{ \begin{array}{l} \min(p_{e1}, p_{e2}, p_{e3}) \\ p_{e1}p_{e2} + p_{e1}p_{e3} + p_{e2}p_{e3} - 2p_{e1}p_{e2}p_{e3} \end{array} \right. \quad (4)$$

Para calcular los factores de ponderación óptimos es necesario minimizar la probabilidad de error y para ello se debe realizar un estudio previo de las relaciones entre las probabilidades de error de las diferentes ramas, p_{e1} , p_{e2} y p_{e3} , con el fin de que la probabilidad de error equivalente, p_e , siempre sea mínima, teniendo en cuenta la expresión (4).

En aquellos casos en los que las dos posibilidades de (4) sean iguales, será equivalente tomar selección o decisión por mayoría, es decir

$$\min(p_{e1}, p_{e2}, p_{e3}) = p_{e1}p_{e2} + p_{e1}p_{e3} + p_{e2}p_{e3} - 2p_{e1}p_{e2}p_{e3} \quad (5)$$

Si se supone, sin pérdida de generalidad, que la probabilidad de error de la rama *uno* es menor que la probabilidad de error de las otras dos ramas, se tiene

$$\min(p_{e1}, p_{e2}, p_{e3}) = p_{e1} \quad (6)$$

Si se sustituye (6) en (5), se obtiene la siguiente relación entre las probabilidades de error de las diferentes ramas

$$p_{e1} = p_{e1}p_{e2} + p_{e1}p_{e3} + p_{e2}p_{e3} - 2p_{e1}p_{e2}p_{e3} \quad (7)$$

$$p_{e1} = \frac{p_{e2}p_{e3}}{1 + 2p_{e2}p_{e3} - p_{e2} - p_{e3}} \quad (8)$$

La igualdad (8) indica la relación entre las probabilidades de error de las diferentes ramas para que sea equivalente decidir por mayoría o seleccionar la mejor de las ramas de diversidad, ya que ambos casos proporcionan la misma probabilidad de error a la salida del combinador. Pero interesa conocer en qué casos es conveniente decidir por mayoría, y en qué casos conviene realizar una selección.

Si

$$p_{e1} > \frac{p_{e2}p_{e3}}{1 + 2p_{e2}p_{e3} - p_{e2} - p_{e3}} \quad (9)$$

se debe decidir por mayoría para obtener la mínima probabilidad de error, y en caso contrario, se debe seleccionar la información proveniente de la rama *uno* (la que se había supuesto que tenía menor probabilidad de error de todas).

Dado que los factores de ponderación utilizados en el combinador determinarán la decisión por mayoría o la selección y debido a que el combinador realiza una suma ponderada de la información que aparece en las distintas ramas de diversidad, conviene transformar la relación entre las probabilidades de error en una relación entre los factores de ponderación.

Por tanto, hay que buscar una relación equivalente para los factores de ponderación. Para ello, es necesario que el producto de probabilidades de error se convierta en suma de ponderaciones. Una función matemática que permite transformar productos en sumas es el logaritmo en cualquier base, por lo que se realizará la siguiente manipulación matemática

$$\begin{aligned} -\log(p_{e1}) &= -\log\left(\frac{p_{e2}p_{e3}}{1 + 2p_{e2}p_{e3} - p_{e2} - p_{e3}}\right) \\ -\log(p_{e1}) + \log(1 + 2p_{e2}p_{e3} - p_{e2} - p_{e3}) &= \\ = -\log(p_{e2}p_{e3}) + 2\log(1 + 2p_{e2}p_{e3} - p_{e2} - p_{e3}) & \quad (10) \end{aligned}$$

y por tanto, se puede considerar como ponderación,

$$\begin{aligned} k_1 &= -\log(p_{e1}) + \log(1 + 2p_{e2}p_{e3} - p_{e2} - p_{e3}) \\ k_2 &= -\log(p_{e2}) + \log(1 + 2p_{e2}p_{e3} - p_{e2} - p_{e3}) \\ k_3 &= -\log(p_{e3}) + \log(1 + 2p_{e2}p_{e3} - p_{e2} - p_{e3}) \end{aligned} \quad (11)$$

donde se supuso inicialmente que p_{e1} era la menor de las tres probabilidades de error.

Con estas ponderaciones se consigue minimizar la probabilidad de decidir erróneamente, que es el objetivo que se pretendía.

3. Resultados

Si se consideran tres ramas de diversidad y que todas ellas tienen la misma probabilidad de error media, se tienen los resultados de las Figs. 1 y 2. En estas

figuras se compara el mecanismo de selección, el mecanismo de ponderación propuesto con los factores de ponderación óptimos y un posible mecanismo de ratio máximo ideal en el que la relación señal a ruido resultante es la suma de las relaciones señal a ruido de las diferentes ramas. Se espera que el mecanismo de ratio máximo presente mejores resultados, es decir, menor probabilidad de error a la salida, mientras que el que tenga mayor probabilidad de error sea el de selección. El combinador, al tener todas las probabilidades de error medias iguales decidirá por mayoría, y por tanto, la probabilidad de error resultante estará entre las otras dos.

Para relacionar, al realizar las simulaciones, la relación señal a ruido con la probabilidad de error, es necesario elegir una modulación. Por simplicidad, se tomó la DPSK con demodulación no-coherente, para la cual, la relación entre la probabilidad de error de bit y la relación señal a ruido es la siguiente,

$$P_b = \frac{1}{2} \exp(-SNR) \quad (12)$$

Para otras modulaciones el comportamiento de los resultados es similar y no se incluyen aquí.

Los valores de BER residual obtenidos por simulación se representan en la Fig. 1, donde se aprecia el comportamiento que se había predicho para los valores medios de probabilidad de error a la salida del combinador. Si se calcula la ganancia de los mecanismos de combinación respecto a los resultados que se obtienen por selección, donde la ganancia se define como

$$ganancia = \log_{10} \left(\frac{BER_{SELEC}}{BER_{MÉTODO}} \right) \quad (13)$$

se obtienen los valores de ganancias representados en la Fig. 2, donde un valor de ganancia de 1 significa que hay un orden de magnitud de diferencia entre el método comparado y el de selección.

Si se considera ahora que a las tres ramas de diversidad no llega la misma relación señal a ruido, los resultados serán peores, ya que no es el mejor de los casos, aquel en el que se puede obtener una mayor aportación al combinar las diferentes señales recibidas. En este caso, si la diferencia de relación señal a ruido entre las distintas ramas es relativamente importante, el mecanismo de combinación propuesto, que minimiza la probabilidad de decidir erróneamente, proporciona los mismos resultados que el mecanismo de selección, ya que al tener una de las ramas con calidad considerablemente mejor que las otras dos, se seleccionará la información de esta, y los resultados serán peores que los que se obtendrían con combinación de ratio máximo (los resultados obtenidos no se presentan aquí).

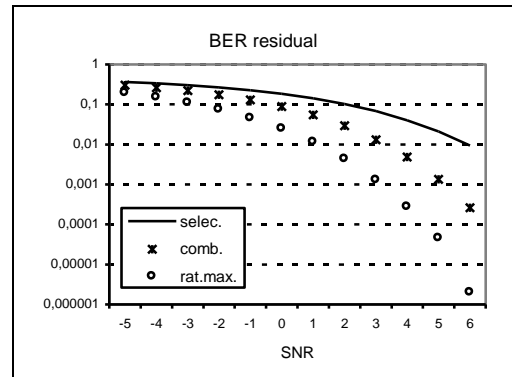


Figura 1: BER residual. Igual probabilidad de error en todas las ramas.

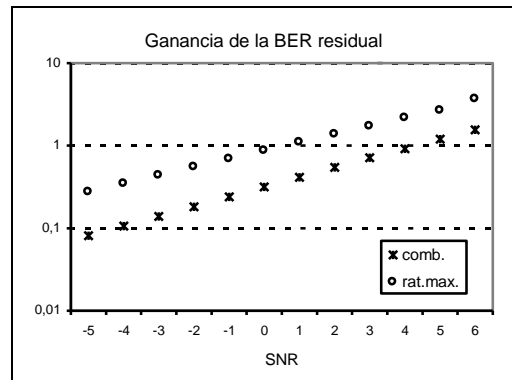


Figura 2: Ganancia de los mecanismos de combinación respecto a la selección.

Dado que se propone este mecanismo para aplicarlo en esquemas de macro-diversidad, a continuación se presenta un caso más realista en el que un terminal móvil se comunica con tres estaciones base distantes, de modo que dicho terminal móvil se desplaza por el área de la celda de una de las estaciones base y se conectará con las tres más 'cercanas' como se aprecia en la Fig.3. Por tanto aparecerán atenuaciones por distancia y también se considerarán las atenuaciones *log-normales*, que en las simulaciones tienen una desviación típica de 8 dB.

En este nuevo caso, las relaciones señal a ruido son más dispares que en los dos casos presentados anteriormente, por ello, lo único que se puede avanzar es que los valores obtenidos de BER residual estarán entre los que se obtendrían por selección y los obtenidos por ratio máximo ideal, ya que no es posible predecir a priori qué relación existirá entre las diferentes relaciones señal a ruido de las distintas ramas de diversidad. En las Figs. 4 y 5 se representan los valores de BER residual y ganancias respecto al esquema de selección obtenidos para una atenuación *log-normal* con $\sigma=8dB$. En la Fig. 4 se aprecia que debido a que los móviles estarán más próximos a una de las estaciones base que a las otras dos, en el caso de combinación de bits, se obtienen unos resultados muy similares a los de selección, y por tanto las ganancias (Fig.5) son inferiores a las que se obtenían en la Fig. 2.

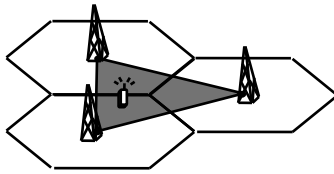


Figura 3: Conexión mediante mecanismo de macrodiversidad de un terminal móvil a las tres estaciones base más ‘cercanas’.

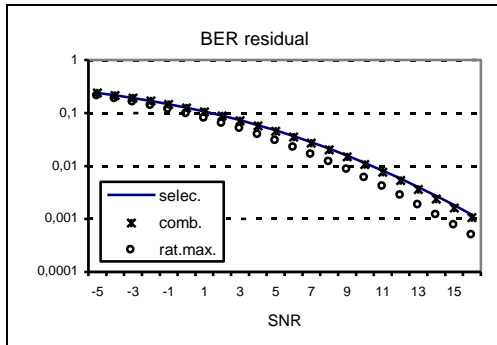


Figura 4: BER residual. Móvil se desplaza por el área de la celda. Con atenuaciones *log-normales*.

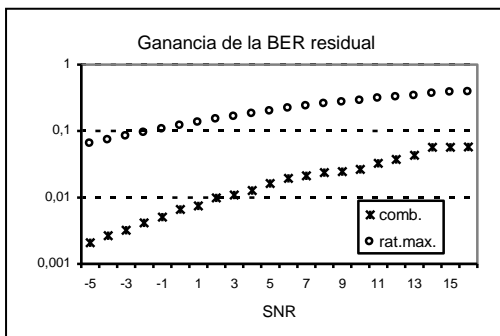


Figura 5: Ganancia de los mecanismos de combinación respecto a la selección.

4. Combinación de símbolos

Si partiendo de los factores de ponderación óptimos que se obtuvieron para el caso de combinación de bits postdecodificación, se realiza una ponderación equivalente de símbolos en lugar de combinar bits, los resultados son considerablemente mejores que para el caso de bits, ya que un bit es un símbolo que ha pasado por un cuantificador de dos niveles, por lo que los símbolos aportan más información que los bits.

Los resultados obtenidos mediante simulación para el caso de símbolos recibidos a través de un canal blanco Gaussiano utilizando los factores de ponderación de (11) son los que se presentan en la Fig. 6 para BER residual y en la Fig. 7 para los valores de ganancias obtenidos. Las simulaciones se han realizado considerando que el terminal móvil se desplaza por el área de la celda, por lo que hay atenuaciones por distancia y *log-normales*. Los resultados que se obtuvieron para la combinación de bits eran muy similares a los que se obtenían por selección, sin embargo, si se combinan símbolos, los resultados están muy próximos a los que se obtienen

con la combinación de ratio máximo (en la Fig. 6 se solapan las dos curvas), y por tanto, hay una ganancia importante respecto al mecanismo de selección.

5. Conclusiones

La idea básica de la combinación postdecodificación es no sólo seleccionar los bits de aquella rama de diversidad que proporcione menor probabilidad de error, sino intentar aprovechar al máximo la información procedente de todas las ramas de diversidad, con el fin de que la probabilidad de error resultante sea mínima.

El mecanismo de combinación propuesto, minimiza la probabilidad de decidir erróneamente, pero sin la necesidad de establecer criterios de trabajo diferentes y seleccionar en cada momento uno de ellos para posteriormente determinar los bits a la salida del elemento decisor.

En lugar de establecer dichos criterios, se utilizan unos factores de ponderación que multiplicados por los *bits* que proporciona el decodificador y sumados, permitirán decidir si la información recibida es un *uno* o un *cero*. Se han calculado los factores de ponderación óptimos que minimizan la probabilidad de decidir erróneamente.

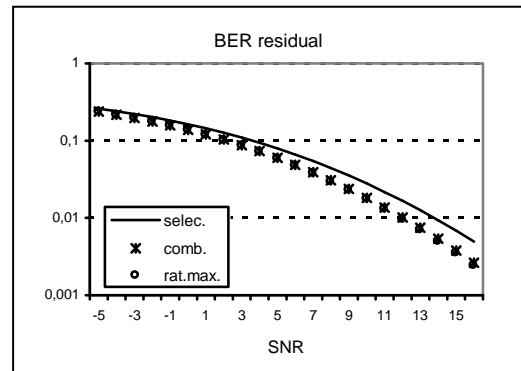


Figura 6: BER residual. Móvil se desplaza por área de celda. Atenuaciones *log-normales*. Combinación de símbolos.

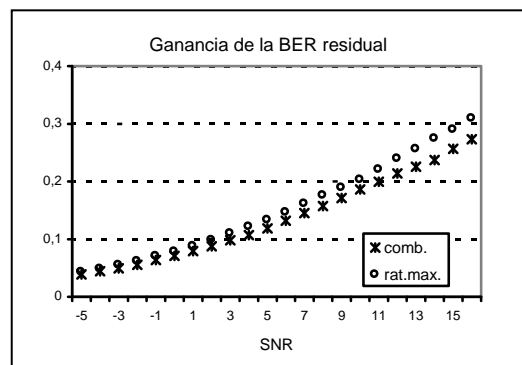


Figura 7: Ganancia de los mecanismos de combinación respecto a la selección. Móvil se desplaza por área de celda. Atenuaciones *log-normales*. Combinación de símbolos.

Los mejores resultados se obtienen cuando los valores medios de las probabilidades de error de las diferentes ramas de diversidad son similares, al igual que sucede con cualquier mecanismo de combinación aplicable (por ejemplo, un hipotético mecanismo de combinación ideal de ratio máximo).

Las ganancias con respecto al caso de selección (único mecanismo aplicable en macro-diversidad) son considerables y hay que destacar que el esquema propuesto es implementable tanto en micro-diversidad, como en macro-diversidad y sólo requiere una estimación de las probabilidades de error en las diferentes ramas.

Si se aplica el mecanismo sobre símbolos en lugar de sobre bits utilizando los mismos factores de ponderación que se utilizaron para el caso de bits, los resultados obtenidos están muy próximos a los que se obtienen con un hipotético esquema de diversidad con combinación de ratio máximo ideal, con la ventaja que la combinación propuesta no requiere que todas las señales estén en fase y por tanto es aplicable a esquemas de macro-diversidad mientras que ratio máximo no lo es.

Por tanto, la combinación propuesta no sólo es aplicable al caso de disponer de bits, sino que proporciona muy buenos resultados al combinar símbolos.

Agradecimientos

Este trabajo ha sido financiado por el proyecto nacional CICYT TIC 97-0942.

Referencias

- [1] W. C. Jakes. "Microwave Mobile Communications". IEEE Press. New York, Wiley & Sons, 1974.
- [2] W.C.Y. Lee. "Mobile Communications Engineering". Mc-Graw-Hill Inc. 1982.
- [3] Z. J. Haas, C. P. Li, "Macrodiversity Technique for Improvement in BER in Wireless Systems", Electronic Letters, Vol. 33, No. 7, March 1997, pp. 556-57.

Análisis de Throughput y Retardo en un Sistema CDMA Celular de Paquetes con Control de Potencia Imperfecto

Juan Manuel Romero Jerez, Margarita Ruiz García, Antonio Díaz Estrella
Departamento de Tecnología Electrónica
E.T.S.I. Telecomunicación, Universidad de Málaga
Campus Teatinos S/N, 29071 Málaga
Telf: 952 13 28 74, Fax: 952 13 14 47
e-mail: romero@dte.uma.es

Abstract

An analytical approach for the calculation of throughput and delay in a cellular slotted CDMA (Code Division Multiple Access) system is presented. Power control is essential for such systems for good performance, however, perfect power control is hard to achieve in real systems. Usually, imperfections in power control after despreading are modelled as a log-normal random variable. It has been shown that the standard deviation of this variable is a function of the mobile's velocity and of whether or not the mobile is communicating with the base station where the power is measured. In this work, we show that when this fact is not taken into account the results obtained are too optimistic. We use the error statistics to model the intercell interference and evaluate the impact of different mobile's velocity and number of resolvable paths on the system performance.

1. Introducción

Durante la última década se ha llevado a cabo un tremendo esfuerzo en la investigación de sistemas CDMA (Code Division Multiple Access) y el análisis de tráfico de dichos sistemas, y se ha comprobado que esta técnica permite soportar una carga de tráfico mayor que otras técnicas de acceso múltiple, tales como FDMA (Frequency Division Multiple Access) o TDMA (Time Division Multiple Access). Entre las ventajas de los sistemas CDMA podemos citar una alta eficiencia espectral, posibilidad inherente de aprovechar la diversidad temporal, y robustez frente a las variaciones del canal (para una descripción detallada de los sistemas CDMA véase, por ejemplo, [1]). Además, la tecnología CDMA permite el empleo de técnicas avanzadas de transmisión radio tales como arrays de antenas adaptativos, turbocódigos y detección multiusuario, lo que puede suponer una gran mejora en estos sistemas [2]. Debido principalmente a las características mencionadas, los sistemas CDMA se han revelado como el principal candidato para los sistemas móviles de tercera generación, englobados por la ITU (International Telecommunications Union) bajo el nombre de IMT-2000 (International Mobile Telecommunications) [3].

En este artículo presentamos el análisis de las prestaciones de un sistema CDMA orientado a paquetes. Los sistemas radio móviles de conmutación de paquetes han ido adquiriendo una relevancia cada vez mayor debido en parte al rápido crecimiento de internet y la esperada demanda de acceso a internet desde terminales móviles. Una de las razones del rápido crecimiento de internet ha sido la aparición de los servicios WWW (World Wide Web). Una típica sesión WWW consta de ráfagas de paquetes, que pueden ser de distinta longitud. Estos paquetes pueden segmentarse para su transmisión a través del interfaz aire. Debido a que pueden existir largos periodos de silencio entre ráfagas de paquetes [4] y a la existencia

de unos recursos limitados, los sistemas de paquetes constituyen una elección más apropiada que los de conmutación de circuitos para el interfaz aire.

Es bien conocido que los sistemas CDMA están limitados por interferencia y que sufren el problema cerca lejos, es decir, si todos los móviles transmiten a la misma potencia la señal de los móviles cercanos a la estación base puede ahogar la de los más alejados. Por tanto, en estos sistemas es esencial la existencia de un control de potencia que regule la potencia transmitida por los distintos móviles. Sin embargo, en los sistemas prácticos se producen errores en el control de potencia, de modo que la potencia de la señal recibida de los diferentes móviles no es constante. En [5] se presenta una aproximación analítica para el cálculo del throughput y el retardo en función de las imperfecciones del control de potencia en un sistema CDMA celular de paquetes. Sin embargo, en el mencionado artículo, la carga de tráfico está limitada por la relación señal ruido de las distintas conexiones. Nosotros proponemos un sistema en el que la carga está limitada por la máxima interferencia que puede soportar un sistema CDMA. En [6] se demuestra que esta limitación es más restrictiva y asegura la existencia de solución del algoritmo de asignación de potencia. En [7] un esquema de paquetes CDMA en presencia de desvanecimientos de señal (fading) se analiza considerando una capacidad fija. Esta aproximación no aprovecha la flexibilidad inherente de los sistemas CDMA, en los que el número de usuarios que pueden transmitir simultáneamente en una celda es función de la interferencia en dicha celda. Es más, en los sistemas CDMA, la desviación de la señal debida a errores en el control de potencia depende de si el móvil que genera la señal se encuentra o no en la celda donde se mide la potencia, hecho que afecta de forma importante a la capacidad, y que no se tiene en cuenta en los artículos mencionados. Nosotros proponemos un esquema en el que se

considera este efecto y mostramos que conduce a unos resultados más realistas. Además, estudiamos explícitamente el efecto del fading en términos de la velocidad de los móviles y del número de componentes de la señal recibida debida a la recepción múltiple provocada por reflexiones de la señal (multicamino).

Este artículo se organiza de la siguiente manera. El modelo de sistema se presenta en la sección 2. El cálculo de la probabilidad de superar la interferencia máxima permitida se realiza en la sección 3. El análisis de las prestaciones del sistema se presenta en la sección 4. En la sección 5 se muestran los resultados numéricos obtenidos. Por último, las conclusiones se presentan en la sección 6.

2. Modelo de sistema

Consideraremos el canal ascendente (desde los móviles hacia la estación base) de un sistema celular CDMA de paquetes con división dúplex de frecuencia, es decir, los canales ascendente y descendente (desde la estación base hacia los móviles) se asignan a bandas de frecuencia diferentes. No tendremos en cuenta sectorización de las celdas, sin perder por ello generalidad, y consideraremos que el sistema no está limitado por el número de receptores disponibles (correladores) En el sistema propuesto, los móviles pueden transmitir paquetes del mismo tamaño en ventanas de tiempo predeterminadas (slots). Como simplificación, consideraremos un sistema con control de potencia, aunque éste es difícil de realizar debido a la corta duración de los paquetes. Asumiremos un canal con fading Rayleigh donde la señal recibida tiene M componentes del mismo nivel de potencia. Por tanto, la potencia de la señal de fading x tendrá una distribución chi-cuadrado con $2M$ grados de libertad [8]:

$$P(x) = \frac{M^M}{(M-1)!} x^{M-1} e^{-Mx} \quad (1)$$

Debido a que en sistemas reales se producen errores en el control de potencia, el nivel de las señales recibidas de cada usuario sufrirá una desviación respecto al valor medio. Modelaremos la potencia recibida en una estación base debida a cada conexión con una variable aleatoria con distribución logarítmico-normal. Entre otros factores, el nivel medio de la señal recibida es función de la velocidad de los móviles [9], y es ajustado por el mecanismo de control de potencia. En un canal con fading, la desviación estándar del nivel de potencia recibido es función de la velocidad de los móviles y de si el móvil se comunica con la estación base donde se mide la potencia o con otra estación base del sistema [10]. Cuando el control de potencia es lo suficientemente rápido para seguir las variaciones de señal provocadas por el fading, las señales procedentes de los móviles que están transmitiendo a una estación base distinta de

la considerada sufren una desviación mayor, ya que la potencia transmitida por estos móviles está tratando de mitigar el efecto del fading en otra estación base. Por supuesto, esto implica que los procesos de fading sean independientes para estaciones bases distintas.

Emplearemos los resultados obtenidos en [10] para modelar el efecto del control de potencia imperfecto. Estos resultados se presentan en la Tabla 1, donde se muestra la desviación estándar, en decibelios, de la señal recibida de usuarios comunicándose con la estación base considerada, σ_s , o con otra estación base, σ_o . Se presentan resultados para distintas velocidades de los móviles y diferente número de componentes de señal recibida.

V(Km/h)	M=2		M=3	
	σ_s dB	σ_o dB	σ_s dB	σ_o dB
5	0.35	4.90	0.33	3.90
20	1.40	4.40	0.94	3.67
40	2.20	3.41	1.75	3.25

Tabla 1. Desviación estándar de los móviles comunicándose con la estación base considerada y con una estación base distinta a la considerada.

3. Cálculo de la probabilidad de superar la interferencia máxima permitida.

En [11] se establece que la carga máxima de tráfico en el canal ascendente de un sistema CDMA se alcanza cuando el conjunto de usuarios que transmiten simultáneamente introducen una densidad de interferencia en el sistema, I_0 , que excede al ruido de fondo, N_0 , en una cantidad $1/\eta_0$. Para un sistema con una sola celda, esto se puede expresar:

$$Z \equiv \sum_{i=1}^N \rho_i > \frac{W}{R_b} (1 - \eta_0) \quad (2)$$

donde N es el número de usuarios transmitiendo simultáneamente en la ventana de tiempo considerada, W/R_b es la ganancia de proceso, $\rho_i = E_{bi}/I_0$ es la relación energía por bit a densidad de interferencia para el usuario i , y $\eta_0 = N_0/I_0$ es un parámetro que indica la carga del sistema. Las medidas de propagación realizadas indican en general [11] que los niveles de potencia recibidos siguen una distribución gaussiana cuando se miden en dB. por lo que los ρ_i , cuando son medidos en dB tienen una distribución normal con media m_i y desviación estándar σ_i . Por tanto, la variable aleatoria ρ_i , cuando se mide en vatios, sigue una distribución logarítmico-normal con media y varianza dadas por:

$$E[\rho_i] = e^{\beta m_i + \frac{1}{2} \beta^2 \sigma_i^2} \quad (3)$$

$$Var(\rho_i) = e^{2\beta m_i + \beta^2 \sigma_i^2} (e^{\beta^2 \sigma_i^2} - 1) \quad (4)$$

donde $\beta = (\ln 10)/10$.

Nuestro objetivo es calcular la probabilidad de superar la interferencia máxima permitida, que viene dada por:

$$P_{out} = P\left(Z > Z_0 = \frac{W}{R_b} X_0\right) \quad (5)$$

donde $X_0 = 1 - \eta_0$ se define como el factor de carga y da una medida de la congestión del sistema. Cuando el sistema está completamente cargado, el factor de carga es uno. Debido a que un sistema totalmente cargado es inestable y puede conducir a todos los usuarios a transmitir a su máxima potencia posible (véase el capítulo 7 en [3]), se requiere un margen de seguridad. En sistemas prácticos el factor de carga debe ser del orden de 0.5 a 0.9.

Debido a que Z es la suma de variables aleatorias con distribución logarítmico-normal, no podemos conocer con exactitud su función densidad de probabilidad, y por tanto, no podemos conocer el valor exacto de la probabilidad de superar la interferencia máxima permitida. Emplearemos por tanto dos aproximaciones para calcular dicha probabilidad. La primera asume que Z es una variable aleatoria gaussiana aplicando el teorema central del límite. La segunda asume que Z tiene una distribución logarítmico-normal, que proviene del hecho de que una suma finita de variables aleatorias con distribución logarítmico-normal puede aproximarse por una variable aleatoria con esa misma distribución [12].

En primer lugar, calcularemos la media y la varianza de Z , que para un sistema con una única celda es la suma de N variables aleatorias, las cuales se asumen independientes. Debido a que estamos considerando un sistema con múltiples celdas, debemos considerar la interferencia recibida debida a los móviles situados en celdas diferentes a aquella donde analizamos el sistema (interferencia intercelda). Un aspecto importante cuando analizamos sistemas CDMA multicelda es el modo en que modelamos dicha interferencia intercelda (una amplia revisión de los métodos más comúnmente utilizados puede encontrarse en [13]). Es habitual asumir que los usuarios del resto de celdas añaden una interferencia equivalente a fN usuarios transmitiendo en la celda considerada, donde f es el factor de interferencia y se define como el cociente entre la interferencia intercelda e intracelda, (la generada por los móviles en la propia celda). El cálculo del factor f para el caso de que no exista fading puede encontrarse en [14]. Suele asumirse igualmente que estos fN usuarios añaden una interferencia, modelada por una variable aleatoria con distribución logarítmico-normal, con la misma desviación estándar que los N usuarios de la celda considerada. Sin embargo, como ya hemos mencionado, la desviación de la señal recibida depende de en qué celda se encuentra el usuario que interfiere, es decir de qué estación base está

controlando la potencia transmitida. Por tanto, la media y la varianza de Z vendrán dadas por:

$$E[Z] = \sum_{i=1}^N E[\rho_{s_i}] + \sum_{i=1}^{fN} E[\rho_{o_i}] \quad (6)$$

$$Var(Z) = \sum_{i=1}^N Var(\rho_{s_i}) + \sum_{i=1}^{fN} Var(\rho_{o_i}) \quad (7)$$

donde ρ_{s_i} y ρ_{o_i} son variables aleatorias con distribución logarítmico-normal que representan la relación E_{b_i}/I_0 recibida debido a usuarios que originan interferencia intracelda e intercelda respectivamente, y en consecuencia, tienen diferentes desviaciones estándar.

3.1 Aproximación gaussiana

Bajo la aproximación gaussiana, la media y la varianza de Z se identifican como la media y la varianza de una variable aleatoria con distribución normal. Mediante (3), (4), (6) y (7) obtenemos:

$$E[Z] = Ne^{\beta m + \frac{1}{2}\beta^2\sigma_s^2} + fNe^{\beta m + \frac{1}{2}\beta^2\sigma_o^2} \quad (8)$$

$$Var(Z) = Ne^{2\beta m + \beta^2\sigma_s^2} (e^{\beta^2\sigma_s^2} - 1) + fNe^{2\beta m + \beta^2\sigma_o^2} (e^{\beta^2\sigma_o^2} - 1) \quad (9)$$

Y fácilmente se obtiene que la probabilidad de superar la interferencia máxima permitida viene dada por:

$$P_{out} = Q\left(\frac{Z_0 - E[Z]}{\sqrt{Var(Z)}}\right) \quad (10)$$

donde:

$$Q(x) = \frac{1}{2\pi} \int_x^{\infty} e^{-t^2/2} dt \quad (11)$$

3.2 Aproximación logarítmico-normal

Bajo la aproximación logarítmico-normal, la media y la varianza de Z se identifican como la media y la varianza de una variable aleatoria ξ con distribución logarítmico-normal, donde:

$$E[\xi] = e^{m_N + \frac{1}{2}\sigma_N^2} \quad (12)$$

$$Var(\xi) = e^{2m_N + \sigma_N^2} (e^{\sigma_N^2} - 1) \quad (13)$$

Imponiendo que $E[Z] = E[\xi]$ y que $Var(Z) = Var(\xi)$, donde la media y la varianza de Z vienen dadas por (8) y (9) respectivamente, obtenemos un sistema de dos ecuaciones con dos incógnitas:

$$Ne^{\beta m + \frac{1}{2}\beta^2\sigma_s^2} + fNe^{\beta m + \frac{1}{2}\beta^2\sigma_o^2} = e^{m_N + \frac{1}{2}\sigma_N^2} \quad (14)$$

$$\begin{aligned}
Ne^{2\beta m + \beta^2 \sigma_s^2} (e^{\beta^2 \sigma_s^2} - 1) + f Ne^{2\beta m + \beta^2 \sigma_o^2} (e^{\beta^2 \sigma_o^2} - 1) &= \\
&= e^{2m_N + \sigma_N^2} (e^{\sigma_N^2} - 1)
\end{aligned} \tag{15}$$

Resolviendo para m_N y σ_N^2 , obtenemos:

$$\sigma_N^2 = \ln \left[\frac{e^{\beta^2 \sigma_s^2} (e^{\beta^2 \sigma_s^2} - 1) + f e^{\beta^2 \sigma_o^2} (e^{\beta^2 \sigma_o^2} - 1)}{N \left(e^{\frac{1}{2} \beta^2 \sigma_s^2} + f e^{\frac{1}{2} \beta^2 \sigma_o^2} \right)^2} + 1 \right] \tag{16}$$

$$m_N = \ln N + \beta m + \ln \left(e^{\frac{1}{2} \beta^2 \sigma_s^2} + f e^{\frac{1}{2} \beta^2 \sigma_o^2} \right) - \frac{1}{2} \sigma_N^2 \tag{17}$$

En este caso, la probabilidad de superar la interferencia máxima permitida viene dada por:

$$P_{out} = Q \left(\frac{\ln Z_0 - m_N}{\sigma_N} \right) \tag{18}$$

4. Análisis del sistema

Consideramos un sistema con ventanas o ranuras temporales en el que los móviles pueden transmitir únicamente en instantes fijos de tiempo (slots). Los parámetros más apropiados para analizar un sistema de este tipo son el throughput y el retardo. Definimos el throughput como el número medio de paquetes que son recibidos con éxito en un slot, o sea:

$$S = \sum_{k=1}^{\infty} k P_T(k) P_s(k) \tag{19}$$

donde $P_T(k)$ es la probabilidad de que se transmitan k paquetes simultáneamente en un slot. Asumiendo que los paquetes llegan al sistema siguiendo una distribución de Poisson, $P_T(k)$ puede escribirse como:

$$P_T(k) = \frac{G^k}{k!} e^{-G} \tag{20}$$

El tráfico ofrecido G se define como el número medio de transmisiones (incluyendo los paquetes transmitidos por primera vez y los retransmitidos debido a errores) en un slot por k usuarios.

Por otra parte, $P_s(k)$ es la probabilidad de que k paquetes sean recibidos correctamente en un slot. Asumimos que, cuando se supera la máxima carga permitida, todos los paquetes transmitidos en

ese slot son destruidos. Por tanto, $P_s(k)$ vendrá dada por:

$$P_s(k) = 1 - P_{out}(k) \tag{21}$$

siendo $P_{out}(k)$ la probabilidad de superar la interferencia máxima permitida cuando se transmiten k paquetes en un slot., que viene dada por (10) y (18) respectivamente para las aproximaciones gaussiana y logarítmico normal.

El retardo medio se define como el número medio de slots que tarda un paquete en ser recibido correctamente. El retardo medio en slots puede calcularse, asumiendo reconocimiento instantáneo y libre de errores, usando [7]:

$$D = (1.5 + d) + \left[\frac{G}{S} - 1 \right] (\lfloor \delta + 1 \rfloor + 1 + 2d) \tag{22}$$

donde δ es el retardo de transmisión medio y d es el retardo de propagación, normalizados al tiempo de slot. S es el throughput y G es el tráfico ofrecido. Aquí, $\lfloor x \rfloor$ es el mayor entero menor o igual a x .

5. Resultados numéricos.

En este apartado presentamos las gráficas que muestran los resultados obtenidos para las aproximaciones logarítmico-normal (en trazo continuo) y gaussiana (en trazo discontinuo). Throughput y retardo son los parámetros seleccionados para medir las prestaciones del sistema. Para todos los resultados presentados se ha considerado un valor del factor de interferencia f de 0.55. El throughput se mide por celda y el retardo en unidades de slot. Para el cálculo del retardo hemos asumido un retardo de retransmisión medio de $\delta=10$ slots y se ha despreciado el retardo de propagación. Estos parámetros pueden ser modificados sin tener que alterar por ello la aproximación seguida para analizar el sistema. Todas las figuras presentadas en esta sección son válidas para el enlace ascendente.

En Fig. 1 y Fig. 2 se muestran el throughput y el retardo para diferentes velocidades de los móviles cuando el número de componentes de la señal recibida por cada usuario M es 2 y el factor de carga X_0 es 0.9. Los valores de las desviaciones estándar de las potencias recibidas de los móviles en la celda considerada y en el resto de celdas se han tomado de la Tabla 1. Para los usuarios desplazándose a 5,20, y 40 Km/h hemos asumido una relación E_b/I_0 requerida de 3.5, 5 y 5.7 dB respectivamente [9]. Esta diferencia se explica por el hecho de que, a bajas velocidades, el control de potencia es muy efectivo en contrarrestar el efecto del fading, ya que estos usuarios no experimentan el fading rápido que soportan los móviles que se desplazan a mayores velocidades. Debe destacarse, sin embargo, que a mayor velocidad, mayores los beneficios del entrelazado de bit, por

tanto, mayores velocidades no implican necesariamente unos requerimientos de relación E_b/I_0 más altos. De las figuras, es claro que las prestaciones empeoran a medida que la velocidad de los móviles aumenta. Peores prestaciones implican un throughput máximo menor y un retardo mayor para un throughput dado. Estas diferencias se deben a que, a mayor velocidad, mayor la desviación de la señal recibida debido a errores en el control de potencia. También influyen en estas diferencias el hecho de que, a mayor velocidad, menor diferencia relativa entre las desviaciones de señal de los móviles en la celda deseada o en otras celdas, y que móviles desplazándose a distinta velocidad tienen diferentes requerimientos de E_b/I_0 .

La influencia de la velocidad en las prestaciones del sistema cuando el número de componentes de señal M es 3 puede observarse en Fig. 3 y Fig. 4. Si comparamos estas figuras con el caso en el que $M=2$, observamos unas peores prestaciones que para el caso en que M es 3. Estas diferencias de prestaciones se deben a que las desviaciones de las señales recibidas decrecen cuando M aumenta.

En Fig. 5 y Fig. 6 se muestran las prestaciones del sistema considerando que todos los móviles del sistema llegan a la estación base con la misma desviación en el nivel de señal recibido. La velocidad de los móviles es 5 km/h, M es 2 y el factor de carga X_0 es 0.9. Se muestran los casos de control de potencia perfecto e imperfecto, y éstos se comparan con el modelo de interferencia propuesto, según el cual las desviaciones de los niveles de señal dependen de la celda donde se encuentran los móviles. Puede verse que hay poca diferencia entre los casos de control de potencia perfecto e imperfecto, debido a la poca movilidad de los usuarios considerada. Sin embargo, cuando consideramos una desviación diferente para los móviles situados en otras celdas, las prestaciones decrecen de forma significativa. En el caso presentado, el máximo throughput decrece un 30%.

En Fig. 7 y Fig. 8 se presentan el throughput y el retardo para diferentes valores del factor de carga, siendo $M=2$, $v=5$ km/h y $X_0=0.9$. Como era de esperar, las prestaciones empeoran a medida que el factor de carga decrece.

Finalmente, debe mencionarse que se ha apreciado una diferencia muy pequeña entre las aproximaciones gaussiana y logarítmico-normal para los parámetros seleccionados, y en algunos casos las curvas correspondientes a ambas aproximaciones están solapadas.

6. Conclusiones

Se ha presentado el estudio analítico de las prestaciones de un sistema celular CDMA de paquetes sobre un canal Rayleigh en términos de throughput y retardo. Estos sistemas se presentan como una buena elección para la transmisión sobre el interfaz aire de

paquetes para satisfacer servicios futuros tales como acceso móvil a internet. Se han presentado resultados numéricos para las aproximaciones gaussiana y logarítmico-normal del sistema para diferentes velocidades de los móviles, factor de carga, y número de componentes distinguibles de señal debidas a las reflexiones. Los resultados obtenidos indican que las prestaciones del sistema son muy sensibles a la velocidad de los móviles y al factor de carga, y que para una alta velocidad o un bajo factor de carga, se observa una disminución significativa de las prestaciones. También se ha mostrado que a mayor número de componentes de señal recibidas, mayor es el throughput máximo y menor el retardo para un throughput dado. Se han considerado una caracterización de la interferencia y un límite de la carga de tráfico más realistas que los utilizados en trabajos previos. De los resultados obtenidos se extrae la conclusión de que para dimensionar un sistema CDMA de paquetes de manera apropiada, es necesario considerar las diferencias entre las desviaciones de señal recibidas procedentes de móviles transmitiendo hacia la estación base deseada, donde se mide la interferencia total, y hacia otras estaciones base.

Agradecimientos

Este trabajo ha sido parcialmente subvencionado por la CYCIT en el marco del proyecto de investigación TIC98-1040.

Referencias

- [1] Lee J.S., Miller L.E., CDMA Systems engineering Handbook, Artech House, Boston, London, 1998.
- [2] Adachi F., Sawahashi M., Suda H., "Wideband DS-CDMA for Next Generation Mobile Communications Systems", *IEEE Communications Magazine*, pp. 56-69, September 1998.
- [3] Ojanperä T., Prasad R., Wideband CDMA For Third Generation Mobile Communications, Artech House, Boston, London, 1998.
- [4] Crovella, M.E., Bestavros, A., "Self-Similarity in World Wide Web Traffic: Evidence and Possible Causes", *IEEE/ACM Transactions on Networking*, Vol. 5, No. 6, pp. 835-846, December 1997.
- [5] Jansen, M.G., Prasad R., "Capacity, Throughput, and Delay Analysis of a Cellular DS CDMA System with Imperfect Power Control and Imperfect Sectorization", *IEEE Trans. on Vehicular Technology*, Vol. 44, No 1, pp. 303-312, February 1995.
- [6] Sampath A., Mandayam N.B. Holtzman J.M., "Erlang Capacity of a Power Controlled

Integrated Voice and Data CDMA System”, *Proc. IEEE VTC’97*, pp. 1557-1561, Phoenix, AZ USA, May 1997.

- [7] Wijffels C.A.F.J., Misser H.S., Prasad R., “A Micro-Cellular CDMA System over Slow and Fast Ricean Radio Channels with Forward Error Correcting Coding and Diversity”, *IEEE Trans. on Vehicular Technology*, Vol. 42, No 4, pp. 570-580, November 1993.
- [8] Jakes W.C., *Microwave Mobile Communications*, Wiley, New York, 1974.
- [9] Padovani R., “Reverse Link Performance of IS-95 based Cellular System”, *IEEE Personal Communications*, Vol. 1, No. 3, pp. 28-34, Third Quarter 1994.
- [10] Hashem B., Sousa E., “Increasing the DS-CDMA System Reverse Link Capacity by Equalizing the Performance of Different Velocity Users”, *Proc. IEEE ICC’98*, Atlanta, Georgia, USA.

- [11] Viterbi A.M., Viterbi A.J., “Erlang Capacity of a Power Controlled CDMA System”, *IEEE Journal on Selected Areas in Communications*, Vol. 11, No. 6, pp. 892-899, August 1993.
- [12] Fenton, L.F., “The Sum of a Log-normal Probability Distribution in Scattered Transmission Systems”, *IRE Trans. on Communication Systems*, Vol. CS-8, pp.57-67, March 1960.
- [13] Evans, J., Everitt, D., "On the Teletraffic Capacity of CDMA Cellular Networks", *IEEE Trans. on Vehicular Technology*, Vol. 48, No. 1, pp 153-165, January 1999.
- [14] Viterbi A.M., Viterbi A.J., Zehavi E. "Other-Cell Interference in Cellular Power-Controlled CDMA", *IEEE Trans. on Communications*, Vol. 42, No. 2/3/4, pp. 1501-1504, February/March/April 1994.

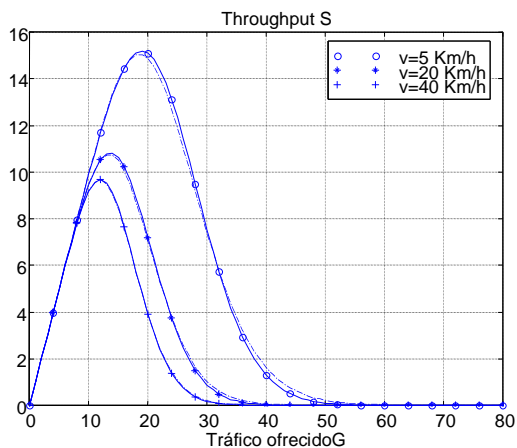


Fig 1. Influencia de la velocidad sobre el throughput. Factor de carga $X_0=0.9$. $M=2$.

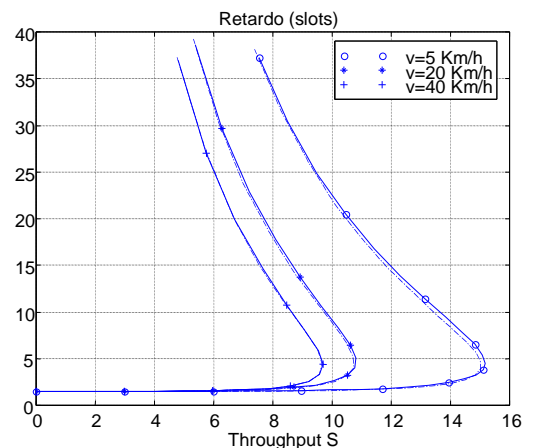


Fig 2. Influencia de la velocidad sobre el retardo. Factor de carga $X_0=0.9$. $M=2$.

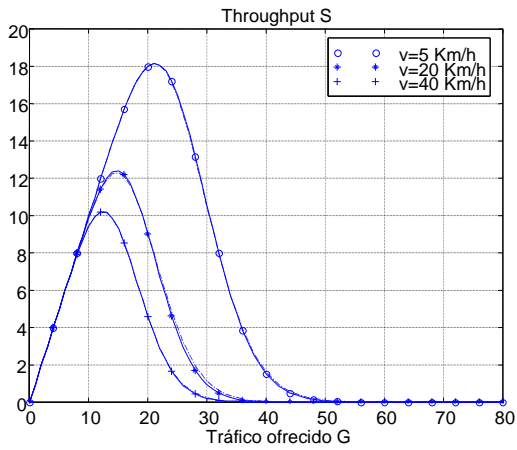


Fig 3. Influencia de la velocidad sobre el throughput. Factor de carga $X_0=0.9$. $M=3$.

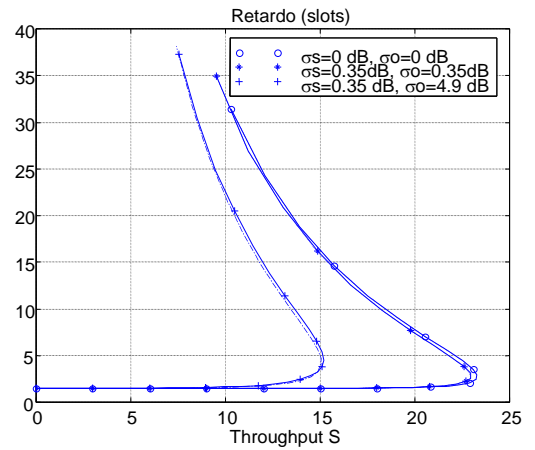


Fig 6. Influencia del error del control de potencia sobre el retardo. Factor de carga $X_0=0.9$. $M=2$. $V=5$ km/h.

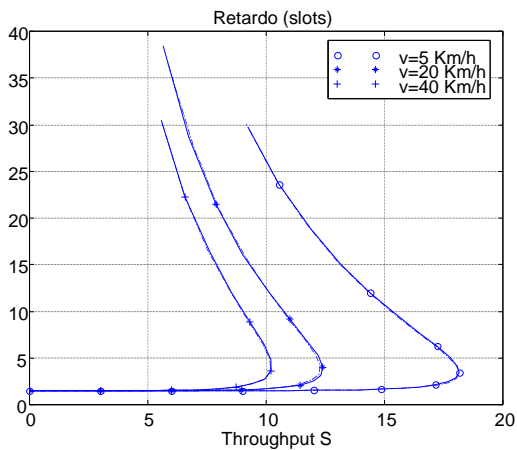


Fig 4. Influencia de la velocidad sobre el retardo. Factor de carga $X_0=0.9$. $M=3$.

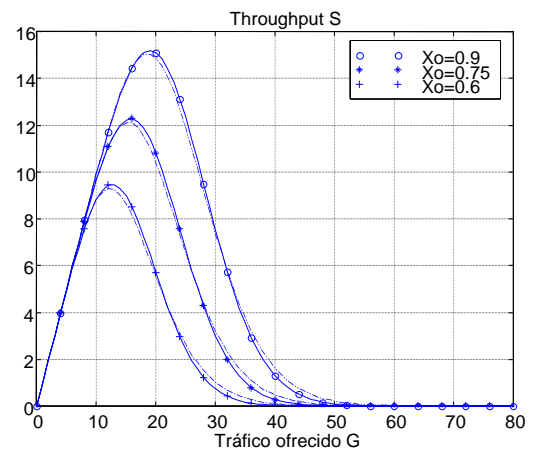


Fig 7. Influencia del factor de carga sobre el throughput. $M=2$. $V=5$ km/h.

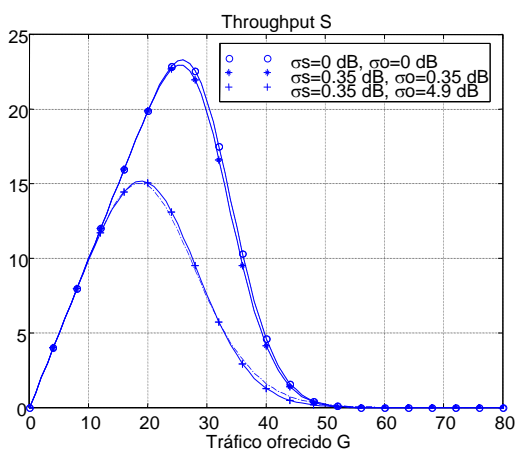


Fig 5. Influencia del error del control de potencia sobre el throughput. Factor de carga $X_0=0.9$. $M=2$. $V=5$ km/h.

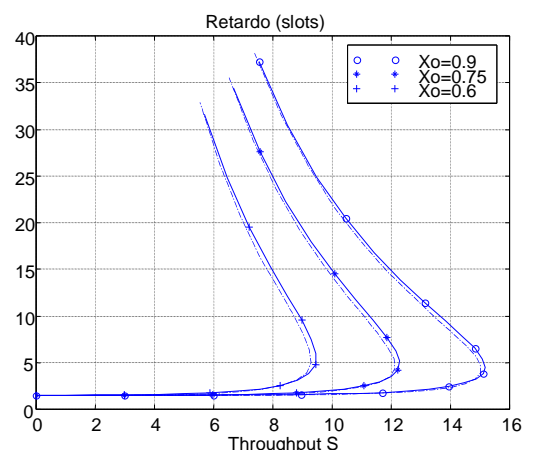


Fig 8. Influencia del factor de carga sobre el retardo. $M=2$. $V=5$ km/h.

Protocolos de Control de Acceso al Medio para Tráfico Multimedia en Redes de comunicaciones Móviles

Loren Carrasco Guillem Femenias*
Universitat de les Illes Balears
Departament de Ciències Matemàtiques
i Informàtica, Ctra Valldemossa km 7.5
07071 Palma ESPAÑA
Email: loren@ipc4.uib.es, guillem@ipc4.uib.es

Ian F. Akyildiz Janise McNair
Broadband & Wireless Networking Laboratory
School of Electrical & Computer Engineering
Georgia Institute of Technology, Atlanta, GA 30332
Email: \{ian,mcnair\}@ee.gatech.edu
Tel: (404)-894-5141; Fax: (404) 894-7883

Abstract

This paper presents a survey on Medium Access Control (MAC) protocols for multimedia traffic in wireless networks. We present a basic overview of MAC protocol concepts and develop a framework on which to base qualitative comparisons. The MAC protocols covered in this paper include classical as well as recently proposed schemes intended for use in a single hop, TDMA or CDMA-based wireless system. The operation of each protocol is explained and its advantages and disadvantages for the operation of multimedia traffic are presented.

I. Introducción

Una de las características más atractivas de la nueva generación de sistemas de comunicaciones móviles es su promesa de incorporar servicios multimedia de alta velocidad. En estos sistemas, un protocolo MAC que asegure una repartición justa y eficiente de los escasos recursos entre servicios muy diferentes se convierte en un elemento crucial del sistema. Este artículo examina los principales protocolos MAC propuestos para redes móviles y trata de evaluarlos de acuerdo con los requerimientos que impone el soporte de tráfico multimedia.

Los protocolos MAC suelen clasificarse según el método utilizado para compartir los recursos [21,41]: acceso aleatorio, asignación dedicada y asignación bajo demanda. En un canal con acceso aleatorio, las transmisiones tienen lugar aleatoriamente y los usuarios deben competir por el canal. Estos esquemas resultan adecuados para tráfico de datos formado por ráfagas cortas e infrecuentes minimizando para estos servicios la carga de señalización asociada y el tiempo de acceso al canal. Sin embargo debido a su incapacidad de garantizar un determinado retardo de transmisión no se consideran como una solución adecuada para el transporte de tráfico multimedia. Los protocolos de asignación dedicada proporcionan a cada usuario una asignación de recursos predeterminada y continua, manteniéndola incluso en ocasiones en las que el usuario no hace uso de ella. Por consiguiente estos protocolos son una buena elección para tráfico con una velocidad de transmisión continua y constante, pero en cualquier otro caso se consideran una mala alternativa ya que implican un elevado desaprovechamiento de los recursos. Finalmente, las técnicas de asignación bajo demanda requieren que el usuario realice peticiones para acceder a los recursos del sistema a medida que los necesita. Las peticiones se realizan a través de un acceso aleatorio o dedicado. A pesar de ser los más complejos, estos protocolos son

los más utilizados para tratar con la combinación de tráficos que forma los servicios multimedia.

Otro concepto fundamental relacionado con los protocolos MAC es la noción de acceso múltiple o esquema de acceso múltiple. Antes de que el ancho de banda disponible pueda ser compartido entre todos los usuarios debe ser dividido en secciones accesibles utilizando un esquema concreto. Los tres métodos más utilizados para la división de los recursos son el acceso múltiple por división en frecuencia (FDMA), por división en el tiempo (TDMA) y por división en códigos (CDMA). En los sistemas de tercera generación se están considerando básicamente esquemas TDMA y CDMA o combinaciones de ambos. El esquema de acceso múltiple adoptado en un sistema condiciona las características del protocolo MAC que vaya a utilizarse. Por este motivo los protocolos MAC analizados se dividen en dos grandes secciones: protocolos MAC propuestos para sistemas TDMA y protocolos MAC diseñados para sistemas CDMA.

Los protocolos considerados están basados en sistemas de comunicaciones móviles con una infraestructura de red en la que un conjunto de estaciones móviles (MS) comparte los recursos radio asignados a un nodo central o estación de base (EB) (Protocolos MAC para redes móviles ad-hoc han sido investigados en [12, 25, 42]). Además el estudio se centra fundamentalmente en la problemática del enlace ascendente donde múltiples MSs compiten por el canal.

Para realizar una selección apropiada de protocolos MAC "multimedia" se han elegido un conjunto adecuado de criterios para este tipo de tráfico:

a) El soporte simultáneo del conjunto de tipos de tráfico multimedia (CBR, rt-VBR, nrt-VBR, ABR y UBR [1])

b) El soporte de tráfico que requiere un bajo retardo tanto absoluto como entre paquetes (jitter)

c) El soporte de un acceso a los recursos justo eficiente y ordenado según un esquema de prioridades

Iniciamos la discusión con protocolos MAC-TDMA, en la sección III, se analizan los protocolos MAC para

* Trabajo subvencionado por la C.I.C.Y.T dentro del proyecto (TIC98-0263), y la Comunidad Autónoma de les Illes Balears, España

CDMA. En este estudio se comentan tanto propuestas clásicas como muy recientes, aunque sólo se analizan con detalle las diseñadas para los nuevos sistemas móviles multimedia.

II. Protocolos MAC para TDMA

Algunos de los protocolos MAC clásicos en este tipo de sistemas son DQRUMA (Distributed Queuing Request Update Multiple Access)[17, 18], RAMA (Resource Auction Multiple Access) [5], MDR-TDMA (Multiservice Dynamic Reservation-TDMA)[38, 39] y PRMA (packet Reservation Multiple Access)[15, 27, 43].

DQRUMA es capaz de soportar tráfico de tipo CBR y VBR, pero no establece ninguna distinción o prioridad entre servicios de tipo VBR y ABR (datos). El protocolo RAMA maneja un esquema de prioridades, pero no permite a los usuarios especificar el ancho de banda requerido. Debido a su incapacidad para el soporte de tráfico multimedia estos protocolos no son incluidos en nuestro análisis. Los protocolos que van a analizarse se agrupan en tres categorías: TDMA dinámicos, derivados del PRMA y TDMA de banda ancha.

A. Protocolos TDMA dinámicos

Los protocolos TDMA dinámicos se caracterizan por realizar diferentes tipos de asignaciones a los usuarios dependiendo del tipo de servicio requerido. Por ejemplo asignaciones de intervalos en modo circuito en el tráfico CBR frente a asignaciones dinámicas para VBR y ABR.

Protocolo TDMA dinámico con duplexación en el tiempo (DTDMA/TDD)

DTDMA/TDD es un protocolo contemporáneo [32] basado en el protocolo descrito en [38, 39]. Es un protocolo para un sistema TDMA con duplexación frecuencial. Utiliza asignación bajo demanda con un canal de acceso aleatorio para realizar las demandas. Cada trama TDMA se divide en una subtrama de asignación fija (tráfico CBR) una subtrama de asignación fija o compartida (tráfico VBR), una subtrama con asignación dinámica (tráfico ABR y UBR) y una subtrama de miniintervalos de demanda.

Para realizar la demanda de una transmisión, un MS con tráfico selecciona aleatoriamente uno de los N_d intervalos de demanda y transmite un paquete de demanda de transmisión. Este paquete incluye los parámetros que caracterizan el tráfico que desea transmitir. Las demandas que sufren una colisión no reciben un ACK desde la EB y son descartadas antes del inicio de la siguiente trama. Las demandas correspondientes a servicios de voz (CBR) tienen prioridad sobre las demandas de datos y en caso de que sufran una colisión son enviadas de nuevo en uno de los intervalos de demanda de la siguiente trama. En el caso de que las demandas para datos sufran colisión, se obliga al MS a esperar un tiempo aleatorio antes de que pueda volver a transmitir su demanda.

Si una demanda llega correctamente a la EB, el MS recibe una señal de ACK y instrucciones sobre el

intervalo que debe utilizar para su transmisión si hay alguno disponible

En el caso del tráfico CBR, la asignación del intervalo se realiza una vez durante el establecimiento de la llamada recibiendo una asignación fija continua del mismo intervalo de tiempo en múltiples tramas (modo circuito), mientras que para el tráfico ABR y UBR se lleva a cabo a cada nueva ráfaga de paquetes vía una reserva dinámica de los intervalos dedicados a este tipo de tráfico o a intervalos inutilizados correspondientes a tráfico VBR y CBR. En el caso del tráfico VBR la asignación de los intervalos se realiza a través de un algoritmo de multiplexación estadística basado en el UPC.

Con este protocolo las llamadas de tipo CBR y VBR pueden ser bloqueadas en el caso de que el sistema este muy cargado, en cambio las llamadas ABR y UBR son siempre aceptadas. Se trata de un protocolo muy enfocado a la transmisión de tráfico ATM, y por tanto establece una mayor diferenciación entre los tipos de tráfico CBR, VBR, ABR y UBR que el protocolo anterior soportando diferentes requerimientos de calidad de servicio. Otra ventaja es la utilización de mini-intervalos para las demandas mejorando así las prestaciones del esquema de acceso.

B. Protocolos derivados del PRMA clásico

El protocolo PRMA es una de las propuestas clásicas para redes móviles [15, 27, 43]. Sus derivados asignan intervalos de tiempo de acuerdo con el ancho de banda reservado y disponible y utilizan algoritmos de multiplexación estadísticos para asignar los intervalos según el tipo de tráfico y las limitaciones en el control de admisión de llamadas. Los protocolos propuestos para mejorar las prestaciones del PRMA incluyen el PRMA++ [11], el PRMA/ATDD (PRMA /adaptive Time Division Duplex) [30], el PRMA/DA (PRMA/Dynamic Allocation)[15], el SIR++ (Service Integration of Radio access)[6], y el C-PRMA (Centralized-PRMA) [9].

El PRMA/DA divide el *uplink* en intervalos para datos, tráfico VBR y tráfico CBR, realizando una asignación dinámica de los intervalos disponibles según el tipo de tráfico. PRMA/ATDD utiliza dos listas de parámetros (una estática para las características de la conexión fijadas en el establecimiento de la llamada y otra dinámica relativa a parámetros relacionados con los niveles de pérdida de paquetes que sufren las conexiones) que permiten a la EB establecer un orden de prioridades de los paquetes dependiendo de los riesgos que sufran de ser descartados (*timeout*). Estos dos protocolos se consideran derivaciones menos interesantes para el tráfico multimedia que los presentados a continuación mientras que el PRMA++ se discute como una parte del SIR++.

SIR++

Los conceptos básicos del SIR++ son: (1) los servicios reciben una asignación garantizada de ancho de banda más una cierta cantidad extra que estará disponible mientras no sea utilizada por otros servicios, y (2) los servicios se asocian con unos niveles de prioridad para gestionar la asignación del ancho de banda extra. El

SIR++ utiliza un mecanismo de riguroso turno para distribuir el ancho de banda extra a los servicios en cada clase o nivel de prioridad, asegurando de esta forma una repartición justa y eficiente de los recursos. Este protocolo soporta un número arbitrario de clases de servicios cada una con unos requerimientos concretos en términos de retardo y *throughput*. Las clases se identifican según números enteros desde 0 (la más prioritaria) hasta un valor máximo *Clasemax*. Los usuarios especifican la clase de servicio que requieren en el momento del establecimiento de la llamada. Los usuarios de clase 0 son tratados como los usuarios de voz en el protocolo PRMA++ . La clase 1 asocia a cada usuario un ancho de banda mínimo garantizado. El ancho de banda adicional puede asignarse si es demandada por el usuario y hay recursos disponibles siguiendo una política del mejor esfuerzo. Por tanto esta clase de servicio es adecuada para tráfico VBR. Las clases que van desde la 2 hasta *Clasemax* tratan de acomodar diferentes tipos de tráfico con distintos requerimientos de retardo y *throughput*. Estas clases se comportan de forma similar a la clase 1 con dos diferencias: la existencia de un ancho de banda mínimo garantizado es opcional y en caso de existir la asignación se realiza conjuntamente para un grupo de sesiones de datos con la misma clase de servicio y no de forma individual. El ancho de banda extra se va repartiendo siguiendo el esquema de prioridades marcado por las clases.

La estructura de trama del SIR++ puede verse como una generalización de la trama de PRMA++. Los intervalos del *uplink* están separados en intervalos para realizar reservas (intervalos R) e intervalos de tráfico (intervalos I). Los intervalos del *downlink* se separan en intervalos de ACK de demandas (intervalos A), emparejados con los intervalos R del *uplink*, e intervalos I. En una trama ascendente, los intervalos I y R son divididos en subconjuntos uno para cada clase de servicio. Los intervalos R-clase0 son accedidos por contención entre los MSs, mientras que el acceso a los otros R intervalos sigue una política de riguroso turno entre los MSs de esa clase. Los intervalos R-clase0 se utilizan para el establecimiento inicial de la conexión en todas las clases. El mecanismo para asignar los intervalos I se basa en la distinción entre los estados disponible y reservado. Un intervalo I se dice que esta reservado cuando su asignación a una sesión individual (clases 0 y 1), o a un grupo de sesiones (clases de 2 a *Clasemax*) implica el uso repetido de ese intervalo durante un período que puede corresponder a un corto período de actividad o a toda la duración de la sesión. Un intervalo I disponible es aquel en el que su uso en una trama concreta no implica su uso repetido por la misma sesión o grupo de sesiones en tramas siguientes. Los servicios de clase 0 sólo utilizan intervalos I reservados mientras que los servicios de clases 1 en adelante pueden utilizar una mezcla de ambos tipos de intervalos I. Una distinción adicional es que mientras los servicios de clase 1 utilizan intervalos I reservados para proveer el ancho de banda mínimo garantizado y intervalos I disponibles para el ancho de banda extra, los servicios de clases mayores o iguales a 2 utilizan intervalos reservados de manera opcional.

Finalmente para asegurar una asignación justa de los recursos una disciplina de riguroso turno es utilizada para la asignación de intervalos I para las clases de la 1 a la *Clasemax*.

Una de las limitaciones de este protocolo es que utiliza intervalos completos para realizar las reservas en lugar de mini-intervalos incrementando el efecto de las posibles colisiones.

C-PRMA, PRMA centralizado

En el C-PRMA, la EB juega un papel fundamental en la temporalización de las transmisiones por parte de los MSs [9]. Las dos principales características del C-PRMA son: (1) su capacidad para integrar diferentes tipos de servicios, y (2) la habilidad de permitir una rápida retransmisión de paquetes con errores.

Esta segunda característica mejora la calidad de servicio en entornos de propagación difíciles a través de la adaptación de un acceso aleatorio y un esquema de *polling*. El acceso aleatorio es utilizado por los MSs para reservar intervalos para la transmisión, mientras que un esquema de *polling* se utiliza para la transmisión de los paquetes de información. El comando de *polling* gestionado por la EB, especifica si un intervalo del *uplink* esta disponible o reservado y en ese caso identifica al MS que tiene permiso para transmitir en él.

Los intervalos disponibles en el *uplink* están dedicados a la transmisión de demandas de reserva. Las demandas de reserva pueden estructurarse como un mini-paquete, ya que contienen una cantidad limitada de información incluyendo el identificador del MS, el tiempo que lleva el primer paquete en el buffer de transmisión y un campo CRC. El MS transmite repetidamente la demanda de reserva en uno de los mini-intervalos disponibles hasta que recibe una señal de ACK de la EB. Entonces el MS transmite sus paquetes de datos en los intervalos reservados por el comando de *polling* de la EB. La secuencia de *polling* para los MSs con una reserva es generada por un algoritmo de temporalización que maximiza las prestaciones del sistema gracias al uso de los diferentes requerimientos de retardo, tasa de pérdida de paquetes de los diferentes servicios. Si se detecta una colisión en uno de los mini-intervalos de acceso, se inicia un ciclo de resolución de la colisión. Este ciclo finaliza cuando todos los mini-paquetes que colisionaron han sido recibidos correctamente en la EB. Las ventajas de este protocolo son como en casos anteriores el uso de mini intervalos para las reservas y la utilización de un esquema de *polling* para las asignaciones. Los intervalos reservados a un MS son asignados dinámicamente por la EB sin ningún tipo de estructura de trama, de forma que un paquete erróneo puede ser recuperado inmediatamente a través de una retransmisión. Otra ventaja es el uso de un algoritmo de separación (*splitting*) para resolver las colisiones de forma rápida y eficiente.

C. Protocolos TDMA de banda ancha

Los protocolos TDMA de banda ancha incluyen protocolos diseñados para el transporte de células ATM, como el DSA++ (dynamic Slot

Assignment++)[28], el protocolo MASCARA (Mobile Access Scheme based on Contention And Reservation for ATM) [10, 31], y el protocolo presentado en [37] que extiende el multiplexado estadístico de los enlaces ATM al interfaz radio. Tanto DSA++ como MASCARA están siendo considerados para su uso en el estándar HIPERLAN tipo 2 para tráfico multimedia. Como representación de este tipo de protocolos se analiza en más detalle el protocolo Mascara.

MASCARA

Este protocolo ha sido propuesto como parte del demostrador ATM radio desarrollado por el proyecto europeo Magic WAND [10, 31]. La subtrama ascendente se transmite utilizando un modo mixto de contención y reserva, mientras que la trama descendente se gestiona en modo TDM. La subtrama del downlink se divide en un período de cabecera de trama y un período para la transmisión de paquetes de información. La subtrama del uplink se divide en un período de contención utilizado para transmitir las demandas de transmisión u otra información de control y un período para la transmisión de paquetes de información. Cada uno de estos períodos dentro de la trama tiene una longitud variable que depende de las características del tráfico que debe transmitirse en el canal.

Si un MS tiene celdas en espera para transmitir, envía una demanda de reserva de recursos, que o bien se envían en un paquete de control en el período de contención, o bien se empaquetan detrás de un paquete de datos que se este transmitiendo en la trama actual. Al final de la trama, la EB construye la trama siguiente a partir de las demandas recibidas, las celdas que esperan para transmitirse en el downlink, y las características del tráfico en ese momento. En el período de cabecera trama, en la subtrama descendente, la EB informa de la forma que tendrá la trama siguiente (incluyendo las longitudes de cada período), los resultados de las transmisiones efectuadas en la trama anterior en modo de contención, y la posición del intervalo asignado a cada conexión ascendente y descendente. Esto permite a los MSs el ahorro de baterías. Con el objetivo de minimizar la carga de señalización producida por la capa física radio, MASCARA introduce el concepto de tren de celdas. Un tren de celdas es una secuencia de 1 a n celdas ATM que pertenecen al mismo MS y tienen una cabecera (capa física) común que se ajusta al tamaño de una celda ATM (53 bytes).

El temporalizador utiliza un algoritmo denominado PRADOS (Priority Regulated Allocation Delay-Oriented Scheduling) que ordena las transmisiones sobre la interfaz radio. El algoritmo PRADOS esta basado las figuras de retardo permitidas en cada una de las conexiones activas y en un esquema de clases de servicios (cada clase tiene un nivel de prioridad) [27]. PRADOS asigna prioridades a cada conexión de acuerdo con su clase de servicio: CBR (5), rt-VBR (4), nrt-VBR (3), ABR (2) y UBR (1) (un número mayor indica un nivel de prioridad más elevado).

Este algoritmo combina las prioridades con un regulador de tráfico basado en la creación de un

conjunto de testigos por cada conexión. Los testigos se generan a una tasa fija igual a la tasa media de generación de celdas ATM de ese VC. El tamaño de este conjunto es igual al número máximo de celdas ATM que pueden ser transmitidas con una velocidad mayor que la media. Empezando por las conexiones con prioridad máxima (5) y finalizando con las de prioridad 2, el temporalizador satisface las demandas de las conexiones mientras existan testigos y intervalos disponibles. Para cada intervalo asociado a una conexión, un testigo es eliminado de su conjunto. PRADOS intenta además transmitir un paquete tan cerca como sea posible de su *timeout* de forma a maximizar en la medida de lo posible el número de celdas transmitidas antes de que se alcance el límite del retardo permitido.

Una de las principales novedades de este protocolo es la introducción del concepto de tren de celdas y su capacidad de gestionar diferentes servicios gracias al algoritmo PRADOS, que es capaz de decidir la asignación de los intervalos en la trama siguiente de acuerdo con parámetros de calidad de servicio negociados para cada conexión.

III. Protocolos MAC para CDMA

Los protocolos CDMA clásicos [46,47] no incorporan mecanismos de prioridad ni ofrecen garantías de calidad de servicio, mientras que otros protocolos CDMA [23, 24, 25] fueron diseñados para voz con un soporte muy limitado de servicios de datos. Actualmente, la tecnología CDMA ofrece nuevos retos para protocolos multimedia, debido a que estos protocolos no solo deben compartir los recursos de forma eficiente sino que además deben garantizar la estabilidad del sistema, es decir, controlar el nivel de interferencias sobretodo durante los picos de tráfico. Además, el nivel de interferencia experimentado por cada tipo de MS puede ser diferente y debe ser considerado separadamente. En esta sección se examinan dos clases de protocolos MAC: protocolos para sistemas híbridos TDMA/CDMA, y protocolos orientados estrictamente a sistemas CDMA.

A. Protocolos Híbridos TDMA/CDMA

Los protocolos híbridos TDMA/CDMA se benefician por un lado de la capacidad de los esquemas TDMA para manejar elevadas velocidades de transmisión y los servicios por conmutación de paquetes, y por otro lado de la flexibilidad de las técnicas CDMA para permitir una coexistencia natural entre diferentes tipos de tráfico.

MD PRMA/BB [21] es una derivación del clásico protocolo PRMA adaptado a un sistema híbrido con soporte de tráfico multimedia. WISPER en cambio [4] utiliza una aproximación mucho más determinista, siendo la EB la que controla las transmisiones de los MSs.

MD PRMA/BB

Este protocolo esta pensado para cualquier esquema híbrido del tipo FDMA/TDMA o CDMA/TDMA. Sin embargo en [21] los parámetros utilizados asumen la utilización del acceso TDMA/CDMA TDD elegido por

el ETSI como acceso para las bandas pares del sistema UMTS. MD PRMA/BB es una derivación del clásico protocolo PRMA adaptado a un sistema híbrido con soporte de tráfico multimedia. Como en PRMA intervalos de tiempo de igual longitud se organizan en tramas, pero en MD PRMA/BB estos intervalos se vuelven a subdividir utilizando hasta ocho códigos CDMA diferentes. Estos sub-intervalos (intervalo+código) están o bien disponibles para su acceso en modo de contención (subintervalo C) o bien están reservados para la transferencia de información de un determinado MS (subintervalo I). Un MS que no tiene ninguna reserva pero que ha sido admitido en el sistema competirá por una reserva de intervalo utilizando los sub-intervalos C. El MS enviará una demanda en un subintervalo C concreto con una probabilidad especificada que viene determinada por el tipo de servicio y el intervalo de tiempo elegido. Las probabilidades para cada tipo de servicio y para cada intervalo de tiempo en la trama ascendente siguiente son transmitidas por la EB en la trama descendente. Las probabilidades dependen del número estimado de MSs en espera (MSs con un paquete para transmitir pero sin ningún intervalo asignado) que estén en ese momento en el sistema, y son calculadas de manera que garanticen la estabilidad del protocolo y las máximas prestaciones en términos de retardo y *throughput*. Además, estas probabilidades están conectadas a un control de acceso basado en la carga del sistema para asegurar el control de las interferencias de la componente CDMA y la estabilidad del sistema. La información de la EB a los MSs incluye además de estas probabilidades las señales de ACK para las demandas realizadas en la trama ascendente anterior. En general se utiliza la reserva implícita, es decir, una vez que el MS ha recibido el ACK iniciará la transmisión en el subintervalo C que utilizo para realizar la demanda. Desde ese momento, ese sub-intervalo se convierte en un subintervalo tipo I reservado para ese usuario. La reserva se mantendrá hasta el final de esa ráfaga en el caso de un servicio de voz, o durante un cierto número de tramas (F) si se trata de servicios de datos. A través del control de las probabilidades de acceso y del número de tramas F asignadas a los servicios de datos, este protocolo es capaz de mantener los requisitos de retardo y probabilidades de pérdida de paquetes de los diferentes servicios. Una de las principales desventajas del MD PRMA/BB es que no puede soportar servicios que requieran una velocidad de transmisión elevada o servicios en tiempo real como el vídeo que requieran la asignación de múltiples sub-intervalos en la misma trama. Otra de sus limitaciones es que no tiene demasiado en cuenta la problemática asociada a la componente CDMA, permitiendo por ejemplo que diferentes servicios (con probabilidades de error permitidas diferentes) compartan en mismo intervalo temporal. De esta forma la capacidad de cada intervalo será variable y estará limitada por el servicio con una probabilidad de error permitida más baja.

WISPER

Este protocolo utiliza la novedosa idea [4] de temporalizar las transmisiones de los paquetes multimedia de acuerdo con las especificaciones de la tasa de error requerida en las diferentes clases de servicios. Utiliza duplexación frecuencial y cada una de estas bandas, se divide en intervalos temporales organizados en tramas. En el uplink, cada trama se divide en N_p intervalos de información que pueden transportar cualquier tipo de tráfico y un intervalo de demandas que los MSs pueden utilizar con dos objetivos: para realizar demandas de admisión (nuevos MSs que quieren acceder al sistema) y para efectuar demandas de

transmisión (MSs ya conectados). Cuando un MS quiere ser admitido en el sistema, primero selecciona un código pseudo aleatorio del conjunto de códigos especialmente reservados para ese propósito. Utilizando ese código, el MS transmite una demanda de admisión en el intervalo de demandas. Esta demanda incluye el número de identificación del MS, información sobre la clase de tráfico a transmitir y la velocidad máxima de transmisión del MS. Se producen colisiones en el intervalo de demandas si dos MSs han utilizado el mismo código, o si el número total de usuarios transmitiendo en ese intervalo genera un nivel de interferencia superior al que es capaz de resolver el receptor. Si no se producen colisiones, y si el controlador de admisión de llamadas determina que los requerimientos de calidad de servicio de los otros MSs ya conectados al sistema pueden todavía ser satisfechos después de la introducción del nuevo MS, el MS recibe una notificación de que ha sido aceptado en el sistema junto con un identificador único y un pseudo código para uso exclusivo de ese MS. Cada vez que un MS tiene nuevos paquetes que transmitir debe enviar una demanda de transmisión a la EB indicando el número de paquetes y el *timeout* de esos paquetes. El MS puede hacer las demandas a través del intervalo de demandas pero también puede empaquetar su demanda a continuación del paquete de datos que este enviando en la trama actual. Esta última opción se utiliza siempre que sea posible para reducir las colisiones en el intervalo de demandas. La EB realiza el seguimiento del conjunto de paquetes asociado con cada demanda hasta que todos los paquetes han sido enviados o hasta que superan su *timeout* y son descartados.

En el caso del *downlink*, tenemos N_p intervalos para la transmisión de información y un intervalo de control. El intervalo de control es utilizado por la EB para notificar la correcta recepción de los paquetes enviados en la trama anterior y para enviar instrucciones a los MSs sobre las condiciones de transmisión en la próxima trama, indicando a cada MS el intervalo(s) que puede utilizar y el número de paquetes que puede transmitir en la trama siguiente.

En el caso de la trama ascendente el intervalo de demandas no puede ser el último de la trama, ya que la EB necesita tiempo para procesar las demandas. También debe considerarse que los MSs deben conocer las asignaciones que les corresponden a tiempo y por tanto es necesario que la trama descendente siempre empiece por el intervalo de control y que las tramas

ascendentes y descendentes estén ligeramente. El tamaño de los intervalos de demanda y de control es idéntico a la de los intervalos de información.

La mejora más significativa introducida por el protocolo WISPER es la agrupación de conexiones con requerimientos de tasas de error similares, en el mismo intervalo de tiempo. Incrementando de esta forma al máximo la capacidad de la componente CDMA. El orden de transmisión se determina de acuerdo con un algoritmo que tiene en cuenta los tiempos de vida de los paquetes en cada una de las conexiones y el número de paquetes preparados para ser transmitidos esperando en el MS.

B. Protocolos MAC para sistemas CDMA

El primer protocolo que se describe [29] está basado en un sistema de colas con prioridades y en la asignación de probabilidades de transmisión para MSs. El segundo protocolo descrito en [25] ha sido diseñado específicamente para la capa física CDMA de banda ancha adoptada por ETSI y ARIB en sus futuros estándares IMT-2000[23].

MR-CDMA (Cellular packet Multirate CDMA)

Este protocolo ha sido diseñado para un sistema DS/CDMA [29]. Se trata de un protocolo orientado a la transmisión de paquetes que incluye un sistema de colas con prioridades para diferentes tipos de tráfico. Cada paquete incluye una cabecera de sincronización, una celda ATM, y una cola con bits de paridad (ECC). Se asume que existe un canal ideal en el envío de las notificaciones de una correcta recepción. Los usuarios se clasifican en diferentes tipos de acuerdo con la velocidad de transmisión que requiere el tráfico que generan. En el caso de que exista tráfico con la misma velocidad de transmisión pero con diferente prioridad, es posible definir diferentes clases para cada uno. Una vez que el tráfico llega desde la fuente se guarda en la cola de longitud finita para ese tipo de tráfico. Si hay un paquete en la cola, el usuario intenta transmitirlo al inicio del siguiente intervalo. El usuario puede asumir tres estados, vacío, activo y bloqueado, dependiendo del estado de la cola del MS y del éxito de la anterior transmisión. Si no hay ningún paquete en la cola el usuario se encuentra en el estado vacío. Un usuario activo o bloqueado asume un subestado dependiendo del número de paquetes esperando en la cola. Si un usuario "vacío" genera un paquete, el paquete pasa a la cola y ese usuario pasa al estado "activo". Un usuario en estado "activo" trata de realizar la transmisión del primer paquete de la cola con probabilidad PA. Este valor puede ser diferente en diferentes usuarios de manera a asegurar diferentes niveles de prioridad en diferentes colas. Una probabilidad más baja corresponde a una prioridad más alta. Si la transmisión tiene éxito, el usuario intenta transmitir el siguiente paquete en la cola si esta no está vacía y el estado del usuario permanece en "activo". Si la transmisión falla el usuario asume el estado de bloqueado y intenta la retransmisión con una probabilidad PB. Si la retransmisión falla, el usuario continúa "bloqueado" en caso contrario el usuario pasa al estado "activo" o

"vacío" dependiendo de si la cola ha quedado o no vacía.

Se trata de un protocolo simple que puede soportar servicios con velocidades de transmisión elevadas sin ningún límite en el número de tipos de tráfico. Sin embargo, el uso de un tipo de acceso con contención implica el riesgo de retardos importantes y pérdida de paquetes.

Protocolo MAC para un sistema CDMA de banda ancha

Está diseñado para gestionar una capa física de 3ª generación que es capaz de transferir datos de diferentes servicios simultáneamente desde un único MS y permitir una velocidad variable de transmisión. En [25] se describen diferentes procedimientos para servicios en tiempo real, servicios de datos y servicios mixtos (servicios multiplexados procedentes de un único MS) *Servicios de datos por conmutación de paquetes*: se distinguen dos modos de operación. En el primero, paquetes cortos pueden ser transmitidos utilizando un protocolo de acceso aleatorio ALOHA a través del canal de acceso aleatorio (RACH). El RACH es un canal común para todos móviles de la célula cuya principal función es la de canalizar las demandas de transmisión. Este modo es utilizado para transmitir paquetes cortos e infrecuentes con un retardo nulo y una carga de señalización mínima. Sin embargo en este modo los paquetes pueden perderse debido a colisiones y además el MS no utiliza control de potencia rápido en este canal con lo que debe aumentar la potencia de transmisión.

En el caso de paquetes de mayor longitud, el MS debe realizar una demanda a través del RACH para obtener un canal dedicado DCH, es decir un código CDMA dedicado con un control de potencia rápido y un canal de control asociado (que transporta comandos de control de potencia y información sobre la tasa de transmisión que se está utilizando en el DCH). Debido a que la longitud del paquete es mayor, la sobrecarga de señalización provocada por la demanda es despreciable. La demanda incluye el tipo de servicio y eventualmente la longitud del paquete. Después de evaluar si los recursos demandados están disponibles, el sistema responderá indicando un conjunto de formatos de transmisión posibles (TFs). Si en ese momento el sistema no está muy cargado, indicará también el TF específico que se debe utilizar para iniciar la transmisión y el momento en que debe iniciarla. En momentos de alta carga, en primera instancia sólo se indica al móvil el conjunto de TFs posibles. Para recibir el TF concreto a utilizar el móvil deberá realizar una nueva demanda a través del RACH. Mientras el MS se halla transmitiendo el sistema aumentará o disminuirá su velocidad de transmisión dependiendo de la carga del sistema en cada momento. Una vez que la transmisión haya concluido, el MS conservará la conexión durante un cierto tiempo. Si genera un nuevo paquete durante ese tiempo podrá iniciar su transmisión inmediatamente con el último TF utilizado. Sin embargo, si ese nuevo paquete es muy grande, el MS deberá primero realizar una demanda a través de su DCH pidiendo permiso para iniciar la

transmisión. Finalmente, si no llegan más paquetes durante ese intervalo, la conexión se perderá, pero el MS todavía mantendrá el conjunto de TFs durante un cierto periodo para facilitar transmisiones futuras.

Servicios en tiempo real: El procedimiento de asignación es muy similar al anterior. Cuando el MS tiene datos a transmitir, envía una demanda a través del RACH. El sistema responde con un conjunto de TFs y entonces el MS puede iniciar la transmisión inmediatamente utilizando cualquiera ellos. Este grado de libertad en el MS permite una velocidad de transmisión variable. Si la carga en el sistema es elevada, este puede verse obligado a limitar a un subconjunto los TFs iniciales.

Servicios mixtos : El sistema asignará un conjunto de TFs para cada servicio. El MS podrá utilizar cualquier TF para los servicios en tiempo real mientras que el sistema debe indicar el TF específico a utilizar con los servicios de datos. Para este tipo de MSs se aplicará una limitación adicional en términos de un límite máximo de potencia/velocidad de transmisión global.

Este protocolo trata con una capa física compleja y muy flexible donde son posibles muchas opciones de transmisión. El control se realiza a través de un protocolo de asignación bajo demanda (excepto para paquetes muy cortos) que descansa en una compleja gestión de recursos realizada en la EB y posiblemente más arriba en el sistema. De hecho, la estabilidad del protocolo depende de funcionalidades en capas superiores como el control de admisión de llamadas, o el controlador de congestiones. Por ejemplo, cuando una célula está muy cargada y los usuarios están sufriendo una degradación en la calidad de su enlace debido a un nivel excesivo de interferencias, las llamadas nuevas serán bloqueadas y el controlador de congestiones ordenará al protocolo MAC que disminuya la velocidad de transmisión (TFs con velocidad de transmisión inferiores) de los MSs ya conectados a la EB. Desafortunadamente estas funcionalidades de capas superiores no están todavía completamente definidas.

Otra característica que requiere una evaluación más detallada es la utilización intensiva de un acceso ALOHA en el canal RACH para la transmisión de paquetes cortos y diferentes tipos de demandas. En situaciones de carga elevada (o si los MSs hacen un uso incontrolado de las transmisiones de paquetes cortos), las prestaciones del protocolo pueden verse severamente degradadas, no solo debido al incremento de los retardos y pérdida de paquetes debido a un mayor número de colisiones, sino también debido al incremento del nivel de interferencias producido por una cantidad importante de transmisiones efectuadas sin un control de potencia rápido.

IV. Conclusión

La mayoría de los protocolos actuales utiliza asignación bajo demanda con algún tipo de período de contención para realizar las demandas. Debido a que los sistemas móviles no pueden aprovechar técnicas como la detección de colisiones utilizadas en sistemas fijos, las transmisiones sufren un retardo debido al

tiempo de espera de los MSs de la llegada de la ACK de una correcta recepción del paquete, y en caso de que la transmisión fracase del tiempo que se necesita para la retransmisión. Líneas de investigación futuras deben orientarse en desarrollar métodos que reduzcan la contención y el retardo que implican las retransmisiones. La asignación de intervalos y/o código ofrece una oportunidad de maximizar la utilización del espectro y es otro aspecto que debe profundizarse. La gestión de un esquema de prioridades incrementa la complejidad de los protocolos pero es necesaria para garantizar la calidad de los diferentes tipos de servicios que conforman el tráfico multimedia.

Finalmente todavía queda mucho por hacer con respecto a los nuevos protocolos MAC diseñados para los sistemas IMT-2000. Estos protocolos deberán trabajar con una capa física muy compleja y flexible donde existen opciones de transmisión distintas y donde los procedimientos utilizados deberán manejar un entorno cada vez más complejo.

Bibliografía

- [1] ATM Forum "Traffic Management Specification" V.4.0, Apr.96
- [2] ETSI TC-RES, "Radio Equipment and Systems (RES); High Performance Radio Local Area Networks (HIPERLAN); Type 1; Functional Specification, Technical Correction," Dec 1996.
- [3] IEEE 802.11D3, "Wireless LAN Medium Access Control and Physical Layer Specifications," Jan .96.
- [4] I.F. Akyildiz, et al., "A Slotted CDMA Protocol with BER Scheduling for Wireless Multimedia Networks," to appear in IEEE/ACM Trans. on Networking, 1999.
- [5] N. Amitay, "Resource Auction Multiple Access (RAMA): Efficient Method for Fast Resource Assignment in Decentralized Wireless PCS," Electronics Letters, Vol. 28, No. 8, pp. 799--801, April 1992.
- [6] G. Anastasi, et al., "An Access Protocol Speech/Data/Video Integration in TDMA Based Advanced Mobile System," IEEE Journal on Selected Areas in Communications, Vol. 15, No.8, pp. 1498--1510, October 1997.
- [7] G. Anastasi, et al., "MAC Protocols for Wideband Wireless Local Access: Evolution Toward Wireless ATM," IEEE Personal Communications Magazine, pp. 53--64, October 1998.
- [8] F. Bauchot et al., "MASCARA, a MAC Protocol for Wireless ATM," Proc. ACTS Mobile Summit'96, Granada, Spain, pp. 17--21, November 1996.
- [9] G. Bianchi, et al., "C-PRMA: A Centralized Packet Multiple Access for Local Wireless Communications," IEEE Trans. on Vehicular Tech., Vol 46, No. 2, May .97.
- [10] C. Zhu et al., "A Five-Phase Reservation Protocol for Mobile Ad Hoc Networks," Proc. of INFOCOM'98, pp. 322-331.
- [11] J. De Vile, "A Reservation Based Multiple Access Scheme for a Future Universal Mobile Telecommunications System," Proc. 7th IEEE Conf. on Mobile and Personal Communications, Brighton, U.K., pp.210-215, Dec. 13--15/93.
- [12] D.J. Goodman and S.X. Wei, "Efficiency of Packet Reservation Multiple Access," IEEE Trans. on Vehicular Technology, Vol. 40, No.1, pp.170-176, Feb.91.
- [13] M.J. Karol et al., "An Efficient Demand-Assignment Multiple Access Protocol for Wireless Packet (ATM) Networks," ACM/Baltzer Journal of Wireless Networks, Vol. 1, No.3, pp.267--279, October 1995.
- [14] M.J. Karol, et al., "Distributed-Queueing Request Update Multiple Access (DQRUMA) for Wireless Packet (ATM) Networks," Proc. IEEE INFOCOM'95, pp. 1224--1231.
- [15] G. Kim, et al., "PRMA/DA: A New Media Access Control Protocol for Wireless ATM," Proc. ICC'96, Dallas, pp.1-19,
- [16] A. Klein et al., "FRAMES Multiple Access Mode 1- Wideband TDMA with and without Spreading," Proc. IEEE Int. Conf. on Personal Indoor and Mobile Radio Communications (PIMRC'97), Helsinki, Finland, Sept.97.
- [17] O. Kubbar and H.T. Mouftah, "Multiple Access Control Protocols for Wireless ATM: Problems Definition and Design Objectives," IEEE Communication Mag., pp. 93-99, Nov.97.

- [18] E. Geraniotis, "Multi-Access Strategies for an Integrated Voice/Data CDMA Packet Radio Network," IEEE Transactions on Communications, Vol.43, Jan. 1995.
- [19] J. Holtzman, "Access Control of Data in Integrated Voice/Data CDMA Systems: Benefits and Tradeoffs," IEEE Journal on Selected Areas in Communications, Vol.15, Oct 97.
- [20] A. Muir and J.J. Garcia-Luna-Aceves, "A Channel Access Protocol for Multihop Wireless Networks with Multiple Channels," Proc. IEEE ICC '98, Atlanta, Jun 98.
- [21] A. Brand, A.H. Aghvami, "Multidimensional PRMA with Prioritized Bayesian Broadcast-A MAC Strategy for Multiservice Traffic over UMTS," IEEE Trans. on Vehicular Technology, Vol.47, No.4, pp.1148-1161, Nov.98.
- [22] S. Nanda, "Analysis of Packet Reservation Multiple Access: Voice Data Integration for Wireless Networks," IEEE GLOBECOM'90, Vol. 3, pp. 1984--1988, 1990.
- [23] T. Ojanpera, R. Prasad "An Overview of Air Interface Multiple Access for IMT-2000/UMTS," IEEE Communications Magazine, Vol.36, No.9, pp.70-79, Sept.98.
- [24] F. Ovesjo, et al., "FRAMES Multiple Access Mode 2- Wideband CDMA," Proc. IEEE Int. Conf. on Personal Indoor and Mobile Radio Comm. (PIMRC'97), Helsinki, Sept.97.
- [25] R. Roobol, et al., "A proposal for an RLC/MAC Protocol for wideband CDMA capable of handling Real Time and Non Real Time Services" IEEE Vehicular Technology Conference, Ottawa, Canada, May 1998.
- [26] N. Passas, et al., "Quality of Service-Oriented Medium Access Control for Wireless ATM Networks," IEEE Communication Magazine, November 1997.
- [27] N. Passas et al., "MAC Protocol and Traffic Scheduling for Wireless ATM Networks," ACM Mobile Networks and Applications, 1996.
- [28] D. Petras and Kramling, "MAC Protocol with Polling and Fast Collision Resolution for an ATM Air Interface," IEEE ATM Workshop, San Francisco, Aug.96.
- [29] R. Pichna and Q. Wang, "A Medium-Access Control Protocol for a Cellular Packet CDMA Carrying Multirate Traffic," IEEE Journal on Selected Areas in Communications, Vol. 14, No. 9, pp. 1728--1736, Dec.96.
- [30] F.D. Priscoli, "Medium Access Control for the MEDIAN System," Proc. ACTS Mobile Summit Granada, pp.1--8, Nov.96.
- [31] X. Qui, et al., "A Multiple Access Scheme for Multimedia Traffic in Wireless ATM," Journal of Mobile Networks and Applications, Vol. 1, No. 3, pp. 259--272, Dec. 1996.
- [32] D. Raychaudhuri et al., "WATMnet: A Prototype Wireless ATM System for Multimedia Personal Communication," IEEE Journal on Selected Areas in Communications, Vol. 15, No. 1, pp. 83--95, Jan. 1997.
- [33] D. Raychaudhuri and N. D. Wilson, "ATM-Based Transport Architecture for Multiservices Wireless Personal Communication Networks," IEEE Journal on Selected Areas in Communication, Vol. 12, No. 8, pp. 1401--1414, October 1994.
- [34] D. Raychaudhuri et al., "Multimedia Transport in Next-Generation Personal Communication Networks," IEEE ICC'93, pp. 858--862.
- [35] J. Sanchez, R. Martinez and M.W. Marcellin, "A Survey of MAC Protocols Proposed for Wireless ATM," IEEE Network, pp. 52--62, November-December 1997.
- [36] S. Singh et al., "PAMAS--Power Aware Multi-Access Protocol with Signaling for Ad-hoc Networks," ACM Computer Communication Review, Vol. 28, No. 3, Jul. 98.
- [37] P.F.M. Smulders and C. Blondia, "A MAC Protocol for ATM-Based Indoor Radio Network," Report for the European Coop. in the Field of Scientific and Tech. Research (EUCCO-COST), COST 231, TD(94)055, Apr94.
- [38] L. Tan and Q.T. Zhang, "A Reservation Random-Access Protocol for Voice/Data Integrated Spread Spectrum Multiple-Access Systems," IEEE Journal on Selected Areas in Communications, Vol. 14, No. 9, pp. 1717--1727, Dec. 1996.
- [39] N.D. Wilson, et al., "Packet CDMA Versus Dynamic TDMA for Multiple Access in an Integrated Voice/Data PCN," IEEE Journal on Selected Areas in Communications, Vol. 11, No. 6, pp. 870--884, Aug 1993.

Análisis de controles de admisión en un sistema CDMA para tráfico heterogéneo

Margarita Ruiz García, Juan Manuel Romero Jerez, Antonio Díaz Estrella
Departamento de Tecnología Electrónica, E.T.S. I. Telecomunicación, Universidad de Málaga
Campus Teatinos s/n, 29071 (MÁLAGA)
Telef: 95-2132843, Fax: 95-2131447
E-mail: marga@dte.uma.es , romero@dte.uma.es, ade@dte.uma.es

Abstract

The capacity of a CDMA (Code Division Multiple Access) system is interference limited, so it's necessary a call admission control to guarantee a level of quality of service and a grade of service. In this paper we present an analytical study about two types of admission controls, one is based on the number of users and the other is based on the interference level. In this study we develop expressions for the blocking rate and the loss probability of communication quality for each admission threshold. We apply the results to three typical systems, one traditional system, voice users only, one system with data users only and an integrated system, with heterogeneous traffic, voice and data users.

1. Introducción

Los sistemas móviles de tercera generación deberán emplear técnicas avanzadas de acceso múltiple para soportar la gran variedad de servicios futuros, con distintos requisitos de tasas de transmisión y calidad de servicio, QoS (Quality of Service). La técnica de acceso CDMA se ha convertido en un serio candidato para estos sistemas de tercera generación [1], [2], [3]. La capacidad de un sistema CDMA está limitada únicamente por la interferencia total de todos los usuarios conectados. Una de sus principales ventajas es su elevada utilización del ancho de banda, que es un recurso escaso en la mayoría de los sistemas radio. Este elevado uso del ancho de banda se debe a la multiplexación estadística de señales con requisitos de transmisión variables, sin necesidad de métodos complejos de compartición del canal como en FDMA (Frequency Division Multiple Access) o TDMA (Time Division Multiple Access). En estos sistemas CDMA, basados en la multiplexación estadística, existe una relación entre la capacidad del sistema y la calidad de la comunicación. Para garantizar un cierto nivel de calidad de comunicación, es necesario limitar el número de usuarios en una estación base. Por este motivo el control de admisión es clave en los sistemas CDMA.

El control de admisión se diseña para garantizar un grado de servicio GoS (Grade of Service), por ejemplo tasa de bloqueo, y una calidad de servicio, por ejemplo probabilidad de pérdidas. Aunque existen diversas referencias en las que se trata del establecimiento de umbrales de admisión [4], [5], [6], es todavía un problema de diseño importante que estos umbrales satisfagan tanto el GoS como la QoS. Pero en CDMA hay otro problema de diseño además del nivel GoS/QoS, que es la máxima capacidad en Erlang de cada celda. Las estrategias de admisión analizadas se clasifican en dos tipos: las primeras basadas en el número de usuarios, NCAC (Number of users Call Admission Control) [5], [7] y un segundo tipo según el nivel de interferencias,

ICAC (Interference Call Admission Control) [4], [6]. En este trabajo, se analiza la máxima capacidad del sistema para un determinado nivel de calidad de comunicación, y en función de ésta, se estudia cómo establecer el umbral en los dos tipos de controles de admisión mencionados anteriormente, para conseguir una determinada tasa de pérdidas (QoS). Además para ambos tipos de controles se formula la probabilidad de bloqueo para cada umbral como medida del GoS. Dada la importancia de la integración de voz y datos en los futuros sistemas móviles, este estudio se orienta hacia tres tipos de sistemas, un sistema tradicional que cuente sólo con usuarios de voz de baja tasa, un sistema que cuente sólo con usuarios de datos de alta tasa y un sistema integrado de voz y datos.

Este documento se organiza como sigue. En la sección 2 se describen los sistemas analizados, seguidamente en las secciones 3 y 4 se realiza un estudio de la capacidad y se modela la interferencia entre celdas, finalmente en el apartado 5 se describen los parámetros del análisis. Las secciones 6 y 7 presentan el estudio de los dos tipos de controles de admisión. Finalmente en el punto 8 se presentan las conclusiones.

2. Descripción de los sistemas

En esta sección se describen las características de los tres entornos bajo estudio, sólo voz, sólo datos, voz y datos.

Sistema tradicional de sólo voz (baja tasa de bits):

- Las llamadas son aleatorias con tasas de inicio y fin λ_0 y μ_0 respectivamente.
- Se emplea un mecanismo de detección de la actividad de los usuarios. Un usuario está activo con una probabilidad ρ (factor de actividad), independientemente del resto de usuarios.
- Todos los usuarios requieren la misma tasa de bits (tasa básica) y el mismo SIR (Signal to Interference Power Ratio).

- La potencia transmitida por cada móvil se controla perfectamente en la estación base, basándose en el nivel de potencia recibido.

En este caso se usa sólo el SIR medio a corto plazo, sin especificar los algoritmos de control de potencia TCP (Transmit Power Control) que se clasifican en dos tipos: compensación del *fast fading* y compensación de la fluctuación del valor medio [8].

Sistema de sólo datos (alta tasa de bits):

- En este caso se van a considerar datos en tiempo real de elevada tasa, que debido a sus fuertes limitaciones de retardo, usan el modo circuito en CDMA, como la voz.
- Las características de este entorno son similares al anterior, llamadas aleatorias (tasas λ_o y μ_o), mecanismo de detección de la actividad de los usuarios (ρ , factor de actividad), todos los usuarios con igual tasa de bits y SIR, control perfecto de potencia. La principal diferencia respecto al escenario de voz es una tasa de bits mucho mayor, lo que implica una ganancia de proceso menor.

Sistema integrado, voz (baja tasa) y datos (alta tasa)

- Los usuarios de voz y datos usan el modo circuito porque ambos son servicios en tiempo real con fuertes limitaciones de retardo.
- La tasa de los usuarios de datos es f_d veces mayor que la de voz, por tanto sus ganancias de proceso son distintas, así:

$$G_v = \frac{W}{R} \quad G_d = \frac{W}{f_d \cdot R} = \frac{G_v}{f_d}$$

- Las llamadas de voz y datos son aleatorias con tasas de inicio y fin [λ_{voz} , μ_{voz}] y [λ_{datos} , μ_{datos}] respectivamente.
- Se usa un mecanismo de detección de la actividad tanto para los usuarios de voz como para los de datos (ρ_v y ρ_d factores de actividad de voz y datos)
- Los dos tipos de usuarios requieren el mismo SIR para simplificar el estudio.
- Existe control perfecto de potencia para los usuarios de voz y datos.

3. Estudio de la capacidad

La capacidad máxima de una celda viene limitada por el nivel de calidad de comunicación exigido, esto es el máximo nivel de interferencias sobre un usuario, $I_{o,max}$. Las interferencias sobre un usuario son:

$$\frac{E_b \cdot (k-1)}{G_p} + N_o + I_{otras} \leq I_{o,max}$$

siendo E_b : Energía de bits, k : número de usuarios activos, N_o : densidad de potencia de ruido térmico, I_{otras} : Interferencia de otras celdas, G_p : Ganancia de proceso

Para investigar la calidad de una comunicación según la teoría de teletráfico, se transforma la interferencia de otras celdas en un número equivalente de usuarios m , así: $m = \frac{I_{otras} \cdot G_p}{E_b}$

Resolviendo la ecuación anterior con este m equivalente, se llega a:

$$c_{max} = k + m \leq \frac{G_p \cdot (1 - \eta^{-1})}{E_b / I_{o,max}} + 1$$

siendo $\eta = \frac{I_{o,max}}{N_o}$, que controla la máxima

potencia de transmisión del móvil. En esta capacidad máxima del sistema, para un determinado nivel $E_b/I_{o,max}$, aparecen las constantes m (usuarios equivalentes de otras celdas activos), que es un número real positivo, y k (usuarios activos en la celda local), que es un número entero positivo. La interferencia de otras celdas se modelará según su función densidad de probabilidad $p_{int}(m)$, que se describirá en una sección posterior. Para el caso de una sola celda, m es 0, por lo que $c_{max} = k$, que representa el máximo número aceptable de usuarios cuando $\rho=1$.

La diferencia fundamental en cuanto a la capacidad máxima del sistema de sólo voz y el sistema de sólo datos se debe a las distintas tasas de bits. Por este motivo, si la tasa de los datos es f_d veces mayor que la de la voz, la máxima capacidad del sistema de datos será menor que la del sistema de voz así:

$$C_{max} = k + m \leq \frac{G_d \cdot (1 - \eta^{-1})}{E_b / I_{o,req}} + 1 \quad G_d = \frac{G_v}{f_d}$$

En el sistema integrado, se van a estudiar tanto las interferencias sufridas por un usuario de voz como las de un usuario de datos, para calcular la máxima capacidad que asegura el nivel de calidad de la comunicación. Para simplificar este estudio se ha considerado que los niveles de calidad exigidos por los servicios de voz y datos son los mismos, pero en caso de considerarlos distintos (los datos con un mayor nivel de calidad, menores pérdidas) bastaría con repetir el siguiente análisis para dos factores distintos $E_b/I_{o,max,voz}$ y $E_b/I_{o,max,datos}$.

- Interferencia sobre un usuario de voz:

$$\frac{E_b \cdot (k_v - 1)}{G_v} + \frac{E_b \cdot k_d}{G_d} + N_o + I_{otras} \leq I_{o,req}$$

E_b : Energía de bits, es la misma para los usuarios de voz y datos

N_o : densidad de potencia de ruido térmico

I_{otras} : Interferencia de otras celdas, se debe a los usuarios de voz y datos activos en otras celdas

k_v : número de usuarios activos de voz

k_d : número de usuarios activos de datos

G_v, G_d : Ganancia de proceso de la voz y de los datos

El número equivalente de usuarios en otras celdas

$$\text{es: } I_{\text{other}} = \frac{E_b \cdot m_v}{G_v} + \frac{E_b \cdot m_d}{G_d}$$

Resolviendo la ecuación se llega a:

$$k_v + f_d \cdot k_d + m_v + f_d \cdot m_d \leq \frac{G_v \cdot (1 - \eta^{-1})}{E_b / I_{o,req}} + 1$$

En esta fórmula se aprecia que cada usuario de datos ocupa la capacidad relativa a f_d usuarios de voz, dado que su tasa es f_d veces mayor.

- Interferencia sobre un usuario de datos:

$$\frac{E_b \cdot K_v}{G_v} + \frac{E_b \cdot (k_d - 1)}{G_d} + N_o + I_{\text{otras-celdas}} \leq I_{o,req}$$

Resolviendo queda:

$$k_v + f_d \cdot k_d + m_v + f_d \cdot m_d \leq \frac{G_v \cdot (1 - \eta^{-1})}{E_b / I_{o,req}} + n$$

Después de este análisis, se observa que el nivel máximo de interferencias para un usuario de voz, es el que más limita y por tanto es el que va a fijar la capacidad máxima del sistema. Si se considerara que los datos requieren una mayor calidad de comunicación, dado que son más sensibles a errores que la voz, el nivel $I_{o,max}$ para los datos sería menor que el de la voz. Con esto la capacidad máxima quedaría más limitada por los datos. Como se ha comentado anteriormente, en este caso se ha efectuado la simplificación de suponerlos iguales.

4. Modelado de la interferencia de otras celdas

En este trabajo la media y la varianza de la interferencia de otras celdas, medida en número equivalente de usuarios m , se aproxima por la interferencia en la misma celda multiplicada por un factor f , [4]. En [8] se propone introducir la probabilidad de bloqueo en esta aproximación, de forma que: $E[m] = a \cdot \rho \cdot (1 - P_{\text{block}}) \cdot f1$

$$\text{Var}[m] = a \cdot \rho \cdot (1 - P_{\text{block}}) \cdot f2$$

La distribución usada para modelar esta interferencia es la distribución gamma, porque como se sugiere en [8], modela mejor que la distribución gaussiana los procesos aleatorios no negativos, y el nivel de interferencia es no negativo. La función densidad de probabilidad de la distribución gamma es:

$$p(m) = \frac{1}{\tau(v)} \alpha^v m^{v-1} e^{-\alpha \cdot m}$$

cuya media y varianza son:

$$E[m] = \frac{v}{\alpha} \quad \text{Var}[m] = \frac{v}{\alpha^2}$$

Los parámetros α y v de esta distribución gamma se extraen de igualar las ecuaciones anteriores así:

$$E[m] = a \cdot \rho \cdot (1 - P_{\text{block}}) \cdot f1 = \frac{v}{\alpha}$$

$$\text{Var}[m] = a \cdot \rho \cdot (1 - P_{\text{block}}) \cdot f2 = \frac{v}{\alpha^2}$$

De estas igualdades se deduce que:

$$\alpha = \frac{f1}{f2} = \text{cte} \quad v = \alpha \cdot E(m) = \alpha \cdot a \cdot \rho \cdot (1 - P_{\text{bloq}}) \cdot f1$$

es decir, v depende del tráfico a , del factor de actividad ρ y de la P_{bloqueo} , en cambio α es contante.

Los valores usados para los factores $f1$ y $f2$ se han tomado de [8], en el que se indican unos valores $f1=0.57$ y $f2=0.22$ como resultado de simulaciones efectuadas con el modelo de propagación siguiente:

$$P_{\text{recibida}} = P_{\text{transmitida}} \cdot r^{-\alpha} \cdot 10^{\xi/10}$$

donde α es la constante de atenuación con la distancia y ξ es variable gaussiana del "lognormal shadowing", de media cero y desviación estándar σ . En la simulación se usaron una cte de atenuación α de 4 y una desviación estándar del shadowing σ de 8 dB. En todos los sistemas que hemos analizado se han usado las mismas constantes, que dependen principalmente del modelo de propagación.

Las integrales de la pdf de interferencia entre celdas se pueden resolver como:

$$\int_{Q-G}^{\infty} p_{\text{int}}(m) \cdot dm = 1 - \text{gammainc}(\alpha \cdot (Q - G), v)$$

5. Parámetros del análisis

En el análisis de los controles de admisión que se realiza en las secciones posteriores se han usado los siguientes parámetros del sistema:

$$f1 = 0.57, f2=0.22, E_b/I_{o,max} = 3 \text{ dB}, \eta = 10 \text{ dB}$$

$$W_{\text{UTRA}} = 4096 \text{ kcps}, P_{\text{bloq}}=1\%$$

El ancho de banda considerado es el correspondiente al sistema UTRA. En cuanto a los tráficos de datos se han analizado dos tasas distintas, ambos con una ρ parecida a la de la voz. Estos datos podrían ser videoconferencias en tiempo real con tasas de alrededor de 28.8 Kbps y 64 kbps

Tabla1: Parámetros de tráfico

	G_p	Tasa	f_d	c_{max}	Tráf	ρ	α	v
Voz	128	32	1	58	60,	0.4	2.59	35.08
					70			40.92
					80			46.78
					90			52.6
					100			58.46
Datos1	64	64	2	29	10	0.45		6.57
					20			13.15
					30			19.73
					40			26.3
					53.3			35.1
Datos2	32	128	4	15	10	0.5	2.59	7.3
					15			10.96
					20			14.61
					30			21.92
					40			29.23

6. Control de admisión basado en el número de usuarios

6.1. Estudio analítico

En este control, se establece un umbral de admisión de N usuarios, de manera que cualquier nueva llamada es bloqueada si ya existen N usuarios

ocupando una estación base. A continuación se calcula, para cada umbral de admisión, la tasa de bloqueo como medida del GoS y la probabilidad de pérdidas como parámetro de calidad de servicio.

- *Tasa de bloqueo*

En este caso basta con usar la fórmula de ErlangB, considerando que el número de servidores es el umbral de admisión N. Así:

$$P_B = \frac{a^N}{\text{fac}(N)} \frac{1}{\sum_{i=0}^N \frac{a^i}{\text{fac}(i)}}$$

siendo a = Tráfico ofrecido en Erlangs = λ_o / μ_o

- *Probabilidad de pérdidas:*

En este apartado se calcula la probabilidad de que no se obtenga el nivel de calidad requerido, es decir la probabilidad de que el número de usuarios activos supere la capacidad máxima para ese nivel de calidad ($E_b/I_{o,max}$). A tal efecto es necesario realizar los siguientes cálculos:

- P_k , probabilidad de k usuarios activos

$$P_K = \sum_{r=k}^N b(k; r, \rho) \frac{a^r}{\text{fac}(r)} \frac{1}{\sum_{i=0}^N \frac{a^i}{\text{fac}(i)}}$$

siendo $b(k; r, \rho)$ una distribución binomial, que indica la probabilidad de k usuarios activos de r usuarios existentes.

- Variable u, que indica si el número de usuarios activos equivalentes k+m excede la capacidad máxima. En ese caso vale 1 y 0 en otro caso.

- La probabilidad de pérdidas se calcula como el cociente de las veces que no se satisface el nivel de comunicación en el intervalo T, es decir se supera la capacidad máxima, entre el número total de transmisiones.

$$P_{pérdidas} = \frac{T \cdot E[uk]}{T \cdot E[k]} = \frac{T \sum_{k=0}^N k \cdot P_k \cdot \int_{c_{max}-k}^{\infty} p_{int}(m) \cdot dm}{T \sum_{k=0}^N k \cdot P_k}$$

La diferencia fundamental en el cálculo de estos parámetros en el *sistema tradicional de sólo voz* y el *sistema de sólo datos* reside en que un usuario de datos ocupa el equivalente a f_d usuarios de voz. Por este motivo si el umbral de admisión es N usuarios de tasa básica, para el caso de los datos el máximo es N/f_d , es decir:

$$P_{\text{Bloqueo, datos}} = \frac{a_d^{N_d}}{\text{fac}(N)} \frac{1}{\sum_{i=0}^{N_d} \frac{a_d^i}{\text{fac}(i)}} \quad N_d = \frac{N}{f_d}$$

$$P_{K, \text{datos}} = \sum_{r=k}^{N_d} b(k; r, \rho) \frac{a_d^r}{\text{fac}(r)} \frac{1}{\sum_{i=0}^{N_d} \frac{a_v^i}{\text{fac}(i)}}$$

$$P_{\text{pérdidas, datos}} = \frac{\sum_{k=0}^{N_d} k \cdot P_k \cdot \int_{c_{max}-f_d \cdot k}^{\infty} p_{int}(m) \cdot dm}{\sum_{k=0}^{N_d} k \cdot P_k}$$

En el *sistema integrado* de voz y datos, si el umbral de admisión es N esto significa que:

$$u_v + f_d \cdot u_d \leq N$$

siendo f_d el factor que indica el número de veces que es mayor la tasa de datos que la de voz, u_v usuarios de voz y u_d de datos. El objetivo en este sistema es:

1) Analizar, para un determinado umbral de admisión N, cómo varía la probabilidad de bloqueo y la tasa de pérdidas de la **voz**, al variar el n° de usuarios de datos en el sistema, tabla 2.

Tabla 2: Probabilidad de bloqueo y tasa de pérdidas de la voz en un sistema integrado.

Para un determinado umbral N
Para una carga de tráfico de voz
Variar el n° usuarios de datos n_d existentes en el sistema n_d de 0 a $N_d=N/f_d$
Calcular para cada valor de n_d , el umbral de admisión de voz: $N_{voz}=N-n_d \cdot f_d$
Calcular P_{bloqueo} de la voz para umbral N_{voz}
$P_B = \frac{a^{N_{voz}}}{\text{fac}(N)} \frac{1}{\sum_{i=0}^{N_{voz}} \frac{a^i}{\text{fac}(i)}}$
Calcular $P_{\text{pérdidas}}$ de la voz
$P_{\text{pérdidas}} = \frac{\sum_{k=0}^{N_{voz}} k \cdot P_k \cdot \int_{c_{max}-f_d \cdot n_d - k}^{\infty} p_{int}(m) \cdot dm}{\sum_{k=0}^{N_{voz}} k \cdot P_k}$
En este caso se ha supuesto que en el resto de celdas sólo hay usuarios de voz

2) Analizar de forma similar al caso anterior, para un determinado umbral de admisión N, cómo varía la probabilidad de bloqueo y la tasa de pérdidas de los **datos**, al variar el n° de usuarios de voz n_v de 0 a N. En este caso:

$$N_{\text{datos}} = (N - n_v) / f_d \quad P_B = \frac{a^{N_{\text{datos}}}}{\text{fac}(N)}$$

$$P_{\text{pérdidas}} = \frac{\sum_{k=0}^{N_{\text{datos}}} k \cdot P_k \cdot \int_{c_{\text{max}} - n_v - f_d \cdot k}^{\infty} p_{\text{int}}(m) \cdot dm}{\sum_{k=0}^{N_{\text{datos}}} k \cdot P_k}$$

En este caso se ha supuesto que en las otras celdas sólo hay usuarios de datos.

6.2. Resultados

Los umbrales de admisión usados han sido:

$$N/c_{\text{max}} = 1.2 \quad 1.4 \quad 1.6 \quad 1.8 \quad 2 \quad 2.2$$

En los resultados de las figuras 1 y 2 se puede observar cómo al aumentar el umbral de admisión disminuye la tasa de bloqueo, a costa de un inicial incremento en la probabilidad de pérdidas, que después de un determinado valor se estabiliza. Así por ejemplo para $a=70$ E, se observa que para umbrales de admisión mayores que 1.6 veces la capacidad máxima, la probabilidad de pérdidas se estabiliza en un 1%, de forma que la elección del mejor umbral se puede usar según la tasa de bloqueo deseada, por ejemplo si queremos una tasa menor que 10^{-5} bastaría con tomar un umbral mayor que 1.8. El uso combinando de estas dos gráficas permite la elección de umbrales de admisión que cumplan tanto los requisitos de GoS (bloqueo) como de QoS (pérdidas). En las figuras 3 y 4 se muestra cómo varía la tasa de bloqueo de los datos al variar la tasa de bits, es decir f_d , y por tanto la ganancia de proceso y la capacidad máxima del sistema. Por ejemplo para $a=70$ E se observa que en la figura 3 la tasa de bloqueo es de 10^{-7} , mientras que en la 4 ha aumentado hasta el 1%. En la figura 5 se observa cómo las pérdidas en este caso son mucho mayores que en el caso de la voz. En las figuras 6 y 7 se observa cómo varía la tasa de bloqueo de la voz al aumentar el número de usuarios de datos en un sistema integrado, y se pueden comparar los distintos ordenes de magnitud al incrementar el umbral de admisión. Por ejemplo para un tráfico de voz $a=70$ E, con 10 usuarios de datos en el sistema, se observa que para un umbral de 70, fig. 6, el bloqueo es del 30% mientras que para un umbral de 93, fig. 7, el bloqueo es del 7%. La figura 8 es similar a las anteriores, salvo que en este caso se observa la variación del bloqueo de los datos al aumentar los usuarios de voz en el sistema integrado. Las figuras 9 y 10 muestran las evoluciones de las pérdidas de voz y datos al variar el número de usuarios de datos y de voz respectivamente.

7. Control de admisión basado en el nivel de interferencias

7.1. Estudio analítico

En este control una nueva llamada es bloqueada si el nivel de interferencia observado excede el umbral T_{bloq} . Cada estación base mide la potencia total de las señales recibidas. T_{bloq} es el umbral que indica el máximo nivel de interferencia aceptable normalizado por E_b/G_p . Por tanto es un número adimensional, que se puede tratar como el número equivalente de usuarios. El inconveniente de este control es que se requiere hardware específico para la estación base y complica su arquitectura, mientras que el control anterior, basado en el número de usuarios, se puede implementar fácilmente por software. En el análisis se asume medidas de potencia sin error y estaciones base independientes.

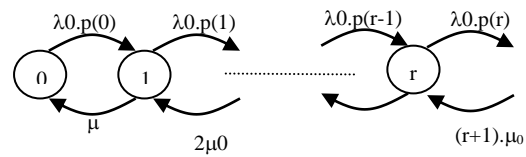


Fig 11. Diagrama de estados en la estación base

• Tasa de bloqueo:

En este caso el diagrama de transición de estados en estación base es el que se muestra en la figura 11, donde $p(j)$ es la probabilidad de que una nueva llamada sea aceptada si existen j usuarios. Resolviendo las ecuaciones de estado de este diagrama, se obtiene la $P_{r,\text{conect}}$, probabilidad de estado de que existan sólo r usuarios:

$$P_{r,\text{conect}} = \frac{\frac{a^r}{\text{fac}(r)} \prod_{j=0}^{r-1} p(j)}{1 + \sum_{k=1}^{\infty} \frac{a^k}{\text{fac}(k)} \cdot \prod_{j=0}^{k-1} p(j)}$$

La probabilidad de que una nueva llamada sea aceptada con j usuarios, $p(j)$ se obtiene como $1 - P_{\text{bloqueo}}$. La P_{bloqueo} es la probabilidad de que la interferencia de un nuevo usuario, que consiste en la interferencia de la misma celda y la de otras celdas, haga que la potencia interferente total en la estación base supere el umbral T_{block} .

$$P_{\text{bloqueo}}(j) = 1 - p(j) = \sum_{k=0}^j b(k; j, \rho) \cdot \int_{T_{\text{bloq}} - k}^{\infty} p_{\text{int}}(m) \cdot dm$$

$$p(j) = 1 - \sum_{k=0}^j b(k; j, \rho) \cdot \int_{T_{\text{bloq}} - k}^{\infty} p_{\text{int}}(m) \cdot dm$$

La probabilidad de bloqueo se obtiene como la suma de todos los posibles

$$\text{estados: } P_{\text{bloq}} = \sum_{r=0}^{\infty} P_{r,\text{conect}} \cdot \{1 - p(r)\}.$$

En este caso surge un problema porque $P_{r,\text{conect}}$ y $p(r)$ dependen de P_{bloq} , dado a que los parámetros de la $p_{\text{int}}(m)$ dependen de P_{bloq} . Por tanto P_{bloq} no se puede obtener directamente de esta fórmula. Lo que se

hace es iterar, asumiendo que las probabilidades de bloqueo medias son las mismas en todas las estaciones base. Se comienza por una $P_{\text{bloq. inicial}}$ suficientemente pequeña, con la que se calculan los parámetros α y ν de la $p_{\text{int}}(m)$, para calcular el término derecho de la ecuación. El resultado se compara con la $P_{\text{bloq. inicial}}$, si no difiere mucho, éste es el valor final, pero si difiere se toma el resultado obtenido como $P_{\text{bloq. inicial}}$ y se vuelve a iterar.

• *Probabilidad de pérdidas*

La probabilidad de que haya k usuarios activos P_k , se calcula usando la probabilidad de estados $P_{r,\text{conect}}$

así: $P_k = \sum_{r=k}^{\infty} P_{r,\text{conect}} \cdot b(k; r, \rho)$. La probabilidad de

pérdidas es el cociente de aquellos que no cumplen el nivel $E_b/I_{o,\text{max}}$ (capacidad máxima) frente al total:

$$P_{\text{pérdidas}} = \frac{\sum_{k=0}^{\infty} k \cdot P_k \cdot \int_{c_{\text{max}}-k}^{\infty} p_{\text{int}}(m) \cdot dm}{\sum_{k=0}^{\infty} k \cdot P_k} . \text{ La diferencia}$$

respecto al NCAC, es que el sumatorio es hasta infinito, dado que el número de usuarios no está limitado, lo que está limitada es la interferencia.

En los sistemas de sólo voz y sólo datos, la diferencia radica en que los usuarios de datos equivalen a f_d de voz, tasa f_d veces mayor, por eso interfieren f_d veces más. Es decir, la $p(j)$ de datos es:

$$p(j)_{\text{datos}} = 1 - \sum_{k=0}^j b(k; j, \rho) \cdot \int_{T_{\text{bloq}}-f_d \cdot k}^{\infty} p_{\text{int}}(m) \cdot dm$$

En el sistema integrado, el interés se centra en:

- 1) Para un umbral de admisión, T_{bloq} , analizar cómo varía la prob. de bloqueo y las pérdidas de la voz al variar el n° de usuarios de datos n_d en el sistema: $T_{\text{bloq,voz}} = T_{\text{bloq}} - f_d \cdot n_d$

$$p(j)_{\text{voz}} = 1 - \sum_{k=0}^j b(k; j, \rho) \cdot \int_{T_{\text{bloq,voz}}-n_d \cdot f_d - k}^{\infty} p_{\text{int}}(m) \cdot dm$$

$$P_{\text{pérdidas,voz}} = \frac{\sum_{k=0}^{\infty} k \cdot P_k \cdot \int_{c_{\text{max}}-f_d \cdot n_d - k}^{\infty} p_{\text{int}}(m) \cdot dm}{\sum_{k=0}^{\infty} k \cdot P_k}$$

- 2) Para un umbral de admisión, T_{bloq} , analizar cómo varía la prob. de bloqueo y las pérdidas de los datos al variar el n° de usuarios de voz n_v en el sistema. $T_{\text{bloq,datos}} = \frac{T_{\text{bloq}} - n_v}{f_d}$

$$p(j)_{\text{datos}} = 1 - \sum_{k=0}^j b(k; j, \rho) \cdot \int_{T_{\text{bloq,datos}}-n_v - f_d \cdot k}^{\infty} p_{\text{int}}(m) \cdot dm$$

$$P_{\text{pérdidas,datos}} = \frac{\sum_{k=0}^{\infty} k \cdot P_k \cdot \int_{c_{\text{max}}-n_v - f_d \cdot k}^{\infty} p_{\text{int}}(m) \cdot dm}{\sum_{k=0}^{\infty} k \cdot P_k}$$

7.2. Resultados

Los umbrales de admisión usados son 46 58 70 81. Los resultados de las figuras 12 y 13 muestran que, como ocurría en el NCAC, a partir de un cierto valor del umbral de admisión, la probabilidad de pérdidas se estabiliza, mientras disminuye la tasa de bloqueo. El interés de estas gráficas es demostrar cómo se pueden cumplir tanto los requisitos de GoS como de QoS para cada umbral de admisión. En las figuras 14 y 15, correspondientes al sistema integrado se puede observar la incidencia de la mayor interferencia de los datos respecto a la voz. Por ejemplo en un sistema con 4 usuarios de datos, un tráfico de voz de 60 E y un umbral de 46, la tasa de pérdidas de voz es 0.0002, mientras que en un sistema con 4 usuarios de datos, un tráfico de datos de 30 E y un umbral de 46, la tasa de pérdidas es 0.002, diez veces menor. En las figuras 16 y 17, se observa también la mayor interferencia de los datos respecto a voz en lo que se refiere a prob. bloqueo.

8. Conclusiones

El análisis efectuado permite realizar la elección de un umbral de admisión, en los controles de admisión basados en número de usuarios y en nivel de interferencias, que cumpla los requisitos de GoS y QoS para una determinada calidad de comunicación. Los tres entornos estudiados permiten efectuar una comparativa de características en función de la tasa de bits de los usuarios del sistema.

Agradecimientos

Este trabajo ha sido parcialmente financiado por la CICYT, proyecto TIC98- 1040.

Referencias

- [1] T. Ojanpera et al, "Comparison of multiple access schemes for UMTS", Proceedings on the IEEE Conference Veh. Technol. VTC'97, 1997, pp.490-494
- [2] T. Nakamura et al, "Configuration and characteristics estimation of a W-CDMA experimental system for third generation mobile communications", Proceedings on the IEEE Conference Veh. Technol VTC'98, 1998, pp. 973-977.
- [3] K.S. Gilhousen et al, "On the capacity of a cellular CDMA system". IEEE Tran. Veh. Technol., vol. 40, pp. 303-312, Mayo 1991.
- [4] A. M. Viterbi, A. J. Viterbi, "Erlang capacity of a power controlled CDMA system". IEEE J. Selected Areas Commun., vol 11, pp. 892-900, Agosto 1993.
- [5] B. Lavery, D. Everitt, "On the teletraffic characterization of cellular CDMA systems". Proceedings on the IEEE Veh. Technol. Conference VTC'98, 1993, pp. 416-419.
- [6] F. D. Prisolci, "Fixed and adaptative thresholds in CDMA cellular networks". Proceedings on the IEEE Veh. Technol. Conference VTC'95, 1995, pp. 1090-1094.
- [7] Z. Liu, M.E. Zarki, "SIR-based call admission control for DS-CDMA cellular systems". IEEE J. Selected Areas Commun., vol 12, pp. 638-644, Mayo 1994.
- [8] Y. Ishikawa, N. Umeda, "Capacity design and performance of call admission control in cellular CDMA systems". IEEE J. Selected Areas Commun., vol 15, pp. 1627-1635, Octubre 1997.

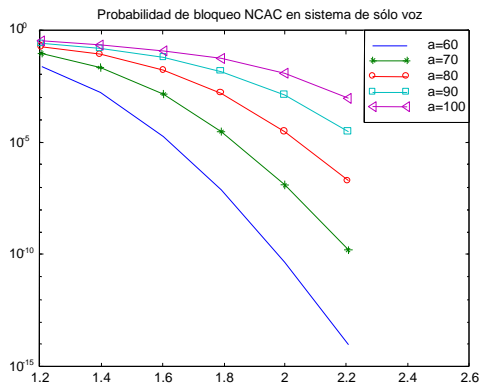


Fig. 1: Sistema sólo voz. NCAC. Bloqueo de la voz

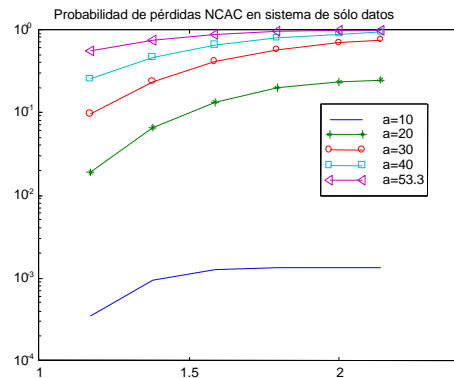


Fig. 5: Sistema sólo datos. NCAC. Pérdidas de los datos.

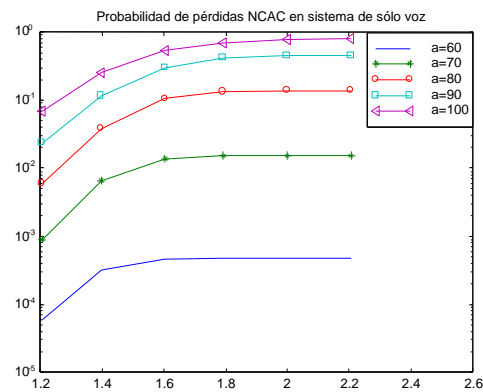


Fig. 2: Sistema sólo voz. NCAC. Pérdidas de la voz.

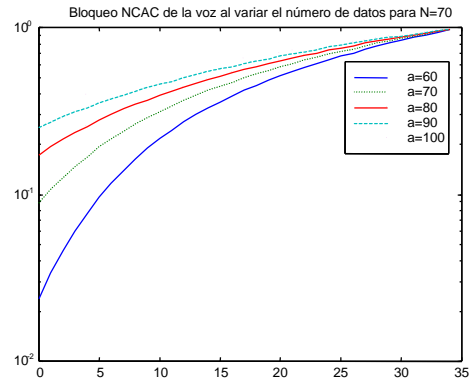


Fig. 6: Sistema integrado. NCAC. Bloqueo voz variando usuarios datos, N=70.

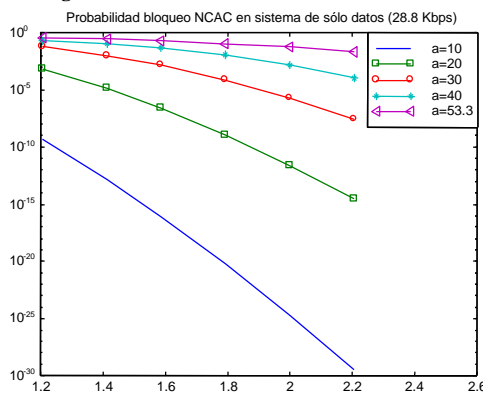


Fig. 3: Sistema sólo datos. NCAC. Bloqueo de los datos de 28.8 kbps.

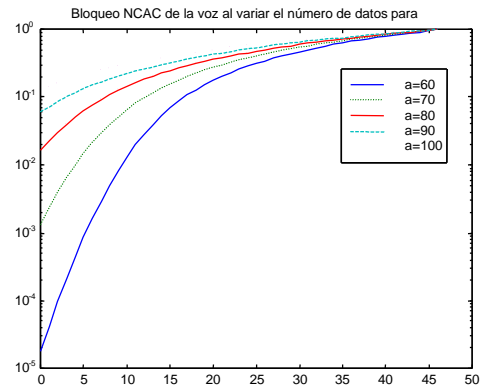


Fig. 7: Sistema integrado. NCAC. Bloqueo voz variando usuarios datos, N=93.

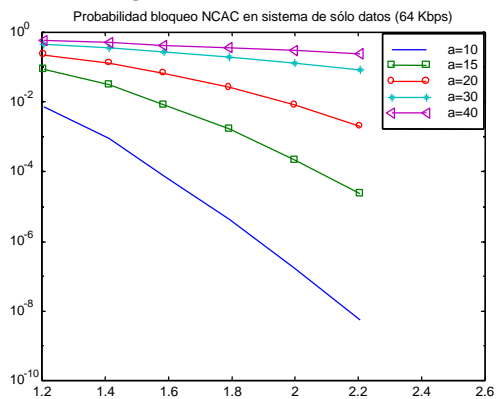


Fig. 4: Sistema sólo datos. NCAC. Bloqueo de los datos de 64 kbps.

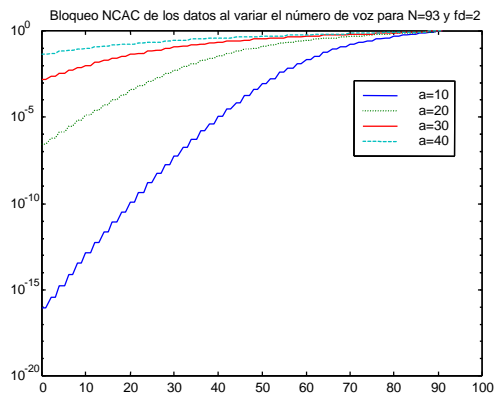


Fig. 8: Sistema integrado. NCAC. Bloqueo datos (28.8Kbps) variando usuarios voz, N=93.

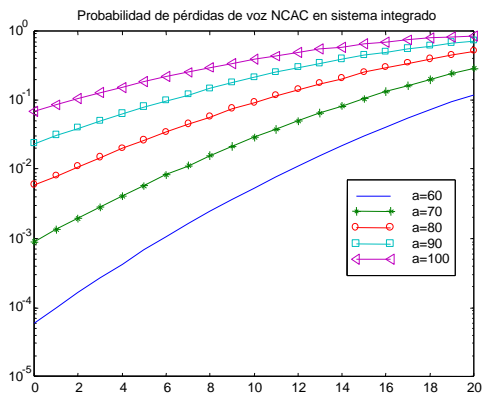


Fig. 9: Sistema integrado. NCAC. Pérdidas voz variando usuarios datos. N=70

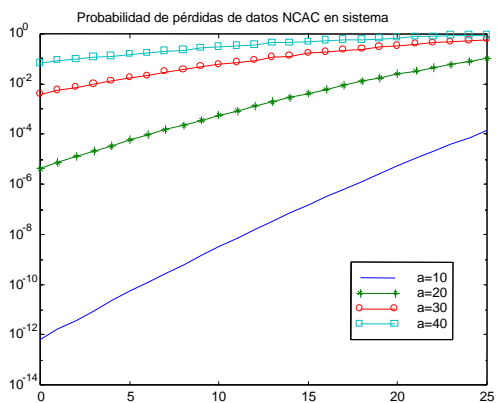


Fig. 10: Sistema integrado NCAC. Pérdidas datos variando usuarios voz. N=93

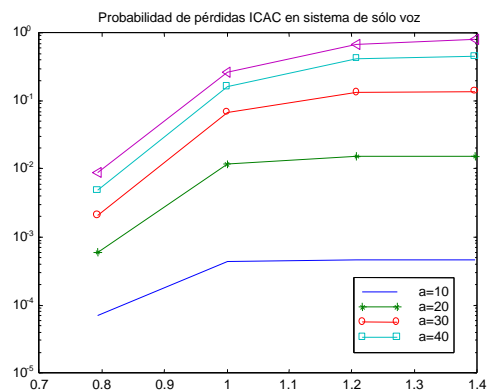


Fig. 12: Sistema sólo voz. ICAC. Pérdidas de la voz

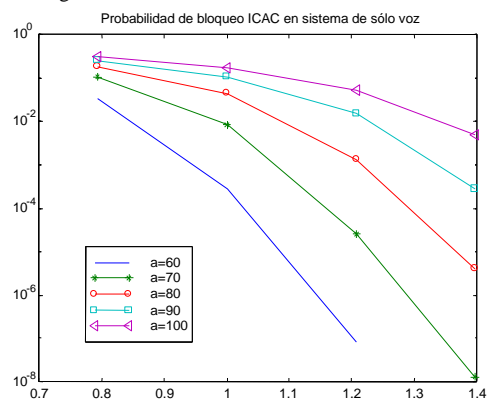


Fig. 13: Sistema sólo voz. ICAC. Bloqueo de la voz

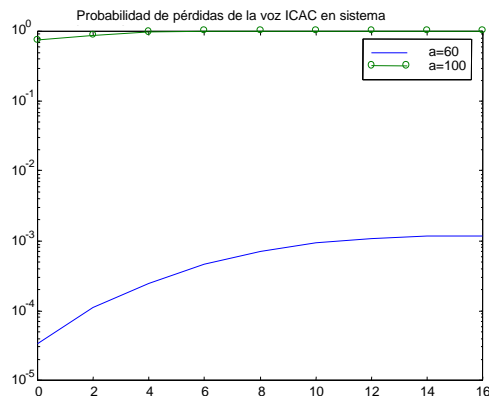


Fig. 14: Sistema integrado. ICAC. Pérdidas voz al variar el n° de datos, para (a=60, $T_{bloq}=46$) y (a=100, $T_{bloq}=81$).

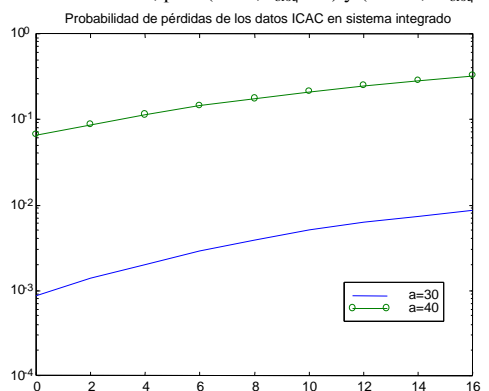


Fig. 15: Sistema integrado. ICAC. Pérdidas datos al variar n° voz, para a=30 y $T_{bloq}=46$ y a=40 y $T_{bloq}=58$.

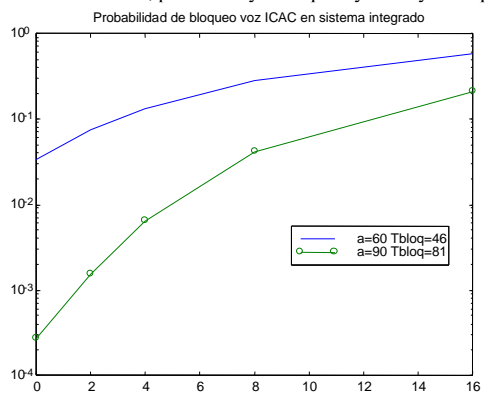


Fig. 16: Sistema integrado. ICAC. Bloqueo voz al variar n° datos, para a=60 y $T_{bloq}=46$ y a=90 y $T_{bloq}=81$.

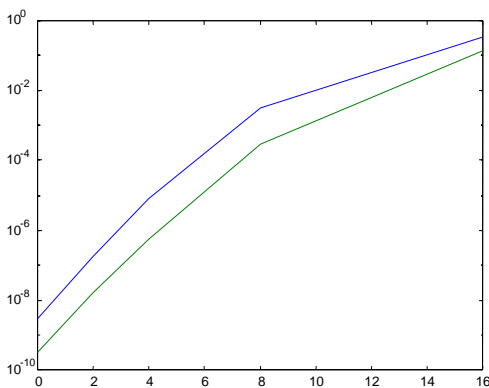


Fig. 17: Sistema integrado. ICAC. Bloqueo de los datos

Capítulo 2A: Implicaciones Socio-Económicas e Integración de Servicios

De la Ingeniería Telemática a la Ingeniería de la Sociedad de la Información.
M. Tomás, A. Alabau (U. Politécnica de Valencia)

Parámetros de medida de la calidad de servicio en ISPs y análisis comparativo de proveedores.
A. Ferro Vázquez (U. País Vasco)

ATM Sobre Ethernet Mediante el Protocolo CIF.
J. M. Arco, B. Alarcos, A. M. Hellín, A. García, D. Meziat (U. Alcalá)

Simulador ATM basado en Componentes.
J. M^a Quinteiro, M. Pilar Rosales, Á. Santana, A. Sarmiento (U. Las Palmas)

Funcionamiento y prestaciones del protocolo HARAL (Hybrid Adaptive Random Access Protocol) para redes de cable HFC.
L. Gutierrez y S. Sallent (U. Politécnica de Cataluña)

Control de Acceso en entornos Multicast: Un enfoque a la autenticación de emisores.
A. F. Gómez Skarmeta, A. L. Mateo Martínez, P. Miguel Ruiz Martínez (U. Málaga)

II Jornadas de Ingeniería Telemática JITEL 99

“De la Ingeniería Telemática a la Ingeniería de la Sociedad de la Información”

Manuel Tomás, Antonio Alabau
Área de Ingeniería Telemática. Universidad Politécnica de Valencia
Camino de Vera s/n, 46022 VALENCIA
Telf: 96-3879303, Fax: 96-3877199
E-mail: [mtomas, aalabau]@dcom.upv.es

Abstract

*As a result of the growing process of technology development, which has taken place during the last thirty years, **Telematic Engineering** sector has become a part of an emerging new sector, called **Information and Communications Technologies (ICT)**. Nowadays, the development of the ICT sector is a key policy issue supported by Governments of the most industrialised countries. To achieve this aim, the concept of **Information Society (IS)** has been created. Therefore, the evolution of Telematics in the near future will be strongly conditional on the target of Information Society deployment. In our mind, that foreseeable evolution involves the emergence of a new kind of engineering, that can be called **Information Society Engineering**. The arrival of this new kind of activity raises some key questions to solve. In this paper, we have tried to identify some of these issues and we propose some choices to begin to broach them.*

1. Introducción

Durante el último tercio del presente siglo, ha tenido lugar un proceso caracterizado por el acelerado desarrollo de la tecnología, el cual, según ciertos autores, tiene su origen en las enormes inversiones en investigación y desarrollo tecnológico (IDT) realizadas durante los años de la II Guerra Mundial.

Independientemente de cuál sea su origen, lo cierto es que este proceso se ha caracterizado por la velocidad de la renovación técnica, que en muchos casos no ha posibilitado siquiera la propia consolidación de dichos avances.

Por lo que se refiere al área específica de la *Ingeniería Telemática*, ésta ofrece un ejemplo muy apropiado para poder constatar lo anterior. El motivo de ello es que esta rama de la técnica ha surgido y se ha desarrollado durante los últimos 25 años, por lo que se da la circunstancia peculiar de que algunos de los profesionales actualmente en activo conservan la memoria histórica suficiente para poder analizar su evolución.

El interés de este análisis reside en que, a partir de él, puede obtenerse una perspectiva que permita realizar un ejercicio de prospección dirigido a determinar cuál puede ser el papel a jugar por la Ingeniería Telemática de cara a los primeros años del siglo XXI, en el momento en el que se está abriendo claramente el camino hacia la llamada *Sociedad de la Información*.

Esto es precisamente lo que se ha pretendido abordar en este artículo.

2. Análisis de los antecedentes

Como muchos recordarán, el término **Telemática** fue acuñado en 1970, al aparecer por vez primera en el Informe *Nora-Minc* publicado ese año. El surgimiento de este término quiso significar la aparición de un nuevo sector de actividad tecnológica, nacido de la fusión de dos sectores hasta entonces independientes: las *Telecomunicaciones* y la *Informática*.

A este respecto, cabe destacar que la integración de las Telecomunicaciones y la Informática fue un hito fundamentalmente tecnológico, consecuencia de la convergencia de las tecnologías de base empleadas en ambos sectores [1].

Durante las décadas de 1970 y 1980, tuvo lugar un fuerte empuje de la IDT en el campo de la Ingeniería Telemática, al ser considerada una fuente potencial de negocio. Fue precisamente en la década de los 80 cuando la disciplina de la Ingeniería Telemática se incorporó a los departamentos universitarios en España —con la creación de un Área de Conocimiento específica con esa denominación.

Asimismo, en los últimos años de la década de los 80 y, sobre todo, durante los 90, se ha producido un fuerte tirón en la demanda de profesionales formados en este ámbito. Una de las consecuencias más notables de ello ha sido la proliferación de titulaciones universitarias específicas de Ingeniería Telemática, así como la incorporación de especialidades con esta denominación vinculadas a un título universitario, existente desde mucho tiempo atrás, que es el de Ingeniero de Telecomunicación.

Dado que el proceso de desarrollo tecnológico que dio origen a la Ingeniería Telemática no sólo no ha finalizado sino que, por el contrario, se ha

intensificado, nos encontramos ante la conveniencia de realizar un ejercicio de replanteamiento del ámbito y funciones de la Telemática tal y como han sido concebidas en el pasado reciente. Y más aún, es lógico que esta reflexión sea liderada desde la Universidad, la cual tiene la responsabilidad de atender la demanda de profesionales que presenta la sociedad que la financia.

3. El concepto de “Sociedad de la Información”

Como se ha señalado anteriormente, la evolución de la tecnología propició a principios de 1970 el nacimiento de la Telemática, como integración de los sectores de las Telecomunicaciones y la Informática.

Dado que este proceso se intensificó durante esa década y la siguiente, a finales de los 80 tuvo lugar otro fenómeno similar. Los avances tecnológicos permitieron en esta ocasión la integración de tres sectores que habían permanecido tradicionalmente disjuntos: las Telecomunicaciones, la Informática y el Audiovisual. O lo que es lo mismo, la Telemática junto con las aplicaciones y los contenidos de información, a los que se ha unido el sector de la Electrónica de consumo.

Este fenómeno de convergencia tecnológica se ha materializado en el surgimiento de un nuevo sector económico, que engloba a los anteriores, el cual ha recibido el nombre de *macrosector de las Tecnologías de la Información y las Comunicaciones (TIC)*, o también, *Hipersector de la Información* [2].

Sin embargo, el nacimiento del sector de las TIC, aunque es un acontecimiento fundamentalmente tecnológico, no puede ser analizado independientemente del contexto en el que ha tenido lugar.

En este sentido, es esencial tener presente que el surgimiento del sector de las TIC se ha producido en un momento histórico en el que han cobrado fuerza las corrientes de pensamiento que abogan por la globalización de la economía, la mundialización de la actividad empresarial y la liberalización general de todos los sectores económicos [2]. Adicionalmente, esto se ha dado coincidiendo con una profunda crisis estructural en la que se halla sumida la sociedad industrial de final de este siglo y que, según algunas teorías macroeconómicas ampliamente aceptadas, parece indicar el agotamiento del modo de producción industrial [3].

En este contexto, las TIC parecen significar una posible solución a la crisis económica y problemas de desempleo que aquejan al actual modelo de sociedad, al convertirse en el nuevo motor que impulse el crecimiento de la economía general de cara al siglo XXI [3] [4].

Sobre la base de lo anterior, puede comprenderse el firme respaldo otorgado por los dirigentes políticos y económicos de las principales potencias mundiales al nacimiento y desarrollo de un verdadero mercado global de las TIC, sobre el cual sustentar la expansión del Sector y, con él, del resto de los sectores económicos [5].

Ahora bien, el apoyo político al desarrollo y consolidación de un fuerte mercado mundial para las TIC ha requerido de ciertos instrumentos para hacerse efectivo.

En concreto, una de las fórmulas elegidas para ello ha consistido en impulsar desde los gobiernos la idea de la creación de la **Sociedad de la Información** [4] [5].

El concepto de la Sociedad de la Información se encuentra sustentado sobre un nuevo modelo de sociedad, basado en la producción y consumo masivos de información y, por consiguiente, de sus tecnologías de soporte. El propósito de ello es estimular la demanda, favoreciendo la creación de la masa crítica necesaria que justifique las enormes inversiones que el sector privado ha de realizar para desarrollar todas las posibilidades de negocio abiertas por las nuevas tecnologías [4] [5] [6].

Como consecuencia de lo anterior, aparece un aspecto de capital importancia, el cual no puede ser ignorado: para que la Sociedad de la Información sea una realidad, las TIC deben pasar a formar parte de la vida cotidiana de los ciudadanos y de las empresas, impregnando profundamente todos sus ámbitos de actividad [5] [6] [7].

Por lo que se refiere al escenario europeo, en el cual nos encontramos inmersos, una prueba efectiva de ello la constituye la intensa actividad que las instituciones de la Unión Europea están desplegando en torno a la Sociedad de la Información [8].

A este respecto, cabe hacer especial mención del diseño que se ha concedido al 5º Programa Marco de IDT de la Unión Europea —que abarca los fondos públicos destinados a I+D para el período 1999-2002. La firme voluntad política de conformar la Sociedad de la Información en Europa se manifiesta claramente en él con la aparición de un programa temático específico titulado “*Creación de una Sociedad de la Información amigable para el usuario*”, reforzado con una dotación presupuestaria cercana a los 3600 millones de euros —el 24,1 % del total [9].

Concretamente, los objetivos de dicho programa temático se centran en “*hacer más acogedora la Sociedad de la Información, mejorando la accesibilidad, la pertinencia y la calidad de los servicios públicos, capacitando a los ciudadanos en su condición de trabajadores, empresarios y*

clientes; favoreciendo la creatividad y facilitando el acceso a la formación; contribuyendo a crear una Sociedad de la Información coherente y no excluyente, al garantizar la disponibilidad del servicio universal, el carácter intuitivo de las interfaces de usuario de nueva generación; en definitiva, fomentando el diseño para todos” [10].

De lo anterior resulta evidente que, en el contexto de la Sociedad de la Información, el interés estratégico se encuentra gravitando en torno a las **Aplicaciones**, mucho más que sobre las tecnologías concretas que las soportan [9] [10]. De otra forma, la tecnología sólo tiene interés en la medida en que se traduce directamente en aplicaciones útiles para los usuarios —esto es, en el grado en que puede generar nuevos segmentos de mercado todavía por explotar.

En nuestra opinión, esto es precisamente lo que las instituciones de gobierno de la Unión han querido poner de manifiesto al aprobar tal estructura de los fondos de IDT para el próximo cuatrienio.

4. Análisis de la situación actual

Observando todas estas cuestiones a la luz del modelo de evolución descrito para el caso de la Ingeniería Telemática, y teniendo bien presente que la progresión de los acontecimientos es cada vez más acelerada, el estado de madurez de la tecnología permite pensar que, en el momento presente, podemos estar en el inicio de una etapa en la que van a comenzar a demandarse en el mercado profesionales formados específicamente en el campo de las tecnologías y aplicaciones de la Sociedad de la Información —esto es, lo que podemos llamar **Ingenieros de las Aplicaciones de la Sociedad de la Información** o, simplemente, **Ingenieros de la Sociedad de la Información**.

Debido al intrínseco carácter multidisciplinar de las aplicaciones de la Sociedad de la Información, es fácil intuir que concurrirán profesionales con muy diversos perfiles curriculares para ejercer las funciones de estos “Ingenieros de la Sociedad de la Información”.

Sin embargo, en nuestra opinión, los Ingenieros Telemáticos disponen de una considerable ventaja competitiva respecto de otros profesionales para llegar convertirse en los futuros Ingenieros de la Sociedad de la Información.

Esta ventaja consiste básicamente en que estos poseen un sólido conocimiento de las tecnologías sobre las que deberán soportarse las aplicaciones de la Sociedad de la Información. Este conocimiento constituye un ingrediente imprescindible para poder desarrollar adecuadamente la función reservada a los Ingenieros de la Sociedad de la Información.

En este punto, merece la pena señalar que esta afirmación viene avalada por lo sucedido

precisamente en el campo de la Ingeniería Telemática en el pasado más reciente. Cabe recordar que, en su momento, la demanda de Ingenieros Telemáticos fue cubierta por aquellos profesionales que dominaban las tecnologías soporte —esto es, de un lado los Ingenieros de Telecomunicación, y de otro, los especialistas en Informática.

No obstante, la naturaleza intrínseca de la Sociedad de la Información hace que el conocimiento profundo de la tecnología sobre la que ésta se basa sea un requisito necesario, aunque no suficiente, en el bagaje profesional característico de los futuros Ingenieros de la Sociedad de la Información.

En este sentido, para que los Ingenieros Telemáticos puedan desempeñar adecuadamente la función de Ingenieros de la Sociedad de la Información, resulta absolutamente necesario complementar su formación universitaria incorporando el estudio de ciertas materias que han sido tradicionalmente ajenas a sus esquemas formativos.

Con relación a ello, la cuestión clave ahora es determinar qué materias y en qué proporción deben ser incluidas por la Universidad en los planes de estudio reglados.

A este respecto, nuestra opinión es que, debido a la fuerte orientación al usuario final y al carácter multidisciplinar de las aplicaciones que impone la Sociedad de la Información, las materias de estudio dirigidas a complementar la formación del Ingeniero Telemático deben ser por tanto de tipo humanístico.

En concreto, entendemos extremadamente oportuno introducir contenidos relativos a los aspectos estratégicos, políticos, socioeconómicos y reglamentarios relacionados con la Sociedad de la Información, para así complementar adecuadamente las materias tradicionales, centradas en cubrir los conocimientos relativos a las tecnologías que sirven de soporte a ésta.

Por lo que se refiere a actuaciones concretas, merece la pena señalar que la E.T.S. de Ingenieros de Telecomunicación de la Universidad Politécnica de Valencia (UPV) ha puesto en marcha una iniciativa singular en esta línea.

Este Centro, que fue pionero al introducir el estudio de la materia de Política de Telecomunicaciones — que ha acabado convirtiéndose en materia troncal en todos los planes de estudio de la carrera de Ingeniero de Telecomunicación en la Universidad española—, se encuentra desplegando el 2º ciclo del Plan de Estudios aprobado en 1996, el cual contiene una especialidad o intensificación específica de Ingeniería Telemática.

Dentro del marco correspondiente a dicha intensificación, la Escuela —a propuesta del Área de Ingeniería Telemática del Departamento de

Comunicaciones de la UPV—, ha entendido oportuno contemplar tres perfiles formativos: el ingeniero de las Redes Públicas, el ingeniero de las Comunicaciones Corporativas y el ingeniero de las Aplicaciones de la Sociedad de la Información [11].

Con relación a ello, merece la pena hacer notar que cada uno de los perfiles contemplados se encuentra en sintonía con una actividad profesional razonablemente bien definida de entre las que actualmente está demandando el mercado. Éstas corresponden, respectivamente, al ingeniero de Telecomunicación tradicional, al ingeniero Telemático original y, como apuesta de futuro, al ingeniero orientado al desarrollo de la Sociedad de la Información.

5. Conclusiones

Como resultado del intenso desarrollo de la tecnología que ha tenido lugar en las últimas décadas, el sector de la Ingeniería Telemática ha quedado integrada en el seno de un sector económico más amplio, que ha recibido el nombre de macrosector de las Tecnologías de la Información y las Comunicaciones (TIC).

Como elemento clave para el desarrollo y consolidación del sector de las TIC, ha surgido la iniciativa política de implantación de la Sociedad de la Información.

Por su parte, el interés estratégico de la Sociedad de la Información radica ante todo en las aplicaciones de las nuevas tecnologías de cara al usuario final, ya sea éste ciudadano o empresa, como medio para incrementar el crecimiento, la competitividad y el empleo.

En este contexto, parece razonable intuir que la evolución del profesional de la Ingeniería Telemática puede dirigirse hacia el *Ingeniero de las Aplicaciones de la Sociedad de la Información*. En cierto modo, esta evolución vendría a constituir el paso siguiente a la transformación del Ingeniero de Telecomunicación tradicional en el Ingeniero Telemático, realizada en su momento como respuesta a las necesidades presentadas por el mercado.

Como es natural, la puesta en escena de esta evolución plantea una multiplicidad de cuestiones y de retos, sobre los cuales hemos tratado de arrojar un poco de luz a lo largo de estas líneas.

Sea como fuere, lo que está claro es que todavía queda mucho por reflexionar en este sentido, pues aún nos encontramos en el principio del camino. Algunos hemos comenzado ya la andadura, pero entendemos que esta labor requiere del concurso de muchos más.

En este sentido, nos encontramos completamente abiertos a establecer relaciones de colaboración para trabajar conjuntamente con todos aquellos que sientan idénticas inquietudes, así como para ofrecer nuestro apoyo a los que pretendan iniciar actuaciones en esta misma dirección —de la misma forma que hicimos en su día con respecto a la materia de Política de Telecomunicaciones.

Referencias

- [1] Alabau, A. “*Teleinformática y Redes de Computadores*”. Editorial Marcombo. 1ª edición. 1982.
- [2] Alabau, A. “*La Unión Europea y su Política de Telecomunicaciones. Un camino hacia la Sociedad de la Información*”. Editorial Fundación Airtel Móvil. 1998.
- [3] Valéry, N. et al. “*Innovation in Industry*”. The Economist. February, 20TH, 1999.
- [4] “*Crecimiento, Competitividad y Empleo. Retos y pistas para entrar en el Siglo XXI*”. COM(93) 700. Libro Blanco. Comisión Europea. Bruselas, 5 de diciembre de 1993.
- [5] “*Conclusions of G7 Summit on Information Society*”. Brussels, February, 20TH, 1995.
- [6] Castells, M. “*La Era de la Información*”, vol 1º. Alianza Editorial. 1997.
- [7] “*Europa y la Sociedad Global de la Información. Recomendaciones al Consejo Europeo*”. Comisión Europea. Bruselas, 26 de mayo de 1994.
- [8] <http://www.ispo.cec.be>
- [9] <http://www.cordis.lu/fp5>
- [10] <http://www.cordis.lu/ist>
- [11] “*Propuesta de desarrollo de la Intensificación de Ingeniería Telemática correspondiente al 2º Ciclo del Plan de Estudios de 1996 en la ETSIT de la UPV*”. Área de Ingeniería Telemática. Departamento de Comunicaciones. Universidad Politécnica de Valencia. Febrero de 1999.

Parámetros de medida de la calidad de servicio en ISPs y análisis comparativo de proveedores

Armando Ferro, Marivi Higuero

{jtpfevaa|jtphiapm}@bi.ehu.es

Grupo de Ingeniería Telemática

Escuela Técnica Superior de Ingenieros Industriales y de Telecomunicaciones de Bilbao

September 15th, JITEL '99

Abstract

This paper presents a strategy in the definition of quality measurements of the service provided by an ISP (Internet Service Provider). As having access to Internet is becoming more important in every professional environment, disposing of a good quality service is also increasing in importance. From the study of the several elements implied in an ISP infrastructure, and taking into account the most usual user scenarios, the objective is to obtain a quality measurement of service nearest the user demands, consisting mainly of a good user service, high rate, and a good price. From these considerations, series of parameters related to throughput measurements, fiability, user attention, and services provided, have been defined, as well as the way to obtain this parameters, different types of access, how to interpret the results, and other considerations.

1. Introducción.

Los usuarios de internet en España empiezan a ser conscientes de la importancia que tiene el saber elegir adecuadamente a su proveedor de acceso (ISP) a la red internet. Ya no sólo se fijan en el precio que les supone el acceso, sino que también importa el servicio que obtienen de su ISP en cuanto a la capacidad del mismo en ofrecer una buena conectividad que se vea reflejada en la reducción de tiempos de acceso a los servicios finales del usuario. Igualmente los ISPs observan como sus clientes les exigen un servicio cada vez mejor y para ello necesitan conocer los parámetros de medida más adecuados que les permitan evaluar su servicio comparándolo con el de su competencia. Varias empresas de ingeniería telemática se están especializando en ofrecer servicios de medida de calidad de servicio a los proveedores ISP para ayudarles a detectar fallos en su infraestructura y en definitiva a mejorar su servicio.

El objetivo principal de este artículo es realizar un análisis de los parámetros que sean medibles que se pueden utilizar para establecer la calidad de servicio de un ISP. Al hablar de parámetros medibles nos referimos a aquellos que se pueden cuantificar fácilmente utilizando herramientas sencillas que en muchos casos están a disposición de los usuarios. Este artículo está dirigido a tres tipos de público:

Usuarios individuales:

Generalmente para uso doméstico, utilizan una plataforma sencilla como puede ser un PC a través de una línea de comunicaciones disponible en su hogar que típicamente será RTC o RDSI. Este tipo de usuarios busca la relación calidad-precio en el servicio que recibe del ISP priorizando inicialmente el precio y buscando cada vez más la calidad.

Empresas:

Se puede distinguir entre un uso individual profesional o el uso de grupo en la empresa. Como ejemplo de uso individual puede ser el caso de

comerciales desplazados, soporte técnico distribuido o sistemas de trabajo corporativo desde el hogar. En este caso la plataforma básica puede ser un PC portátil que accede a través de su ISP a los recursos de su empresa. En cuanto al uso como grupo, en general, la propia empresa dispone de una infraestructura suficiente que accede al ISP a través de un router y líneas de acceso contratadas contra el proveedor. En ambos casos el aspecto principal a la hora de seleccionar el proveedor ISP es la calidad de servicio, pues en un entorno de negocios los problemas de acceso a la información suponen pérdidas económicas difíciles de valorar.

Proveedores de servicios de internet (ISPs):

Son los que proporcionan la infraestructura de acceso a internet a los usuarios tanto individuales como empresas, y para ofrecer lo que les demandan sus clientes deben de mejorar constantemente su servicio. Lo que necesitan los ISPs para mejorar su servicio, entre otras cosas, son; herramientas que les facilite la detección de problemas en su infraestructura, establecer mecanismos de medida de calidad que sean adecuados y poder analizar la calidad de su servicio frente a su competencia.

2. Servicios solicitados por los usuarios.

Para poder estimar los parámetros de calidad a tener en cuenta en un ISP, antes debemos analizar cuáles son los servicios que espera un usuario de su proveedor. Inicialmente y si nos olvidamos del precio se pueden distinguir tres tipos de servicios como principales:

- Servicio básico de transporte de la información al mundo internet. Importante para que el usuario disponga de una conectividad con la red de cierta calidad que le permita enviar y recibir datagramas con cualquier punto de internet.
- Servicios complementarios. Albergue de páginas web, e-mail, cuenta de usuario, distribución de utilidades, portal de acceso a Internet, comercio electrónico, etc. Se trata de un servicio complementario en el sentido que un

usuario puede conseguirlos en diferentes puntos de internet al margen que su proveedor se los ofrezca.

- Soporte a usuarios para la resolución de problemas de conectividad de usuarios poco expertos y para mejorar el servicio interesante para los usuarios más expertos.

Según un artículo de investigación [1] reciente en entrevistas realizadas a los gestores de red de las mejores empresas de USA sus demandas de servicio a los proveedores ISP estaban enfocadas a tres puntos fundamentales; un buen servicio de atención al cliente, un buen capacidad en cuanto a ancho de banda (backbone) y un buen precio.

3. Entorno

En la provisión de servicios de acceso a internet son muchos los elementos que intervienen. Es necesario conocerlos y saber manejarlos para garantizar a los usuarios una calidad de servicio adecuada. En este apartado vamos a identificar cuáles son los elementos principales que debemos considerar, qué tipo de problemas pueden surgir en cada uno de ellos y qué soluciones puede aportar nuestro ISP para resolverlos y asegurar así un buen servicio.

3.1 Elementos que intervienen

El usuario:

Se pueden distinguir diferentes tipos de usuario pero en general consideraremos dos clases; los usuarios individuales y los usuarios de grupo. En los individuales podemos encontrar los de uso doméstico o los de uso profesional. En ambos casos utilizan un equipamiento básico sencillo con una sola dirección IP y acceso esporádico por RTC o RDSI. Los usuarios de grupo utilizan un equipamiento más completo, un router, que les permite utilizar un rango de direcciones IP y en general con líneas de acceso permanente.

La línea de transmisión:

Se trata del medio de transmisión que el cliente utiliza para acceder a los servicios del proveedor. Muchas veces la línea que se utiliza restringe el caudal de información que se puede obtener de la red y puede ser un factor a considerar. En el caso de usuarios individuales las opciones más razonables son RTC o RDSI. Los clientes de grupo pueden disponer de diferentes alternativas en función de cual sea la red de acceso que utilicen. Podemos hablar de líneas punto a punto hasta el proveedor ISP y de líneas de acceso a redes de datos como Frame Relay para acceder por ellas al ISP.

La red de acceso de los usuarios:

Los proveedores ISP locales no pueden disponer de infraestructura en todas las provincias del estado para facilitar el acceso de sus clientes a su red con tarifa metropolitana. Por ello surgieron operadores de redes de acceso de usuarios como Infovía con la idea de ofrecer ese valor añadido a los ISPs por un

lado, y facilitar a los usuarios el acceso a servicios de información a un coste reducido e independiente de su ubicación geográfica en el territorio nacional. Estas redes permiten a los ISPs ofrecer un servicio a nivel nacional sin necesidad de instalar equipamiento en cada provincia. Basta con contratar un enlace dedicado con la red de acceso.

Infovía como es ya conocido ha dejado de ofrecer servicio desde primeros de este año. Para mantener el servicio han surgido varias iniciativas como Infovía-Plus (Grupo Telefónica), ReteNet (Retevisión) e InterPistas (Grupo BT).

La infraestructura del proveedor ISP:

Como se ha indicado anteriormente, las redes de acceso de los usuarios han simplificado mucho las necesidades en infraestructura de un proveedor ISP. Sin embargo hay ISPs que mantienen una red de acceso propia con intención de independizar su servicio de los demás y ofrecer mejor servicio. La infraestructura de un proveedor ISP puede contemplar los siguientes elementos:

- Enlaces a redes de acceso para ofrecer conectividad a nivel nacional por ellas.
- Red LAN interna de alta capacidad para ofrecer servicios propios e instalar los equipos de comunicaciones.
- Enlaces de salida a internet para ofrecer capacidad de salida al resto de servidores.
- Red WAN interna entre centros para aprovechar el tráfico interno entre nodos y optimizar el uso de los enlaces de salida.
- Puntos de acceso propios para proporcionar acceso directo a los usuarios a coste reducido.

Los operadores ISP internacionales:

Los proveedores ISP locales contratan enlaces de acceso a internet a través de operadores internacionales que están especializados en dar conectividad a determinadas zonas de internet. Estos enlaces se establecen generalmente a través de redes de datos nacionales y en algunos casos por líneas punto a punto. Hay que saber elegir adecuadamente los operadores internacionales que se precisan para así asegurar una buena conectividad con todas las partes importantes de internet. No todos los ISPs contemplan esta circunstancia.

Los servidores de información:

Se trata de los servicios que se ofrecen en internet y que pueden estar ubicados en cualquier parte de su infraestructura global. Será necesario estudiar cuales son los servidores preferidos de los usuarios para favorecer en lo que se pueda la conectividad a los mismos o establecer la política adecuada de duplicación o réplica de sus servicios.

3.2 Claves para un buen servicio de ISP

En este apartado se pretende indicar de una forma escueta las soluciones que un proveedor ISP puede proporcionar a sus clientes frente a problemas que le puedan surgir en diferentes partes del entorno de su servicio. Se trata de una aproximación muy general

al tipo de problemas típicos que sufren los proveedores y de las soluciones que estos pueden aportar para en definitiva mejorar su servicio. Se presentan a continuación los entornos susceptibles de recibir problemas y se ofrecen las soluciones que se entienden como más adecuadas.

En el equipo de usuario

Los problemas que pueden surgir en el equipo de usuario a la hora de acceder a su proveedor ISP suelen ser en general debidos a una configuración poco adecuada de los equipos. La forma de resolver estos problemas por parte del ISP consiste en disponer de un servicio de atención al cliente adecuado y en proporcionar configuraciones estándar que permitan a sus clientes acceder a sus recursos con un perfil suficientemente genérico.

En la red de acceso

Muchas veces los proveedores ISP dejan en las manos de los operadores de las redes de acceso la resolución de los problemas de acceso de sus usuarios. Es más, incluso pueden usarlos como tapadera para ocultar problemas propios de su servicio. No basta con contratar un enlace con un operador de una red de acceso a través de una red de datos (ej: FR) para dar conectividad a los usuarios. Muchas veces las redes de datos (caso FR) sufren problemas que provocan pérdidas de información que afectan a la calidad de servicio. También son típicos los problemas de falta de capacidad de estos enlaces para cursar el tráfico de demanda de los usuarios. Ante esto las soluciones típicas de los proveedores ISPs suelen ser el dimensionamiento y configuración adecuados de los enlaces de entrada y en algunos casos utilizar líneas de acceso propias. Con la entrada en el mercado este año de varios operadores de redes de acceso se hace necesario la elección adecuada del mismo por parte de los ISPs, y no es descartable que en algunos casos sea aconsejable tener conectividad con varios de ellos a la vez para así garantizar el acceso a los usuarios por vías independientes.

En la infraestructura del ISP

Es en su infraestructura interna donde nuestro ISP puede actuar más fácilmente y resolver los problemas del servicio. Son varias las cuestiones técnicas que un ISP puede tener en cuenta para mejorar y que en muchos casos no se contemplan. Una forma clara de ofrecer mejor servicio es mejorando la conectividad con el mundo internet. Hay dos formas de hacerlo; aumentando la capacidad de los enlaces y/o contratando más líneas de salida. La primera será necesaria cuando se detecte una ocupación excesiva en el enlace, tanto para enlaces de entrada como de salida. La segunda es más recomendable pues no sólo se aumenta la capacidad de salida, sino que también se ofrece redundancia en la red para el caso de problemas en el servicio y tanto más si se eligen operadores internacionales que se complementen.

Otra cuestión a tener en cuenta es la diferenciación interna del tráfico de los usuarios en función de la calidad contratada. Es típico de algunos proveedores ISPs ofrecer servicios diferentes con tarifas diferentes a los usuarios domésticos y a usuarios profesionales, pero sin embargo a la hora de proporcionar el servicio no establecen mecanismos para garantizar la calidad comprometida viéndose así perjudicado el usuario que ha contratado mejor calidad. Un servicio de calidad superior exige que se reserven recursos específicos para garantizar esa calidad, recursos que no pueden ser utilizados en menoscabo de los que los han contratado.

Otra cuestión que algunos ISPs están ya considerando es el tráfico interno entre ellos. Ocurre que muchas veces un usuario de un ISP pretende acceder a servidores que están en otro ISP próximo, sin embargo la comunicación progresa por los enlaces de ambos ISPs con los operadores internacionales ocupando capacidad en unos recursos muy apreciables por ambos. Una forma de mejorar y evitar esas pérdidas de capacidad es estableciendo puntos neutros entre ISPs próximos para rutar el tráfico interno entre ellos.

En cualquier caso, independientemente de la infraestructura del proveedor ISP, uno de los puntos principales para proporcionar un buen servicio es el soporte técnico adecuado de la red propia. Es necesario que el ISP tenga claro el mantenimiento de su red y que no resuelva los problemas por indicación de sus usuarios.

En los puntos negros de internet

Se entiende por "punto negro" en internet una zona de la red con la cual la comunicación de los usuarios es complicada debida a los problemas de conectividad o a la baja calidad de la misma. Cuando un ISP contrata enlaces de salida a internet generalmente utiliza un operador internacional que está especializado en dar conectividad a una zona determinada de internet y que tiene problemas con el tráfico hacia otras zonas. Los operadores internacionales también tienen políticas de establecer puntos neutros entre ellos, pero esto a veces no es suficiente. Luego es importante que un ISP sepa elegir los enlaces internacionales que necesite para asegurar conectividad global incluso contratando el servicio con varios operadores diferentes. Esto incluso es importante desde el punto de vista de redundancia en el servicio pues en caso de fallo se puede disponer de accesos alternativos. Incluso, es frecuente que la calidad de servicio de los operadores internacionales próximos no ofrezcan prestaciones suficientes para el tráfico de los usuarios de un ISP, sobretodo considerando que todos los ISPs utilizan a los mismos operadores y eso no supone ninguna ventaja competitiva importante. Por ello puede resultar importante contratar líneas directas de acceso a operadores internacionales situados en los puntos más importantes de internet y así asegurar un acceso

directo a esas zonas. Este es el caso de enlaces satélite con operadores americanos. Los enlaces satélite tienen la ventaja de proporcionar un ancho de banda bueno con unas prestaciones suficientes que permiten a un ISP en un solo salto evitar los puntos conflictivos de las líneas terrestres internacionales que suelen estar sobrecargadas. El inconveniente principal de estos enlaces suele ser los retardos implícitos que pueden ser críticos para servicios isócronos como la telefonía por IP pero que no son importantes para acceso a servicios de datos.

En los servicios propios

Puede haber problemas en el acceso a determinados servicios propios del ISP debido a diferentes causas pero que en definitiva aparecen por un mal servicio de mantenimiento y soporte técnico que no ha previsto ni solucionado con anterioridad tales problemas. No es poco frecuente la aparición de problemas motivados por errores de configuración de la red. Cuando las demandas de los usuarios crecen pueden colapsar los recursos del proveedor si no ha previsto ese aumento de tráfico. La solución consiste en dimensionar adecuadamente la red. Cuando un ISP alberga un servidor con mucha aceptación ocurre que el servicio se ve degradado por falta de capacidad del equipo en atender tantas peticiones. Para poder mejorar esa respuesta se pueden dar varias soluciones entre las que se encuentran la duplicación del servicio entre varios servidores, la mejora de los servidores en capacidad de proceso, la utilización de servidores caché, etc.

4. Medidas de calidad

En este apartado vamos a identificar los parámetros medibles que se pueden utilizar para evaluar la calidad de servicio de los proveedores ISP tanto en su servicio de red como en el servicio de atención a sus clientes.

4.1 Parámetros medibles en el servicio de red

Se entiende por servicio de red el servicio básico de transporte de datagramas que ofrece conectividad hacia y desde el mundo internet. Como se indicó anteriormente, este servicio es el más importante de los que espera el usuario. Son dos los tipos de medidas que se utilizan para evaluar este servicio:

- Medidas de rendimiento. Miden las prestaciones del acceso en cuanto a capacidad en la conectividad con diferentes puntos de la red.
- Medidas de fiabilidad encaminadas a medir la disponibilidad de la red y de sus servicios.

Medidas de rendimiento

Se suelen establecer sobre dos objetivos principales; los servicios propios incluyendo el equipamiento básico de red (DNS, routers, gateways, etc) y los servicios externos más utilizados por los usuarios (servidores Web, e-mail, ftp).

Son también dos los tipos de medidas que se realizan; el número de datagramas perdidos y el tiempo de respuesta de las peticiones que sirven para evaluar respectivamente las pérdidas de información y los retardos que se producen en las comunicaciones que se realizan. Este tipo de medidas son fáciles de realizar incluso por parte de los propios usuarios. Existe una utilidad conocida como “ping” que permite generar un tráfico continuo hacia un servidor cualquiera de la red desde el terminal de usuario y que está diseñado para ser respondido en eco desde el receptor y así poder analizar los parámetros mencionados. Esta utilidad sirve a los usuarios para poder estimar si los servicios propios de su ISP funcionan en condiciones y la capacidad de su ISP en ofrecer conectividad a servicios externos. Para que los resultados sean contrastados es necesario establecer un método de medida adecuado.

Medidas de fiabilidad

Estas medidas van encaminadas a evaluar los siguientes aspectos referidos al servicio de la propia red y a la de los servidores propios:

- **Accesibilidad al servicio, ¿se puede acceder?.** Se entiende por accesible si es viable establecer alguna conexión con el mismo.
- **Disponibilidad del servicio, ¿está el servicio disponible?.** Se entiende por disponible si supera un umbral de conexiones exitosas.
- **Duración de la indisponibilidad, ¿cuánto va a durar?.** Importante para evaluar la capacidad técnica del ISP en la resolución de problemas.
- **Tiempo entre indisponibilidades, ¿cada cuánto ocurre?.** Permite evaluar la solidez del servicio del ISP a la hora de mantener su red.

4.2 Parámetros medibles en el servicio de atención al cliente

Los usuarios pueden jugar un papel importante en el mantenimiento de los servicios de red y en la detección de problemas. El servicio de atención a clientes bien utilizado por los proveedores ISPs puede ser una herramienta clave en la medida de la calidad de su servicio. Se entiende por bien utilizado cuando el ISP establece mecanismos de intercambio de información entre departamentos de las demandas de los usuarios, cuando se hace un análisis adecuado de la información recibida y cuando se valora adecuadamente la información de los clientes. Muchas veces los proveedores omiten las demandas de los usuarios lo que les puede conducir a la pérdida de confianza mutua. El usuario es el más indicado para controlar el funcionamiento correcto del proveedor, entre otras cosas porque está conectado permanentemente. “*El usuario vigila*”.

Los parámetros de medida de la atención al cliente que se pueden establecer son:

- **Disponibilidad del servicio.** Medida en general en horas de presencia (8, 16, 24 horas)
- **Volumen de consultas clasificadas por temas.** Facilita la detección de problemas.

- Tiempo medio de respuesta frente a cada tipo de problema. Mide la duración de cada problema.
- Nivel de satisfacción del usuario (% consultas resueltas). Permite evaluar la calidad del servicio de atención al cliente en todas sus facetas (técnica, atención comercial, etc).
- Frecuencia de aparición de cada tipo de problema. Mide la capacidad técnica de resolución de los problemas por parte del proveedor ISP. Mide la rigidez de la red.

Estos parámetros se pueden controlar a través del servicio de atención al cliente si el proveedor ISP se ha preocupado de establecer los mecanismos adecuados de captura de la información bien sea a través de formularios que rellena el personal de atención o por métodos más automatizados.

4.3 Selección de destinos

Un aspecto muy importante a tener en cuenta en la realización de las pruebas sobretodo en medidas de rendimiento es la selección adecuada de los destinos sobre los que se realizan las medidas. Resulta fundamental respetar algunas normas de objetividad que eviten obtener resultados desvirtuados. Por ello, la selección de los destinos se ha de realizar en base a unas premisas previas de objetividad e independencia que evite en lo posible situaciones de privilegio de unos proveedores frente a otros. Existen en internet listas de los servicios más utilizados por los usuarios que en principio parecen las más indicadas para realizar las medidas. Hay que comprobar la situación de esos servidores para evitar primar en las medidas a los proveedores que los albergan frente al resto de competidores.

Otra cuestión a considerar es la homogeneidad de las pruebas de tal forma que se proporcionen unos datos suficientes sobre el ámbito de actuación de los proveedores en toda su extensión. Algunos análisis se restringen a medir parámetros específicos que se conoce a priori que favorecen a unos frente a otros. Es importante a la hora de analizar una información sobre medidas conocer el método de pruebas utilizado para poder evaluarlas correctamente.

4.4 ¿Quién debe medir?

Las medidas se pueden realizar por diferentes agentes e incluso conviene que cada uno establezca sus medidas de cara a poder contrastar los resultados. Sin embargo, en función de cuales sean los intereses de la entidad que realice las medidas los resultados pueden ser diferentes e incluso pueden estar cuestionados. Independientemente de que el método sea bueno o malo, los resultados pueden ser más o menos creíbles en función de quién realice las medidas. Por supuesto que los propios proveedores ISPs pueden y deben realizar sus propias medidas, sin embargo la utilización de las mismas estará restringida a su ámbito de actuación. Incluso son difícilmente aceptables por otros departamentos. Los problemas más típicos asociados a las mediciones de calidad internas son:

- Proporcionan resultados diferentes. Se suelen basar en métodos de medida parciales y propios que son diferentes de los establecidos por otros ISPs y por lo tanto difícilmente comparables.
- No pueden publicar sus datos con credibilidad suficiente. Al ser parte activa en los resultados su credibilidad es limitada.
- No pueden monitorizarse entre ellos para comparar resultados. Al tratarse de competencia los ISPs son reacios a intercambiar los datos de sus medidas para evaluarse comparativamente.
- Internamente se pueden ocultar problemas. Es difícil que unos departamentos asuman responsabilidades frente a otros cuando los resultados son comprometedores.

Luego, como norma general,

“No debe de realizar medidas de calidad nadie que pueda estar comprometido con el servicio”.

Esto incluye a proveedores de equipamiento y a empresas contratadas para cubrir el servicio de red ISP. Se hace necesaria la aparición de entidades diferentes que ofrezcan:

- Garantía suficiente en cuanto a independencia y objetividad.
- Capacidad tecnológica para realizar el trabajo.
- Mediciones fiables, exactas, uniformes e imparciales.
- Posibilidad de contrastar resultados.

Varias empresas internacionales como Matrix [2] y WorldScan [3] se han dado cuenta de esto y están ofreciendo sus servicios de monitorización a los proveedores ISPs principales en internet. Nuestro grupo GIT aprovechando su imagen de independencia pretende proporcionar este tipo de servicios en el País Vasco para ISPs.

5. Nuestra plataforma y método

La plataforma de pruebas ha de ser lo más parecida a la típica de un usuario. En cualquier otro caso se podrían desvirtuar los resultados frente a los obtenidos por los usuarios. En nuestro caso, se han definido dos plataformas de pruebas; una para accesos por RTC y otra a través de RDSI. En ambos casos se utiliza un equipamiento básico de usuario que consiste en un ordenador personal Pentium II a 300 MHz y con sistema operativo Windows 98. Los accesos a los proveedores se realizan vía módem (56Kbps) por RTC y con un adaptador de terminal por un acceso básico (2B+D) directo de RDSI.

En cuanto al método utilizado de pruebas se basa en tres pilares fundamentales:

- Objetividad. Se evitan las decisiones que puedan ser parciales (Ej: los destinos se han elegido de fuentes externas independientes AIMC –Asociación para la Investigación de los Medios de Comunicación-)
- Intensidad de las pruebas. Se establece un número de pruebas estadísticamente suficiente

para sacar conclusiones comparativas en diferentes franjas horarias y con diferentes destinos.

- **Fiabilidad.** Se hacen pruebas automatizadas utilizando programas comerciales que están específicamente diseñados para este tipo de análisis.

En fases posteriores se pretende diseñar programas propios para realizar análisis de parámetros internos en la propia infraestructura de los proveedores ISPs de forma continua. Se utilizarán los resultados de análisis previos para validar los productos desarrollados.

5.1 Parámetros medibles

Se describen a continuación las medidas típicas que se realizan para los proveedores ISPs. Estas medidas corresponden a tres grupos diferenciados:

Medidas de rendimiento internas

Se realizan sobre diferentes puntos importantes de la infraestructura interna del proveedor y en ellas se estiman los retardos medios y las pérdidas de datagramas que se producen. Los puntos típicos susceptibles de análisis son; el DNS primario, el router principal y pruebas cruzadas entre ISPs

Cabe destacar las pruebas cruzadas realizadas entre ISPs en el sentido que las medidas de rendimiento están enfocadas a evaluar la necesidad de establecer puntos neutros entre ISPs para mejorar su conectividad y evitar las pérdidas de rendimiento en los enlaces de salida hacia internet.

Medidas de fiabilidad interna

Estas medidas están orientadas a medir tanto la disponibilidad de la red de acceso del proveedor como la de sus servicios internos principales (servidores web y servidores de e-mail, ftp, etc).

Medidas de rendimientos por zonas

Se trata de medidas de rendimiento sobre los servicios externos situados en las zonas más utilizados por los usuarios potenciales del proveedor. Como se indicó anteriormente para poder obtener parámetros comparables con otros ISPs se eligen adecuadamente los destinos finales sobre los que se realizan las pruebas basándose en fuentes externas independientes y evitando la toma de decisiones parcial que pueda primar a unos ISPs frente a otros. Las medidas que se realizan tienen por objetivo básicamente los siguientes servidores:

- **Servidores Web.** Se miden los retardos y la velocidad de descarga tanto del contenido de texto, como el de imágenes y el de ambos.
- **Servidores FTP.** Se miden los retardos y la velocidad de descarga de unos ficheros tipo.

5.2 Tipo de pruebas y herramientas

Las pruebas típicas realizadas para evaluar la calidad de servicio de los proveedores ISP en nuestros análisis corresponden a tres tipos de acceso muy

comunes entre los usuarios de carácter individual; accesos por RTC a su ISP a través de una red de acceso (Ej: Infovía Plus), accesos por RDSI usando también la red de accesos, y por último accesos directos por RDSI. Los datos recogidos permiten obtener información suficiente para poder analizar la calidad de servicio que los proveedores proporcionan para cada tipo de servicio estudiado.

En cuanto a las herramientas utilizadas para realizar las medidas indicar que se trata de una plataforma totalmente estándar y con aplicaciones comerciales específicamente diseñadas para la realización de las pruebas hasta el punto que están recomendadas por varios analistas [4] [5] del mercado.

5.3 Resultados de las pruebas realizadas

Las pruebas han sido realizadas en la segunda quincena del mes de Enero de 1999 en el laboratorio de investigación del área de Telemática en la Escuela de Ingenieros de Telecomunicaciones de Bilbao sobre varios proveedores ISPs de carácter nacional. Los resultados de tales pruebas por motivos de confidencialidad con los clientes no pueden ser publicados tal cual se obtuvieron, pero si que se pueden proporcionar algunos valores de referencia global que nos permiten analizar la calidad de servicio de nuestros proveedores. En concreto, se dan los datos medios correspondientes a las pruebas realizadas sobre ocho proveedores ISP. Para poder estimar la dispersión en los resultados entre ISPs se presenta el resultado más alto, el resultado medio y el más bajo para cada parámetro sin indicar a quién corresponde cada valor.

También se presentan datos separados por franjas horarias diferentes entendiendo dos horarios; el de empresas (de 8:00 a 20:00) y el de usuarios domésticos (de 20:00 a 8:00). Donde se precisa se separan las pruebas realizadas por RTC de las realizadas por RDSI de tal forma que se puede evaluar el impacto que tiene la línea de acceso en los rendimientos que consigue el usuario.

6. Interpretación de los resultados

Medidas de rendimiento por zonas

Se han establecido la medida de tres parámetros principales; retardo medio, porcentaje de pérdidas y el retardo ponderado. El retardo medio nos da una idea de los tiempos de latencia de la información en la red en función del número de enlaces entre origen y destino y de la clase de enlaces. Realmente no es un parámetro indicador de la calidad de servicio, por eso se habla del retardo medio ponderado que intenta evaluar la calidad de la conectividad con diferentes puntos de internet en función del retardo medio y la tasa de pérdidas de datagramas.

En las gráficas proporcionadas cabe destacar la diferencia en los retardos medios y en los ponderados que existe entre los accesos por RTC y los realizados por RDSI. Se observa un mejor comportamiento para RDSI.

Igualmente se observa un mejor comportamiento en cuanto a retardos en el horario de los usuarios y en las conexiones nacionales frente a las internacionales. También se observa una diferencia notable en las estadísticas de pérdidas de paquetes y de retardos ponderados entre unos ISPs y otros. Estas diferencias en muchos casos resultan escandalosas, de aquí la importancia al seleccionar el proveedor adecuado.

Acceso a los servidores Web más utilizados

En cuanto a las pruebas realizadas sobre transferencia de páginas con servidores Web en diferentes zonas de internet cabe destacar algunas consideraciones. Se puede comprobar fácilmente las ventajas que tiene un usuario al acceder por RDSI. Se consiguen prestaciones que en muchos casos duplican los rendimientos sobre RTC. En el horario de usuarios se consiguen ciertas mejoras pero no tantas como se podría esperar.

Si nos fijamos en los valores medios de transferencia de datos podemos observar algunos números mágicos que nos pueden servir como referencia. Por ejemplo para RTC se consiguen velocidades medias de transferencia de datos con USA, Europa y España de 8, 9 y 10 Kbps respectivamente. Para acceso por RDSI hablamos de 12, 13 y 20 Kbps.

Medidas de disponibilidad del servicio de red

En el último gráfico se presentan las estadísticas correspondientes al estudio de la disponibilidad de servicio de red de un proveedor ISP. El aspecto más claro a observar es el porcentaje de pérdida de llamadas que algunos proveedores ISP analizados están proporcionando. En algunos casos las gráficas muestran que hay proveedores que llegan a perder más del 40% de las llamadas recibidas, mientras que otros aceptan prácticamente el 100%. Esto muestra claramente que la preocupación por el servicio de red no es la misma entre unos proveedores y otros.

7. Limitaciones de las pruebas

Para una correcta evaluación de los resultados se debe tener en cuenta una serie de limitaciones que afectan a la exactitud de los resultados, entre las cuales cabe mencionar las siguientes:

- Las pruebas normalmente se realizan desde una ubicación geográfica determinada lo que puede influir en cierta medida en los resultados.
- Algunos equipos tienen inhabilitada la respuesta a los pings por lo que no se permite realizar pruebas de este tipo. Otros, por el contrario, aumentan la prioridad de estos servicios distorsionando así las medidas.
- Se pueden producir situaciones transitorias poco satisfactorias (caídas de algún nodo, problemas con alguna línea, etc..) debidas a problemas puntuales que en modo alguno pueden ser cuantificados en unas pruebas cuya duración sea de unos pocos días. Para obtener en este caso unos resultados más precisos se aconseja la

realización de pruebas con una periodicidad adecuada.

8. Conclusiones

Desde el punto de vista de los usuarios:

No todo es el precio, cada vez es más importante la calidad de servicio de nuestro ISP, sobre todo para usuarios profesionales. Hay formas de medir esa calidad de servicio. Incluso el propio usuario puede hacerlo para extraer sus propias conclusiones. Hay que exigir a nuestro proveedor ciertas garantías en cuanto a la calidad de servicio que ofrece. Los usuarios deben de tener cuidado con los análisis de calidad publicados. Hay tendencia en el mercado por publicar información referente a la calidad de servicio de los ISPs con cierta parcialidad. Hay que analizar la fuente, estudiar el método de pruebas y ver la importancia de los datos.

Desde el punto de vista del ISP:

Cada vez más la QoS es un factor fundamental diferenciador tenido en cuenta por los usuarios a la hora de elegir a un ISP. Conviene establecer mecanismos de medida de la calidad de servicio al usuario que sean objetivos e independientes tanto en el servicio básico de red como en la atención al usuario. El ISP debe de contrastar su QoS: Es importante tener una referencia exacta del tipo de servicio que se está ofreciendo comparativamente con la competencia. Se deben utilizar criterios comunes de análisis que sean objetivos e independientes.

Las medidas de QoS pueden ayudar a la detección de problemas en el servicio de red. Para mantener los parámetros ideales de calidad es necesario introducirse en un proceso de mejora continua. De esta forma se darán soluciones adecuadas a los problemas que vayan surgiendo.

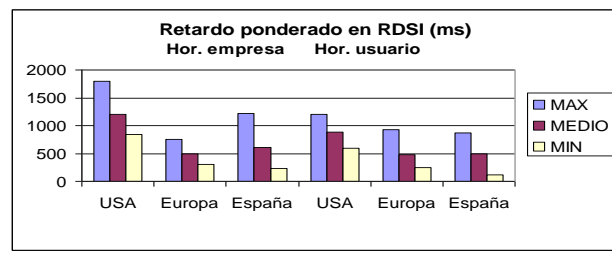
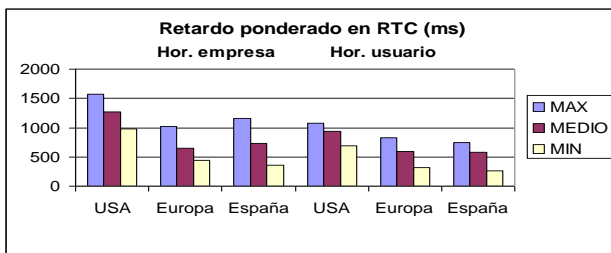
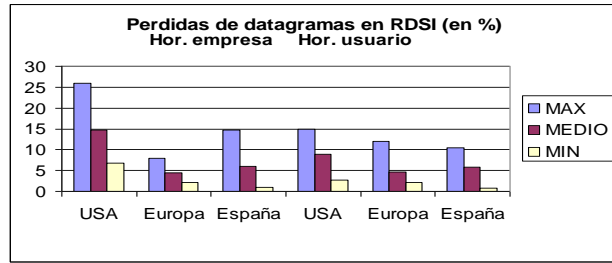
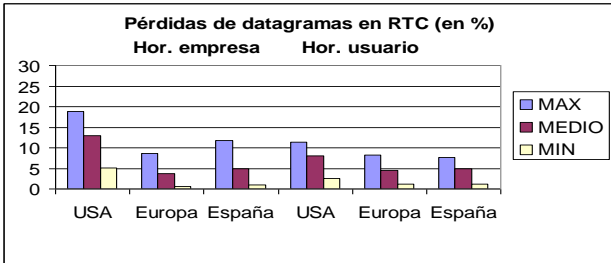
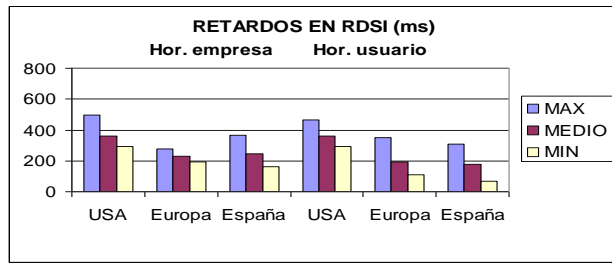
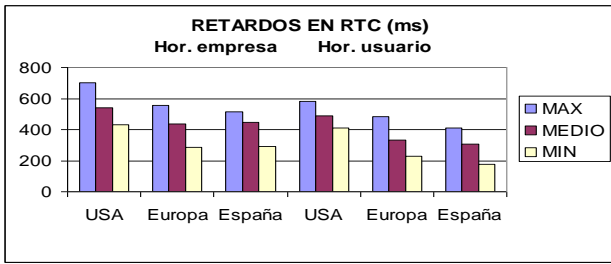


Fig.1 Medidas de rendimiento por zonas

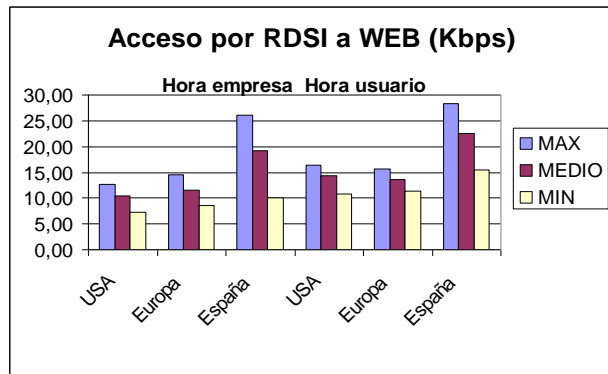
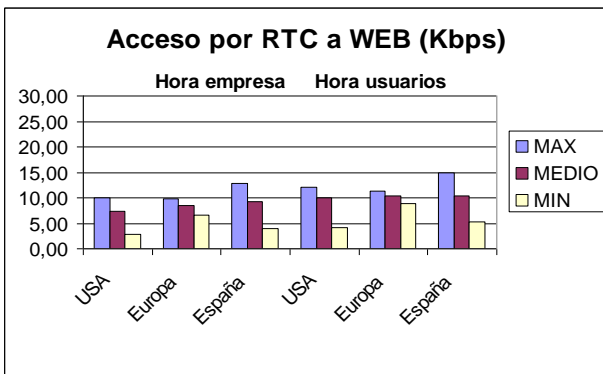


Fig.2 Medidas de rendimiento en el acceso a los servidores Web más utilizados

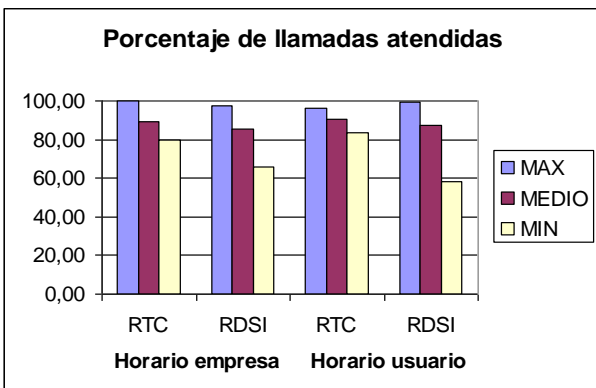


Fig.3 Medidas de fiabilidad en el servicio de red

Referencias

- [1] Data communications International. "Corporate Class ISPs". ISP Investigation.
- [2] Matrix Information and Directory Services, Inc. (MIDS). "The Internet's most experienced independent analysts". <http://www.mids.org/>
- [3] Worldscan Internet Intelligence Services, Ltd.. "Worldscan welcomes you". <http://www.worldscan.com>.
- [4] CAIDA Measurement Tool Taxonomy. "Internet measurements tool Survey". <http://www.caida.org/Tools/taxonomy.html>
- [5] El Sacapuntas. "Pruebas de velocidad a proveedores". <http://www.areas.net/dp/home.htm>.

ATM Sobre Ethernet Mediante el Protocolo CIF

J. M. Arco, B. Alarcos, A. M. Hellín, A. García, D. Meziat
Universidad de Alcalá
Escuela Politécnica
28871 Alcalá de Henares
Teléfono: +34 91 8856627-28
{jmarco, bernardo, hellin, antonio, meziat}@aut.alcala.es

Abstract

This paper describes the research carried out to obtain the advantages of ATM technology over legacy Ethernet networks. A protocol called "Cells In Frame" (CIF) allows to encapsulate ATM cells in Ethernet frames. We have implemented CIF protocol and several queuing algorithms in the Linux operating system kernel. Measures of bandwidth allocation, delay and throughput have been carried out in order to check that the developed system fulfils the requirements of multimedia traffic. Obtained results show that it provides the necessary ATM quality of service, paying a minimum throughput decrease respect to native Ethernet. Also a queuing algorithm to optimize traffic management is being researched.

1 Introducción

El objetivo global de nuestro trabajo consiste en aportar las ventajas de la calidad de servicio de ATM a redes clásicas como Ethernet. Se pretende conseguir de esta forma llevar las ventajas de esta tecnología hasta el usuario sin necesidad de realizar las inversiones para migrar a ATM [1].

El escenario en el que se pretende aplicar este desarrollo consiste en redes LAN interconectadas mediante WAN, utilizado en la mayoría de los entornos de trabajo. La tendencia actual en las redes WAN es la utilización de la tecnología ATM [2][3]. En cuanto a las redes de área local la tecnología dominante en la actualidad es Ethernet, no siendo probable que ATM se imponga en este entorno a medio plazo [4] debido a que, entre otros, presenta el inconveniente de requerir mayores inversiones, tanto en el conmutador como en el adaptador de red, además de ser una tecnología más compleja. Sin embargo, Ethernet presenta el inconveniente de no ser adecuada para la transmisión de vídeo y audio, ya que no ofrece garantía de calidad de servicio. Para superar esta dificultad se puede introducir algunas variaciones sobre la red Ethernet y dotarla de las prestaciones de ATM, sin perder las propias.

Una posibilidad para ello consiste en la utilización de un nuevo protocolo que integre la red local dentro de ATM, este protocolo es el denominado "Cells In Frame" (CIF) [5]. La implantación de CIF requiere dos actuaciones, añadir software en los sistemas finales, y el desarrollo de un conmutador especial que permita conectar ambas tecnologías. La topología genérica se ilustra en la figura 1.

En el sistema final, CIF envía dentro de una trama Ethernet una cabecera específica y la carga útil de varias celdas ATM. La función del conmutador CIF consiste en extraer la carga de las celdas ATM de las tramas Ethernet, reconstruir su cabecera y enviarlas a la red ATM. Con CIF también conseguimos que la

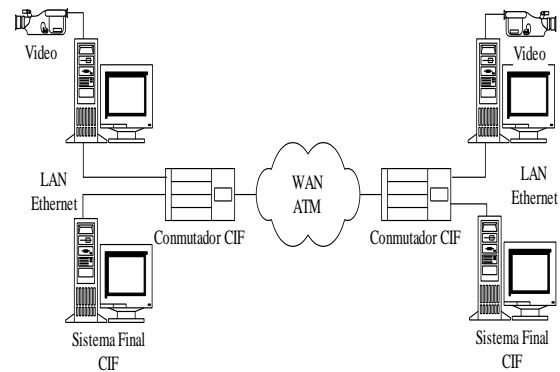


Fig. 1. Topología de red.

señalización ATM llegue al usuario, manteniendo la tecnología Ethernet.

2 Arquitectura de Protocolos CIF

La arquitectura de protocolos a desarrollar, representada en la figura 2, debe cumplir la especificación CIF 1.0 [6] del Forum CIF.

La torre de protocolos ATM-CIF es similar a la ATM [7], con la diferencia de que el nivel físico es Ethernet. A continuación se describen brevemente las particularidades de cada nivel de CIF respecto a ATM.

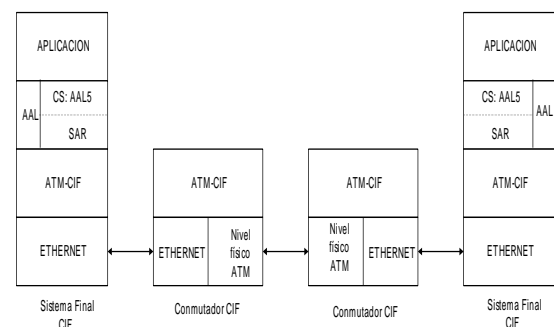


Fig. 2. Torre de protocolos.

La capa de adaptación a ATM, AAL (ATM Adaptation Layer), proporciona una interfaz estándar a los niveles superiores, que pueden ser directamente aplicaciones ATM nativas, u otra pila de protocolos como TCP/IP.

En cuanto a la subcapa de convergencia CS, con CIF se puede utilizar cualquier AAL. Normalmente se utiliza AAL5 por ser la más eficiente.

La subcapa de segmentación y reensamblado (SAR), se encarga de trocear la información en los bloques de 48 bytes que forman la celda. En CIF, como se explica en el siguiente párrafo, no se realiza esta función salvo que la longitud de los datos de usuario sea mayor de 1480 bytes, límite para no superar la capacidad de la trama Ethernet.

La capa ATM-CIF se diferencia de una capa ATM en que no tiene que generar una cabecera por cada celda, si no una cabecera patrón para un grupo de celdas pertenecientes a la misma conexión, que se encapsulan en una misma trama Ethernet. Esto se hace por motivos de eficiencia.

El conmutador CIF debe realizar varias funciones:

- Control de los Parámetros de Uso (UPC), que simula la función policía de la red ATM, para comprobar que el tráfico recibido está de acuerdo con la velocidad pactada. Para implementar esta función se utiliza el algoritmo del cubo con goteo (leaky bucket).
- Extracción de las celdas de la trama Ethernet y su inserción en la red ATM.
- Gestión de las colas de salida para garantizar la calidad de servicio.
- Mantenimiento del enlace con los sistemas finales, ya que puede haber varios por segmento.

La figura 3 ilustra el flujo de datos por los distintos niveles, suponiendo que la longitud máxima del mensaje es 1480 bytes.

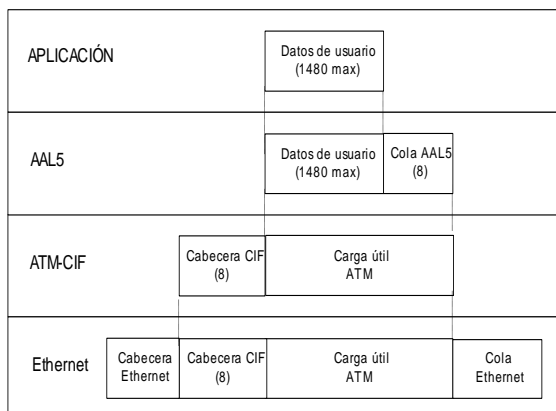


Fig. 3. Flujo de datos

La información de la cabecera CIF, determina cómo se van a realizar un serie de procedimientos, tales como el control de errores, la señalización ATM y el mantenimiento del enlace CIF. También indica si los tiene que realizar el conmutador o el sistema final CIF.

3 Implementación realizada de CIF

3.1 Escenario

Se está desarrollando la implementación de un sistema final y un conmutador CIF, de acuerdo con el esquema de la figura 1. El desarrollo se ha realizado sobre el sistema operativo Linux, de dominio público, muy difundido y que además dispone de fuentes, depuradores y documentación para desarrollo.

La primera fase de la implementación se ha realizado de acuerdo con el esquema de la figura 4, que es una simplificación de la red de la figura 1. La simplificación consiste en unir dos PCs, (que actúan de sistemas finales) mediante un único conmutador CIF. Hay que indicar que este no es el escenario final pero se puede considerar válido para garantizar la QoS.

El enlace entre el PC y el conmutador CIF es Ethernet half-duplex, esto puede producir problemas de compartición del medio si la carga es elevada. Para evitarlo habría que implementar un control de admisión de conexiones, o utilizar un Ethernet full-duplex.

Además es un escenario real que se da cuando se quiere transmitir con calidad de servicio en conexiones LAN-LAN.

3.2 Descripción de la implementación

En el sistema final se ha desarrollado una torre de protocolos CIF dentro del núcleo, para que pueda ser usada eficientemente por varias aplicaciones [8]. El conmutador CIF consiste en un PC y sus correspondientes tarjetas de red, en el que se ha introducido una torre CIF específica.

Se han implementado los aspectos básicos de la norma CIF, no incluyendo en esta primera fase algunos aspectos, como por ejemplo el tratamiento

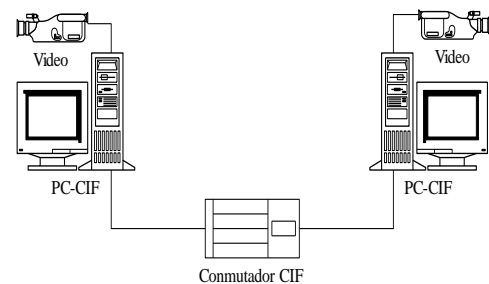


Fig. 4. Escenario considerado.

de errores de la cabecera ATM (HEC) ya que en la práctica son muy poco frecuentes, y en caso de producirse su detección se realiza a nivel Ethernet. Esta primera fase soporta tráfico CBR y UBR.

Para facilitar las tareas de desarrollo y depuración se ha realizado la implementación de la torre de protocolos CIF, en un módulo que permite compilarlo independientemente y cargarlo dinámicamente en el núcleo [9].

3.2.1 Arquitectura de comunicaciones

Para realizar la implementación en el núcleo se ha aprovechado la arquitectura de comunicaciones utilizada en Linux, añadiendo a la familia de protocolos estándar [10], uno nuevo (que denominamos *AF_CIF*) y una estructura de direcciones específica (*sockaddr_cif*).

Cuando un proceso se comunica a través de la red [11] utiliza las funciones de la capa *socket* BSD (figura 5). Esta capa realiza tareas de administración de *sockets*, utilizando para ello unas estructuras generales de datos (*socket* y *msg_hdr*). El nivel *socket* BSD simplifica la portabilidad de las aplicaciones de red.

Debajo de esta capa se encuentra otra denominada *socket* INET, que hace de interfaz entre el nivel *socket* BSD y los protocolos específicos, en nuestro caso CIF. Esta capa utiliza la estructura *sock*.

Entre la capa CIF y los dispositivos de red se encuentra un interfaz común (*dev.c*) y un driver específico para cada dispositivo.

En la recepción el código contenido en *datagram.c*, se encarga de entregar el mensaje al protocolo correspondiente.

En la figura 6 se muestra la secuencia de los módulos que intervienen en el proceso de comunicaciones.

Las funciones implementadas en el conmutador CIF son el establecimiento del enlace la retransmisión de tramas y la gestión del tráfico.

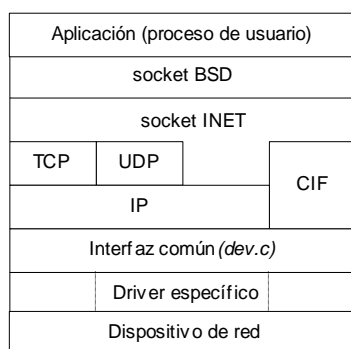


Fig. 5. Torre implementada.

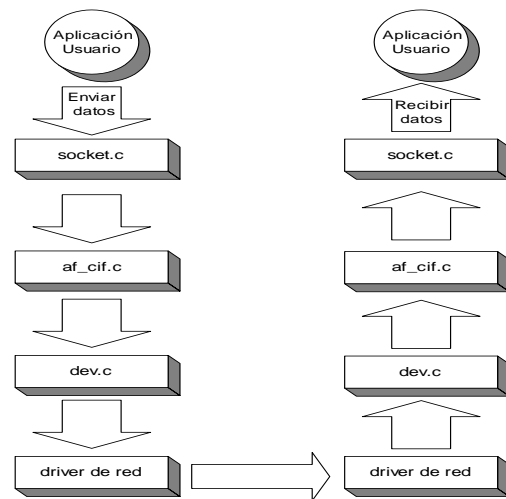


Fig. 6. Flujo de programas.

3.2.2 Algoritmos de encolado

Se ha realizado un análisis y programación de diferentes algoritmos de la familia WFQ (Weight Fair Queuing) [12][13], para ver cual se adapta mejor a nuestro sistema. Comenzamos con un algoritmo sencillo, SCFQ (Self-Clocked Fair Queuing)[14] y continuamos con FFQ (Frame-Base Fair Queuing) y SPFQ (Starting Potential-Based Fair Queuing) [15].

En los algoritmos WFQ cuando llega una trama se calcula y asocia una etiqueta (Time Stamp, *TS*) que va a determinar el orden de salida y se envía a la cola de su conexión. La siguiente trama a transmitir será la que tenga el valor de *TS* más pequeño. *TS* se calcula de acuerdo a la fórmula:

$$TS_i^k = \max(v(t), TS_i^{k-1}) + \frac{L_i^k}{\phi_i r}$$

donde:

$v(t)$ es la función de tiempo virtual que representa el reparto de ancho de banda realizado por el sistema.

TS_i^{k-1} corresponde a la trama anterior de esa sesión.

L_i^k es el tamaño de la trama en bits.

ϕ_i es el peso normalizado de la sesión.

r es la velocidad de transmisión.

Cuando una sesión tenga tráfico pendiente se cumple que $v(t) \leq TS_i^{k-1}$ por lo que $v(t)$ no influye en *TS*. En caso contrario, al recibir la primera trama después de un periodo sin tráfico, se tiene en cuenta $v(t)$ para actualizar la sesión. La dificultad del planificador WFQ radica en calcular $v(t)$, por lo que se han propuesto otros algoritmos que simplifican su cálculo.

El algoritmo SCFQ aproxima $v(t)$ utilizando el valor de la etiqueta de la última trama transmitida.

Los algoritmos FFQ y SPFQ son del tipo RPS (Rate Proportional Server). Estos algoritmos se basan en el cálculo y calibración de la función potencial, cuyo incremento para una sesión durante el periodo de tiempo $[\tau, t)$, se define como:

$$P_i(t) - P_i(\tau) = \frac{W_i(\tau, t)}{\rho_i}$$

donde:

$W_i(\tau, t)$ es el servicio recibido por la sesión i durante el periodo $[\tau, t)$.

ρ_i es la velocidad mínima de la sesión i .

Se define la función potencial del sistema como:

$$P(t) = F(P_1(t-), P_2(t-), \dots, P_N(t-), t)$$

es decir, depende de las funciones potenciales de cada sesión y del tiempo. En los algoritmos RPS, $P(t)$ realiza una función similar a $v(t)$.

$P(t)$ tiene pendiente 1 y debe calibrarse periódicamente para reflejar el comportamiento real del sistema. En FFQ la calibración se realiza después de transmitir F bits y en el SPFQ cada vez que se transmite una trama.

3.2.3 Implementación en el conmutador

En la figura 7 puede verse el esquema básico desarrollado. En el funcionamiento estándar de Linux, todos los mensajes seguirían el mismo tratamiento a través de una única cola. De este modo, si enviásemos un mensaje CIF, pasaría a través de las funciones $dev_queue_xmit()$ y $do_dev_queue_xmit()$, que encolan la trama y a través de la función $dev->hard_start_xmit()$ que llama al driver específico del dispositivo de red.

Para obtener QoS se ha modificado el esquema anterior (contenido en el fichero $dev.c$) añadiendo tantas colas como conexiones haya, e implementando el algoritmo de encolado para gestionarlas.

Conviene destacar, que se han modificado campos de la estructura $device$, que es la que aporta las funciones necesarias para la comunicación con los drivers de red [16]. La modificación fundamental ha ido orientada al incremento del número de colas mediante el campo $DEV_NUMBERS$. El array de estructuras sk_buff_head $buffs$ $[DEV_NUMBERS]$ permite acceder a las mismas.

Cada trama está contenida en una estructura sk_buff . Una cola está formada por estructuras sk_buff doblemente enlazadas, que son gestionadas por la estructura sk_buff_head . Para manejar las colas se

utilizan las funciones $skb_dequeue()$ y $skb_queue_tail()$ para encolado y desencolado respectivamente [17].

El proceso de entrada del algoritmo clasifica la trama y calcula la etiqueta de salida. El de salida selecciona la trama con etiqueta de salida más baja, para su transmisión.

Inicialmente se ha realizado la señalización para el establecimiento de la conexión y la especificación de sus parámetros, de forma manual. Hay una serie de circuitos permanentes reservados para cada aplicación. En el conmutador CIF se pueden modificar los parámetros de cada conexión.

4 Pruebas

Las pruebas se han realizado sobre dos escenarios distintos como el de la figura 4, uno con PCs a 133 MHz y otro a 266 MHz, para comprobar la evolución de las características. Las pruebas permiten comparar el rendimiento de los algoritmos implementados (SCFQ, SPFQ, FFQ).

Se han realizado medidas de:

- Reparto de ancho de banda.
- Retardo.
- Throughput [18].

Todas las pruebas se han realizado con un tamaño de datos de 1480 bytes y con las sesiones transmitiendo a la máxima velocidad.

Para la prueba de reparto de ancho de banda, hay que tener en cuenta que los planificadores de tráfico deben garantizar a las sesiones el ancho de banda asignado. La capacidad sobrante, ya sea por ausencia de tráfico o por no estar asignado, se reparte en proporción a sus reservas. Las sesiones sin QoS sólo recibirán servicio cuando no haya tráfico de sesiones con QoS.

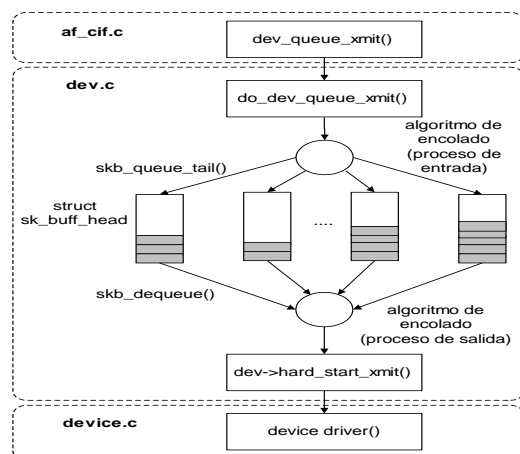


Fig. 7. Gestión de tráfico.

Para medir el retardo que introduce el algoritmo, se mide el RTT (Round Trip Time) entre sistemas finales.

Para la medida del reparto de ancho de banda y throughput se ha empleado dos PCs (uno de ellos el usado como conmutador) como transmisores con un proceso de dos sesiones y un PC como receptor con un proceso de cuatro sesiones.

La pérdida de throughput debida al planificador se ha medido repitiendo la prueba y aumentando sucesivamente el número de tramas transmitidas, hasta que se empiezan a perder.

4.1 Resultados

4.1.1 Asignación de ancho de banda.

La tabla 1 indica el peso de las cuatro sesiones. Los resultados se han obtenido tomando medidas de tiempo en cada trama transmitida a la salida del interfaz común (*dev.c*).

No se observan diferencias en los resultados entre el escenario con PCs a 133 y a 266 MHz.

4.1.2 Retardo

Se ha medido el RTT durante un periodo de 20 segundos, el cual es suficientemente grande para evitar el efecto de la ejecución de algún demonio. Con el tamaño de 1480 bytes se mide el retardo máximo [19]. En las tablas 2 y 3 se observan los retardos para PCs a 133 y 266 MHz.

Tabla 1: Asignación de ancho de banda.

BW reservado	BW teórico	FFQ	SPFQ	SCFQ
0,5	0,625	0,628	0,626	0,615
0,2	0,250	0,248	0,250	0,262
0,1	0,125	0,124	0,123	0,123
0	0	0	0	0

Tabla 2. Retardo en PC-133

	RTT (μsec.)	Diferencia (μsec.)	Diferencia en %
FIFO	8523	0	
SCFQ	8536	13	0,15
SPFQ	8561	38	0,44
FFQ	8558	35	0,41

Tabla 3. Retardo en PC-266

	RTT(μsec.)	Diferencia (μsec.)	Diferencia en %
FIFO	5163		
SCFQ	5169	6	0,12
SPFQ	5173	10	0,19
FFQ	5171	8	0,15

4.1.3 Throughput

Las medidas tomadas se muestran en las tablas 4 y 5. Dado que el throughput depende de numerosos factores (número de sesiones, número de procesos, pesos de las sesiones, modelo de tráfico, tamaño de los mensajes, tamaño de las colas, sentido del tráfico, etc), estos resultados deben interpretarse como una simplificación válida para la finalidad perseguida.

4.2 Análisis de resultados

No se aprecian grandes diferencias de rendimiento entre los distintos algoritmos, observándose que a mayor velocidad del procesador estas se reduce.

El FFQ tiene mejores resultados que el SPFQ ya que en el segundo las actualizaciones de la función potencial son más frecuentes. La ventaja del SPFQ es que transmitirá tramas de manera más justa que el FFQ.

Los resultados de la distribución del ancho de banda se corresponden con los calculados teóricamente.

Los valores de retardo son análogos a los medidos en otro escenario similar [20]. En cuanto al throughput las diferencias medidas son menores que en dicho escenario, lo cual puede ser debido a la utilización de algoritmos diferentes.

Tabla 4. Throughput en PC-133

	Throughput (Mbps.)	Diferencia (Kbps.)	Diferencia en %
FIFO	9,7248		
SCFQ	9,707	8	0,08
SPFQ	9,70	24	0,25
FFQ	9,680	44	0,45

Tabla 5. Throughput en PC-266

	Throughput (Mbps.)	Diferencia (Kbps.)	Diferencia en %
FIFO	9,72749		
SCFQ	9,7055	1	0,01
SPFQ	9,724	3	0,03
FFQ	9,7245	3	0,03

5 Conclusiones y líneas de futuros trabajos

La introducción del protocolo CIF permite que se puedan respetar parámetros garantizados del tráfico para distintos tipos de servicios. Estas garantías no son posibles utilizando Ethernet pura, al no poder especificar las características de cada tipo de tráfico.

Se ha realizado una implementación en el núcleo del sistema operativo, que permite ser usada por varios procesos. También se ha implementado un conmutador que soporta calidad de servicio.

Seguimos investigando en la búsqueda del algoritmo de encolado más eficaz para nuestro objetivo. Actualmente trabajamos en un nuevo algoritmo basado en el DRR que aporte una mejora en la uniformidad de la distribución del tráfico, para lo cual nos estamos apoyando en técnicas de simulación.

Para completar el escenario real, queda pendiente la incorporación de interfaces ATM en los conmutadores. También se completará la especificación CIF y el soporte de IP clásico que permita junto a las aplicaciones ATM nativas, la comunicación con otros ordenadores TCP/IP.

Referencias

- [1] J. M. Arco, A. Martínez, B. Alarcos, A. García, D. Meziat. "Quality of service over Ethernet". Online Educa Berlin, pp. 42-46, 2-4 December 1998,.
- [2] V. Castelo, "Informe de la gestión de RedIRIS durante el año 1997". Boletín de la Red Nacional de I+D RedIris, número 41-42, pp 5-7, Diciembre 1997.
- [3] A. Tanenbaum "Computer Networks, third edition", Prentice Hall, 1997.
- [4] J. McQuillan, "Convergencia de routes y conmutadores". Global Communication, número 10, pp 42-43, Enero 1998.
- [5] L. Roberts "CIF: Affordable ATM, At last" Data Communication, April, 1997.
- [6] Cells In Frames Version 1.0, specification, <http://www.ziplink.net/~lroberts/Atmf-961104.html>
- [7] Prycker M. "Asynchronous Transfer Mode, Solution for Broadband ISDN, Third Edition", Prentice Hall, 1997.
- [8] Armitage, J. "The Application of Asynchronous Transfer Mode to Multimedia and Local Area Network" PhD thesis, University of Melbourne, Australia, January 1994.
- [9] weikku.tkey.hut.fi/LDP/HOWTO/Module_HOWTO.html
- [10] J.M. Arco, B. Alarcos, A. Domingo, "Programación de aplicaciones en redes de comunicaciones bajo entorno Unix". Editado por la Universidad de Alcalá, 1997.
- [11] M. Beck, H Bohme, M Dziadzka, U Kunitz, R Magnus, D Verworner. "Linux kernel internals" Second Edition, Addison-Wesley, 1998.
- [12] Hui Zhang, "Service Disciplines for Guaranteed Performance Service in Packet-Switching Networks" Proceeding of the IEEE, Vol 83, N° 10 October 1995, pp. 1374-1396.
- [13] W. Stalling, "High-speed networks TCP/IP ATM design principles", Prentice Hall, 1998.
- [14] S Golestani, "A self-clocked fair queuing scheme for broadband applications" Proceeding of the IEEE INFOCO '94 Toronto, CA, pp. 636-646, June 1994.
- [15] D. Stiliadis and A. Varma, "Hardware Implementation on Fair Queuing Algorithms for Asynchronous Transfer Mode Networks," IEEE Communications Magazine, December. 1997.
- [16] Alessandro Rubini, "Linux device drivers". O'Really & Associates, Inc.1998.
- [17] Alan Cox, "Network Buffers and memory Management" Linux Journal, September 1996.
- [18] S. Branner, "Bechmarking Terminology for Network Interconnection Devices", RFC 1242.
- [19] J. M. Arco, A. Martínez, B. Alarcos, A. García D. Meziat. "Implementación de ATM sobre Ethernet para aplicaciones de Teleenseñanza", Actas IV Jornadas de Informática, Las Palmas de Gran Canaria, pp. 459-465, Julio 1998.
- [20] Kenjiro Cho. "A Framework for Alternate Queueing: Towards Traffic Management by PC-UNIX Based Routers." In Proceedings of USENIX 1998 Annual Technical Conference, New Orleans LA, June 1998.

Simulador ATM basado en Componentes

José M^a Quinteiro, M.Pilar Rosales, Ángelo Santana, Alfredo Sarmiento

Dpto. de Ingeniería Telemática. Universidad de Las Palmas GC

Campus Universitario de Tafira, Pab.C-205

35017 Las Palmas GC-Las Palmas

Tel: 928 451236, Fax: 928 451243

E-mail: jquinteiro@cma.ulpgc.es

Abstract

In this article we present a tool for system simulation with a wide support for ATM network simulation. It performs in two different environments. Firstly, aiding simulation developers (to create new protocols, sources or multiplexers) with a programming environment based on Object Oriented Programming (OOP) and Components, and secondly, aiding the user, whose sole objective is to implement a system for obtaining and analysing the results given from the simulation.

1. Introducción

Cuando se decide realizar un simulador existen tres opciones para implementarlo. La primera es utilizar lenguajes de programación de propósito general, como pueden ser C, C++, ADA y Modula-2, este tipo de lenguajes no están diseñados especialmente para su uso en simulaciones y aunque esta opción nos da gran versatilidad y potencia, presenta como contrapartida la necesidad de programar la mayor parte del código. La segunda opción consiste en utilizar lenguajes específicos para simulación como son GPSS [4], SIMAN [11], SIMSCRIPT [10], SLAM [13], éstos son lenguajes de alto nivel desarrollados específicamente para facilitar la construcción de modelos. La tercera opción es mediante paquetes específicos de simulación como NETWORK, COMNET [3], estos proporcionan un entorno integrado que permite al usuario construir un modelo, simularlo y analizarlo sin necesidad de hacer uso de ningún lenguaje de programación. Esta última solución es óptima cuando lo que se pretende simular es un sistema bastante estándar, ya que sólo se requiere la descripción de los bloques que constituyen el sistema, sus parámetros y la interconexión de los mismos. No obstante, es casi imposible utilizar estos paquetes para simular sistemas más complejos.

La implementación de esta nueva herramienta no se ajusta en su totalidad a ninguna de las tres opciones, intentando extraer de cada una de ellas las ventajas que aportan. La herramienta está totalmente integrada en un entorno de programación visual C++, lo que le da gran versatilidad y potencia. Por otra parte, la creación e integración de la “paleta” de simulación, basada en componentes, en un entorno visual facilita en gran medida el diseño de la red al usuario final, ya que éste no tendrá que introducir

necesariamente código para diseñar la red. El diseño se consigue simplemente añadiendo los elementos que forman parte de la red en la plantilla de programación, y para su interconexión sólo tenemos que hacer uso del valor de algunas de sus propiedades.

La existencia de un entorno integrado hace que esta herramienta disfrute de las ventajas que presentaban los simuladores basados en paquetes específicos de simulación, pero en este caso no nos vemos limitados en cuanto a la complejidad del sistema a simular.

Además, esta nueva herramienta suministra un interfaz entre el código interno del simulador y el usuario final que quiera ajustar o interactuar con el simulador. Esta interfaz se proporciona mediante la llamada de eventos por parte del simulador, con lo que el usuario final la puede utilizar sin más que escribir código en el punto de entrada del evento. Este modo de funcionamiento, junto con lo descrito anteriormente para diseñar redes, a veces se lleva a cabo mediante el uso de scripts, como en el caso de los simuladores implementados mediante la segunda opción.

En la actualidad, la investigación sobre el Modo de Transferencia Asíncrono (ATM) se está expandiendo rápidamente, ya que se trata de un protocolo flexible, escalable y de gran potencia. Además, es considerado como el soporte para la construcción de B-ISDN[5]. Debido a ello, han surgido numerosos simuladores como son OSSCAR [12], YATS [14], Libra [2] o ATMTrap [6].

Uno de los principales objetivos en el diseño de la herramienta ha sido mantener la compatibilidad con la definición de objetos de red (con algunas

pequeñas modificaciones) útiles para otras herramientas de simulación, en particular YATS.

El desarrollo de esta nueva herramienta de simulación se ha realizado sobre una plataforma Windows NT, realizando las simulaciones en una máquina Dual Pentium. Los componentes se han implementado bajo un entorno de programación visual Borland Builder C++.

2. Arquitectura del Sistema

En la mayoría de los simuladores de tiempo discreto que se utilizan para el estudio de redes ATM, el núcleo está constituido por un simulador de eventos y un intérprete, y la descripción del sistema se realiza mediante lenguajes no estándar, scripts.

Una vez que los eventos son generados, se almacenan en una lista para posteriormente ser activados por el núcleo del simulador. Se suele utilizar una sola lista para almacenar los eventos, aunque en alguna arquitectura, como es el caso de YATS, Fig.1, se opta por dividir la lista en pequeñas sublistas a las que se accede mediante funciones hash. La función hash que determina en que sublista se va a almacenar el evento viene dada por el módulo del tiempo de activación y el número de sublistas.

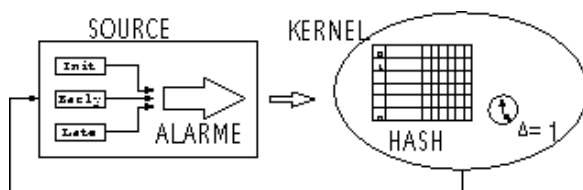


Figura 1

Las sublistas no están ordenadas y los nuevos eventos simplemente se añaden por la cabecera de la lista. Cada sublista se procesa desde el principio hasta el final, ya que el proceso de búsqueda en el caso de listas no ordenadas debe ser completo. Además, el tiempo en el simulador sólo se puede incrementar en una unidad en cada paso. El número de sublistas es un factor determinante en el rendimiento del simulador, por tanto, si el usuario final quiere cambiar este valor tanto el núcleo como los objetos de red tienen que ser recompilados, lo que puede suponer un inconveniente.

La arquitectura propuesta se presenta de la siguiente forma. El núcleo está constituido por un gestor de eventos, y dado que éste está incluido en un entorno de programación visual, no se incluyen elementos típicos de un intérprete, como pueden ser un analizador léxico o sintáctico. Como resultado se obtiene un código más puro y transparente.

Con el objetivo de mejorar el manejo de eventos se ha optado por el método tradicional en el que se utiliza una sola lista ordenada, Fig.2, a diferencia de aquellos sistemas de almacenamiento basados en funciones hash y colas calendario. De esta forma, el

tiempo de simulación puede ser incrementado en más de una unidad en cada paso si el estado actual así lo permite.

El reloj del simulador se ha diseñado para trabajar con variables aleatorias, tanto en tiempo discreto como en tiempo continuo. El gestor de eventos divide cada slot en dos fases, *early* y *late*. Aunque esta división no es necesaria en tiempo continuo, se realiza para tratar la simultaneidad de eventos, típica de los sistemas en tiempo discreto que son tratados como un subconjunto de los sistemas en tiempo continuo.

Los métodos, *alarme* y *alarml*, son los responsables de la generación de eventos y de la introducción de los mismos en la lista ordenada. Los métodos de alarma se llaman en primera instancia por el constructor de cada objeto y posteriormente por los métodos *early* y *late*.

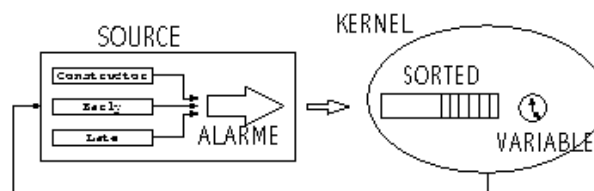


Figura 2

La paleta utilizada por el usuario para realizar el diseño de la red está constituida por objetos de red de YATS, que han sido adaptados, y por nuevos tipos de fuentes y multiplexores, entre los que podemos destacar modelos VBR autosimilares [1][7], ficheros MPEG [8][9] y muestras de tráfico en tiempo real.

3. Modos de Trabajo

En el manejo de la herramienta propuesta distinguimos dos modos de trabajo: modo usuario y modo programador.

3.1. Modo Usuario

El diseño de redes es asistido mediante el uso de componentes, no siendo necesaria la descripción de la red mediante el uso de scripts. Estos aunque simples, pueden resultar tediosos cuando el usuario intenta utilizar una nueva herramienta.

El uso de componentes para caracterizar a cada uno de los elementos de la red, tales como fuentes y multiplexores, facilita la especificación de cada uno de los parámetros asociados a cada elemento, ya que estos se ajustan con los valores de las propiedades asociadas a cada componente y mediante el tratamiento de eventos. Además, el uso de propiedades nos permite relacionar los componentes entre sí, así como con el núcleo del simulador.

Las funciones miembro de una clase se implementan como métodos en un componente, esto implica que para que se puedan ejecutar dichas funciones el usuario final tiene que escribir código. Una de las

ventajas que presenta esta herramienta es que se ha creado una interfaz para algunas funciones miembro mediante el uso de propiedades, de forma que el usuario no necesita escribir código.

Como se puede observar, el diseño de red asistido por el uso de componentes facilita en gran medida la ejecución de las tareas necesarias para llevar a cabo una simulación.

Otro aspecto a destacar es el hecho de que esta herramienta de simulación permite la ejecución de ciertos grupos de procesos en paralelo, lo que mejora el funcionamiento en plataformas multiproceso sobre máquinas multiprocesadoras. Por ejemplo, si decidimos que las variables aleatorias de las fuentes se generan en la fase *early*, éstas se pueden ejecutar en diferentes hilos (*threads*), resultando una ejecución concurrente de esta fase. En la Fig.3 se muestra el funcionamiento en paralelo del simulador, así como los puntos de espera entre fases.

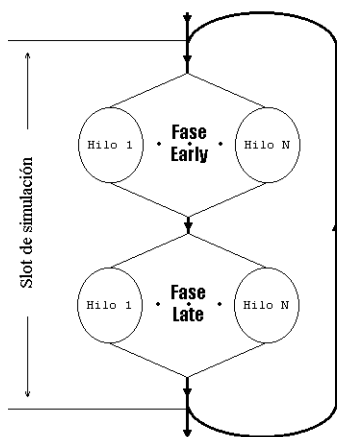


Figura 3. Simulación en paralelo

No obstante, debido a que cierto código se tiene que ejecutar en exclusiva, como el registro de los eventos en la cola, algunos hilos pueden tener que esperar en determinados momentos hasta que finalice la sección crítica ejecutada por otro hilo. Además, hay que tener en cuenta la gestión de los hilos en sí mismos, por parte del sistema operativo, así como las secciones críticas, por lo que no siempre se obtiene el rendimiento deseado. Estas dos limitaciones quedan reflejadas en los resultados obtenidos en las simulaciones de 20 fuentes CBR correladas. En este caso, se obtiene un incremento del 30% del tiempo de simulación con un simulador con dos hilos respecto a un simulador con un único hilo de ejecución. Este resultado se debe a que las dos limitaciones anteriormente reseñadas se ven amplificadas por el hecho del poco procesamiento que necesitan las fuentes CBR. En cambio, aumentando artificialmente el código a ejecutar por las fuentes, se consiguen reducciones en el tiempo de simulación significativas al emplear dos hilos. Por lo tanto, el funcionamiento multihilo es ventajoso en aquellos casos en los que se simule objetos de red pesados.

3.2. Modo Programador

En algunos simuladores, la creación de un nuevo objeto de red implica la recompilación de todo el código. Esto se debe a la posible dependencia entre módulos, lo que implica tiempo extra en la depuración. El diseño de redes asistido por componentes, elimina este tiempo extra debido a su naturaleza independiente. Teniendo en cuenta que un componente es un objeto, y que puede ser heredado a partir de otros ya existentes, la modificación de uno de estos componentes no implicará la recompilación de todo el código, sino únicamente la de la parte modificada.

Nos hemos visto beneficiados de todas aquellas ventajas introducidas por la programación orientada a objetos, OOP, en la que existe todo un conjunto de paradigmas típicos, tales como la derivación de una clase a partir de otra. Algunos de éstos se han utilizado, haciendo posible añadir nuevas variables, sobrecargar operadores o añadir nuevas funciones miembro.

La interfaz gráfica necesaria en el simulador se proporciona mediante el uso de componentes, consiguiendo así un diseño interactivo con el usuario final.

Uno de los elementos básicos de los componentes son las propiedades. Generalmente, una propiedad puede parecer una simple variable, de la que se puede obtener su valor o bien asignarlo. No obstante, una propiedad es algo más complejo, ya que puede llevar asociada la ejecución de métodos pertenecientes al objeto durante el tiempo de diseño. Esto supone una gran ventaja, ya que simplemente cambiando el valor de una propiedad se pueden llevar a cabo un conjunto de acciones.

Un componente también dispone de métodos, éstos no son más que funciones públicas que pueden ser utilizadas por parte del usuario durante el tiempo de ejecución. El uso de estos métodos nos permite dar órdenes al núcleo del simulador tales como: *Run*, si se desea simular la red para un número determinado de slots; *ResetTime*, si se quiere resetear el reloj del simulador; o cualquier otra orden que el programador del simulador crea que pueda ser útil al usuario final.

La simulación también se ve beneficiada por el tratamiento interactivo de los eventos. Estos eventos pueden ser causados, por ejemplo, por pérdidas de paquetes, llegadas o salidas de células de un multiplexor, o bien por anomalías en el proceso de simulación, tales como el desbordamiento de un contador. Esto último puede ocurrir si consideramos la calidad de servicio, QoS, de algunos servicios en redes ATM, como se muestra en la tabla 1 para la CLR (Tasa de Pérdida de Células):

Aplicación	Tasa de bit	CLR
Videoteléfono	64 Kbps	10 ⁻⁸
Videoteléfono	2 Mbps	10 ⁻⁸

Videoconferencia	5 Mbps	4×10^{-9}
Distribución TV	20-50 Mbps	10^{-10}
MPEG1	1.5 Mbps	10^{-8}
MPEG2	10 Mbps	2×10^{-8}

Tabla 1. CLR en varias Aplicaciones B-ISDN.

En algunos simuladores el tiempo de simulación es implementado mediante un entero sin signo, cuyo rango es $0-4.2 \times 10^9$, esto supone un problema si se intenta simular, por ejemplo, el comportamiento de un servicio de videoconferencia. Como se puede observar con el CLR considerado en la Tabla 1, después de 4×10^9 células, sólo se tendría, por término medio, una pérdida de una célula, que es obviamente insuficiente para un análisis estadístico.

El tratamiento de eventos, permite simplificar las decisiones sobre qué acciones se deben tomar dependiendo de las condiciones dadas por el simulador. De esta manera, el código de los componentes no se sobrecarga de forma excesiva y se sigue manteniendo su transparencia.

4. Conclusiones

En este artículo, hemos presentado una nueva herramienta que actúa en dos entornos diferentes. En primer lugar, intenta mejorar la productividad del programador del simulador, permitiéndole concentrarse exclusivamente en el funcionamiento de los objetos de red y no en todos aquellos aspectos relacionados con la presencia de un intérprete. El uso de componentes facilita la programación visual de los objetos de red sin tener que escribir código específico para un entorno visual. Además, el programador no pierde versatilidad, ni potencia, ni la

flexibilidad de un lenguaje de propósito general, ya que esta herramienta se integra en un entorno de programación visual C++.

En segundo lugar, el usuario final dispone de varios niveles de interacción con el simulador. En el primer nivel, no es necesario teclear código. El usuario sólo requiere realizar un “*drag and drop*” desde la paleta, que contiene los elementos de red, a la plantilla y cambiar los valores de algunas propiedades para especificar los parámetros asociados a cada elemento de red. En el segundo nivel, se requiere código C++ en los puntos de entrada a los eventos, dando la posibilidad de poder ejecutar las acciones que se deseen ante la presencia de un evento determinado. En este punto el simulador podría ser integrado con otras librerías de programación o aplicaciones.

Otra de las ventajas introducidas por esta nueva herramienta de simulación es el uso de hilos, que nos permiten ejecutar de forma paralela ciertos grupos de procesos. Sin embargo, en aquellas simulaciones en las que la carga no resulta muy pesada no se obtienen ventajas frente a la simulación secuencial debido a la gestión de los hilos, y al manejo de las secciones críticas fundamentalmente.

Finalmente, el simulador se presenta con un soporte amplio para la simulación de redes ATM.

En la Fig. 4 se muestra la interfaz de usuario para el diseño de redes y posterior simulación, y en la Fig. 5 se pueden observar los resultados obtenidos para una simulación concreta:

Bibliografía

- [1] “Variable Bit Rate video traffic and long range dependence”, J. Beran, R. Shreman, M. Taqqu, W. Willinger. IEEE Transactions on Communications, Vol 43, Nº 2-4, part.3, pp 1566-1579, Feb-Apr 1995.
- [2] “Design and Implementation of An Object-Oriented ATM Network Simulator”. Wen-Shyen E. Chen, Chun-Hong Lin And Ying-Chung Chen. Proceedings of MASCOTS’96.
- [3] “NETWOR II.5, LANNET II.5 and COMNET II.5” Garrison, W. J., Simulation Conference, 1991. Proceedings., Winter, 1991, pp 172-176.
- [4] “The Application of GPSS V to Discrete System Simulation”, G. Gordon, P-H 1975.
- [5] “ATM Networks: Concepts, Protocols, Applications” R. Handel, M.N. Huber and S. Schroder. 2nd Ed., Addison Wesley. 1994.
- [6] “ATMTrap: An Asynchronous Transfer Mode Traffic and Performance Measurement Tool” C.A. Johnston, D.J. Smith and K.C. Young Jr. Proc of INFOCOM’95, pp.220-229. April 1995.
- [7] “Analysis of an ATM Buffer with Self-Similar (“Fractal”) Input Traffic”, N.Likhanov, B.Tsybakov, N.D.Georganos. Proceedings of INFOCOM’95, Fourteenth Annual Joint Conference of the IEEE Computer and Communication Societies, Boston, MA, USA, Vol3, pp 985-992, 1995.
- [8] “Variable Bit Rate MPEG Video Characteristics, Modeling and Multiplexing”, D. Reininger, B.Melamed, and D. Raychaudhuri. Proc. Of the 14th International Teletraffic Congress France, Jun 1994 pp 295-306.
- [9] “Statistical Properties of MPEG Video Traffic and their Impact on Traffic Modeling in ATM Systems”, O. Rose. University of Würzburg, Institute of Computer Science, Research Report Series, Repot No. 101, Feb 1995.
- [10] “SIMSCRIPT II.5 and MODSIM II: A brief introduction”, Russell, E. C., Simulation Conference, 1991. Proceedings., Winter, 1991, pp 62-66.

- [11] "Introduction to SIMAN", Sturrock, D. T., Pegden, C. D., Simulation Conference, 1990. Proceedings., Winter, 1990, pp 109-114.
- [12] "Simulationsumgebung OSSCAR". Roya Ulrich. TR 19/94, Informatik VII, Universität Erlangen-Nürnberg, 1994.
- [13] "SLAM - A New Continuous Simulation Language", N.A. Wallington et al, in SCS Simulation Council Proc Series: Toward Real-Time Simulation (Languages, Models and Systems), R.E. Crosbie et al eds, 6(1):85-89 (Dec 1976).
- [14] Dresden University of Technology, Communications Laboratory, Germany, [ACTS project EXPERT (AC 049)]. <http://www.ifn.et.tu-dresden.de/TK/yats/yats.html>.

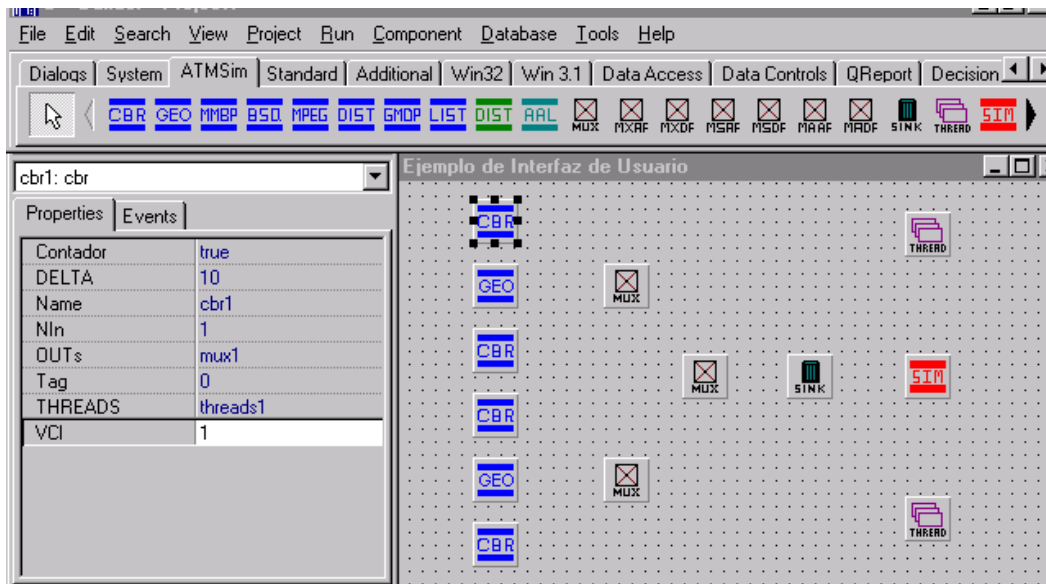


Figura 4. Interfaz de usuario

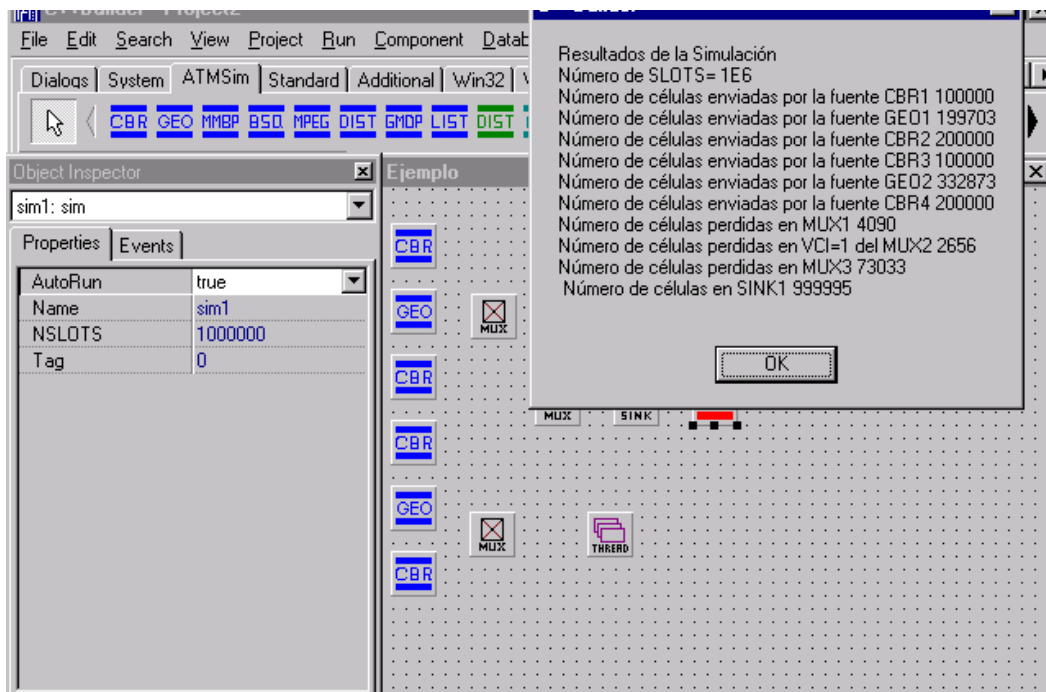


Figura 5. Resultados de la simulación

Funcionamiento y prestaciones del protocolo HARAL (Hybrid Addaptive Random Access Protocol) para redes de cable HFC

LLuís Gutiérrez
Departament de Matemàtica Aplicada i
Telemàtica, EUPVG
Avda Victor Balaguer s/n
Universitat Politècnica de Catalunya
08800 Vilanova i la Geltrú - Spain
e-mail: {luisg} @ mat.upc.es

Sebastià Sallent
Departament de Matemàtica Aplicada i
Telemàtica
Campus Nord - Universitat Politècnica de
Catalunya
08034 Barcelona - Spain
e-mail: {sallent} @ mat.upc.es

Abstract

HFC (Hybrid Fiber Coax) networks are being deployed all over Europe nowadays. A plethora of new services are going to be distributed over this type of networks such as: Video-on-demand, teleshopping, teleworking, telelearning, access to Internet, etc. Current technology is based in cable-modems, which are not standardized, they follow partially what the consortium MCNS has decided, on the other hand the IEEE has finally approved a new standard, 802.14. The MAC protocol used over such networks is very important thus HFC networks should cope with the QoS. each type of traffic requires. We propose in this paper a new protocol named HARAL (**H**ybrid **A**ddaptive **R**andom **A**ccess **P**rotocol), its main features are to be robust, adaptive, simple and easy to implement, it also can be easily modified for different services, thus it can be configured to be blocked/non-blocked, windowed/not-windowed, fix/variable number of minislots, and so on. We also present some simulations and analytical results, all them compared to those protocols that are also being developed for similar purposes. We also think that HARAL could be suitable for other multiaccess networks such industrial, wireless or others.

1. Introducción

Las redes de cable que inicialmente fueron diseñadas para transmitir información en modo difusión desde un punto servidor a las distintas estaciones receptoras, se han actualizado para proporcionar a las estaciones una manera de comunicarse con la estación servidora y aumentar en gran medida el abanico de aplicaciones que esta innovación representa. En las redes actuales de cable coaxial (HFC, hybrid fiber coax) la topología es del tipo arborescente, y se utilizan dos canales: el llamado canal de bajada que utiliza la estación de cabecera de la red (*Headend*) para transmitir al resto de estaciones (downstream) en modo difusión, y el canal de subida (upstream) que utilizan las estaciones para enviar información a la estación cabecera en modo contienda, es decir, en el canal de subida todas las estaciones comparten el medio físico por lo que es necesario un protocolo MAC que controle el acceso al canal que minimice al máximo la pérdida de información debida a las colisiones, y que garantice el libre acceso por parte de todas las estaciones al canal.

El protocolo HARAL que presentamos es básicamente un protocolo de contienda, en el que se usan minislots para la reserva de ancho de banda, y colas distribuidas para la petición y la transmisión. Además presentamos dos variantes: Normal y Mezcla. Las mejoras introducidas por nosotros respecto a otro similar, DQRAP, hacen que HARAL sea más simple, dinámico y robusto, además de adaptable a la cantidad de tráfico que hay en cada momento. Comprobaremos que las simulaciones realizadas muestran claramente que este protocolo mejora los resultados en cuanto a rendimiento y retardo respecto de protocolos anteriores.

2. Funcionamiento del protocolo HARAL

El protocolo HARAL organiza el canal en tramas (slots), en las que se pueden apreciar dos partes bien diferenciadas: el campo de datos (*Data Slot*, DS) y el campo de minislots de control (*Control Mini Slot*, CMS), ver figura 1. El primero es utilizado por las estaciones para transmitir la información en forma de paquetes de longitud

fija y el segundo lo utilizan las estaciones para transmitir las peticiones de reserva. Realizar una reserva consiste en elegir aleatoriamente uno de los minislots de que consta la trama (tirar un dado de m caras, siendo m el número de minislots que contiene la trama actual) y transmitir en él un señal de petición. Se supone que el tipo de tráfico será en forma de paquetes y que el canal estará ranurado, de modo que cada ranura estará ocupada por una celda ATM.

Para que una estación pueda acceder a los dataslots debe haber realizado con anterioridad una reserva, que debe haber sido reconocida por la cabecera. Por lo tanto las transmisiones en los dataslots están libres de colisión.

En cambio los minislots son de tipo contienda, es decir se pueden producir colisiones. Se dice que un minislot ha colisionado cuando se ha utilizado para transmitir más de una petición, o sea, ha sido elegido por más de una estación. Cuando esto ocurre se pierde toda la información que las estaciones colisionantes habrían transmitido.

La realimentación utilizada por la cabecera para indicar el estado de cada minislot en la trama anterior es de tipo ternaria: vacío (no se utiliza por ninguna estación), válido (utilizado por una única estación) o colisión (utilizado por más de una estación).

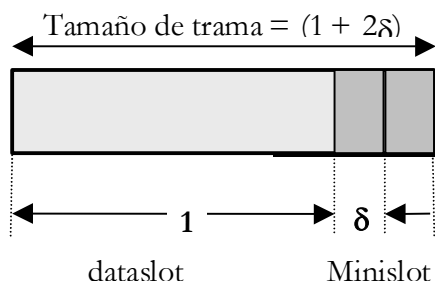


Figura 1. Formato básico de trama

Partiremos de una trama base (estado inicial o de reposo) formada por un dataslot y dos minislots, esta trama puede variar tanto en la cantidad de dataslots como de minislots para adaptarse a las necesidades del tráfico. Para cargas bajas utilizaremos la trama base, y a medida que aumenta el tráfico podemos llegar hasta dos slots de datos y 5 o 6 minislots.

En nuestro protocolo HARAL, cada estación mantiene dos colas: La cola de transmisiones (*Transmission Queue*, TQ) y la cola de peticiones (*Request Queue*, RQ). La primera se comporta como una cola FIFO en la que la estación que se encuentra en la primera posición de la cola será la primera estación en transmitir.

La segunda funciona en modo contienda, en ella permanecen las estaciones que intentan reservar un dataslot usando el mecanismo de los minislots. El principio básico de los protocolos en árbol es el de resolver una colisión antes de intentar otra (bloqueo). Esto implica que al producirse una colisión el sistema se bloquea y las nuevas estaciones transmisoras no pueden acceder a los minislots hasta que éstos no se han resuelto satisfactoriamente.

El protocolo HARAL se puede configurar en modo *Normal* o modo *Reserva*, la diferencia estriba en el funcionamiento de la cola RQ. Todo ello se entenderá mejor observando detenidamente el funcionamiento exacto del protocolo resumido en las siguientes reglas:

a) Reglas de la cola de transmisión:

- Si $TQ=0$ y $RQ=0$: Las estaciones con mensajes que hayan llegado en el intervalo hábil de peticiones (ETI, *Enable Transmission Interval*) transmiten este mensaje en el dataslot (acceso directo) y además también realizan su petición en un minislot elegido aleatoriamente. Si el acceso directo se resuelve con éxito (sin colisión) se logrará una transmisión óptima, con retraso igual a cero.
- Si $TQ>0$: La estación a la que pertenece la primera posición de la cola TQ transmite su mensaje en el dataslot.
- Cada estación incrementa una posición la cola TQ por cada minislot que ha llegado con éxito a la cabecera. Esta cola indica el número de paquetes que esperan a ser transmitidos, y por tanto cuando la cabecera recibe una petición significa que el paquete de la estación que la realizó se transmitirá dentro de q tramas (siendo q la longitud de la cola TQ).
- Cada estación decreta una posición la cola TQ por cada paquete transmitido con éxito. La cola TQ se vacía a medida que se van transmitiendo los paquetes en los dataslots.

b) Reglas de la cola de resolución de colisiones:

- Si $RQ=0$: Las estaciones con mensajes llegados dentro de la ETI actual transmiten las peticiones (una por cada estación). En caso de colisión cada estación incrementa n posiciones la cola RQ, siendo n el número de minislots colisionados (seleccionados por más de una estación). Mediante este sistema la resolución de la contienda se podrá realizar de forma independiente según el minislot en que se haya producido.
- Si $RQ>0$: Las estaciones a las que pertenece la primera posición en RQ

transmiten las peticiones. En caso de colisión cada estación incrementa $n-1$ posiciones la cola RQ, siendo n el número de minislots colisionados.

c) Reglas de funcionamiento general:

- Se definen unos umbrales máximo y mínimo que limitan el número de minislots que podrán tener las tramas (figura 2).
- Si el porcentaje de minislots que han colisionado es superior al umbral de incremento del número de minislots, se incrementará en uno el número de minislots en la próxima trama, hasta el valor máximo (m_{max}).
- Si el porcentaje de minislots que han colisionado es inferior al umbral de decremento del número de minislots, se decrementará en uno el número de minislots hasta el valor mínimo (m_{min}).
- De igual manera, cuando la cola TQ alcance un valor determinado (TQ_{max}) el número de dataslots de la trama siguiente se incrementará a 2.
- Cuando la longitud de la cola TQ disminuye al valor umbral determinado (TQ_{min}), en la siguiente trama el número de dataslots volverá a ser uno.

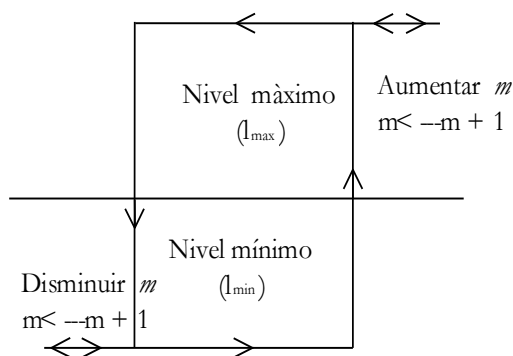


Figura 2. Histéresis del número de minislots

El hecho de variar de forma dinámica el número de dataslots y de minislots en cada trama hace que haya una mejor adaptación a la cantidad de tráfico que hay en cada momento, traduciéndose en un mejor rendimiento y una disminución considerable del retardo en comparación con otros protocolos.

3. Simulación del protocolo en modo contienda

En primer simularemos el funcionamiento de nuestro protocolo en un sistema con un solo tipo de tráfico. Supondremos un tráfico de tipo

Poisson, o sea todas las estaciones generan un tráfico agregado cuya tasa (λ) será constante.

La simulación de este protocolo se ha realizado utilizando los siguientes parámetros:

- Número de minislots por trama: de 2 a 5
- Número de dataslots por trama: de 1 a 2
- Umbral aumento nº minislots: 34%
- Umbral decremento nº minislots: 21%
- Umbral variación nº dataslots: 10

Comparamos los resultados obtenidos por simulación con otros protocolos como el DQRAP con 3 minislots [1] y el MFA [2].

En la figura 3 mostramos los resultados obtenidos del caudal (rendimiento). En primer lugar cabe observar el caudal de salida del protocolo HARAL, en ambas variantes, y el DQRAP se comportan como una cola ideal del tipo M/D/1 hasta casi el 100%. El protocolo MFA de una sola cola tiene un rendimiento muy inferior como cabía esperar.

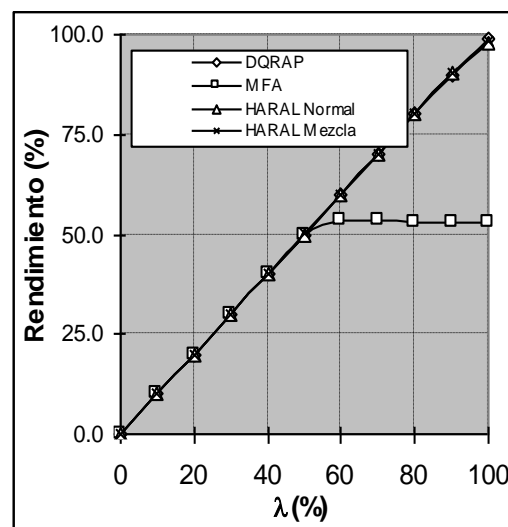


Figura 3. Caudal (Throughput)

En la figura 4 observamos el retardo debido a la resolución de colisiones. En este caso HARAL se comporta mucho mejor cuando la red está próxima a la saturación ($\lambda = 100\%$).

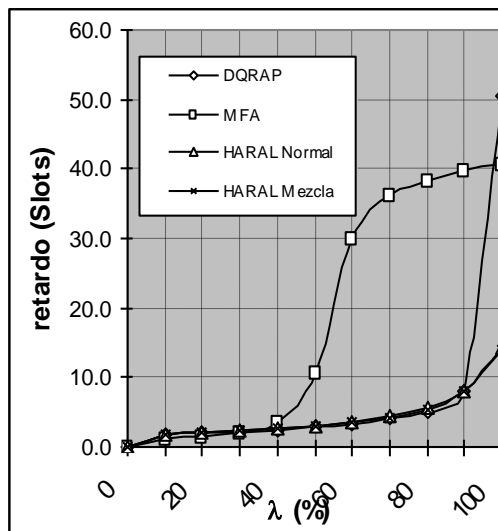


Figura 4. Retardo (Delay)

En comparación con otros protocolos, el rendimiento es muy parecido al DQRAP con $m=3$, y bastante superior al MFA con $m=3$, pero donde se aprecian las grandes mejoras es en el apartado del retardo que es mucho menor que en el resto de protocolos gracias al control de la longitud de cola TQ que influye directamente en el número de datslots que contiene cada trama.

Como se puede observar los resultados obtenidos mediante los protocolos HARAL Normal y Mezcla son muy parecidos tanto en rendimiento como en retardo.

4. Comparación para distintos tipos de tráfico ATM

Pensando en una implementación actual donde se están imponiendo las redes HFC y que mayoritariamente soportarán tráfico ATM (*Asynchronous Transfer Mode*), se ha adaptado el protocolo HARAL a estas características.

El modelo a seguir para los tipos de tráfico se tomó de la referencia [3]. En nuestro escenario de simulación distinguiremos cuatro clases de tráfico, cada uno de ellos requiere una calidad de servicio (QoS) distinta:

- CBR (*Constant Bit Rate*): Consideraremos dos tipos diferentes de llamadas CBR: las llamadas de voz (75%) y las de videotelefonía (25%).
- VBR (*Variable Bit Rate*): Tráfico con tasa variable, sigue el proceso ON/OFF.
- ABR (*Available Bit Rate*): Tráfico de baja prioridad utilizado por fuentes de datos. Se considera que no existe MCR ($MCR=0$).
- UBR (*Unspecified Bit Rate*): Igual que las fuentes ABR pero de menor prioridad.

En realidad tal como está planteado este escenario tendremos 2 prioridades de tráfico. Siempre que tengamos paquetes del tipo CBR o VBR accederán antes al canal que los del tipo ABR o UBR.

Para la simulación de las distintas clases de tráfico se han utilizado dos escenarios en los que varían las proporciones de cada clase de tráfico. Así en el primer escenario denominado de voz, predomina la ocupación del canal por llamadas de voz y videotelefonía (60% CBR, 20% VBR, 10% ABR y 10% UBR), mientras que en el segundo denominado de datos aumenta la proporción de la ocupación por parte de llamadas de baja prioridad (30% CBR, 10% VBR, 30% ABR y 30% UBR).

Nuestro objetivo es ver hasta que punto HARAL es capaz de cumplir con los requisitos (QoS) que cada una de las clases de tráfico requiere y compararlo también con protocolos similares propuestos para las redes de cable (IEEE 802.14): MLAP [4], ADAPt [5], i XDQRAP [6].

Los parámetros que se estudiarán y que mejor definen el funcionamiento de un protocolo son los siguientes:

- Rendimiento (similar al *Throughput*): Es la relación entre el número de bits de información útil transmitidos con éxito y el número total de bits transmitidos.
- CTD (*Cell Transfer Delay*): Retardo medio para cada tipo de tráfico. Para las fuentes CBR y VBR se entiende como el tiempo que transcurre desde que es generada una celda ATM por una fuente hasta el momento en que la celda se transmite completamente. Las fuentes ABR y UBR producen paquetes de longitud variable (de media 30 slots), y el CTD para estas fuentes se considera como el tiempo que transcurre entre la generación del paquete y el instante en que todos los slots que forman el paquete pueden ser transmitidos.
- CDV (*Cell Delay Variation*): Es la separación relativa entre celdas ATM que son transmitidas con éxito. Este parámetro cobra importancia para el tráfico CBR y VBR que tienen que cumplir la QoS, y para ello el CDV no puede superar un valor máximo denominado CDVT (*Cell Delay Variation Tolerance*).

La simulación de este protocolo se ha realizado utilizando los siguientes parámetros:

- Número de minislots por trama: de 2 a 5 en modo Normal (escenario voz y datos).

- Número de minislots: de 3 a 6 sólo en el caso de la variante Mezcla para el escenario de voz.
- Umbral aumento n° minislots: 34%
- Umbral decremento n° minislots: 21%
- Velocidad de transmisión en el canal de subida (upstream) 2 Mbps.
- Tamaño de trama básica 64 bytes (dataslot).

Analizamos en primer lugar (figuras 5) el rendimiento en el escenario de voz del protocolo Mezcla (0,6848 para $\lambda=100\%$) es muy parecido al del XDQRAP (0,6761), y se ven ampliamente

superados por el protocolo Normal (0,7762) y MLAP (0,8011). En este escenario se produce un gran número de contiendas, debido al mayor número de llamadas CBR y VBR, y por tanto hemos necesitado que el protocolo Mezcla tuviese un mínimo de 3 minislots. Por el contrario en protocolo Normal consigue un rendimiento mayor con un mínimo de 2 minislots.

En el escenario de datos (figura 6) tanto el protocolo Normal como Mezcla (0,8190 y 0,8220 respectivamente) mejoran el rendimiento del protocolo XDQRAP (0,7611).

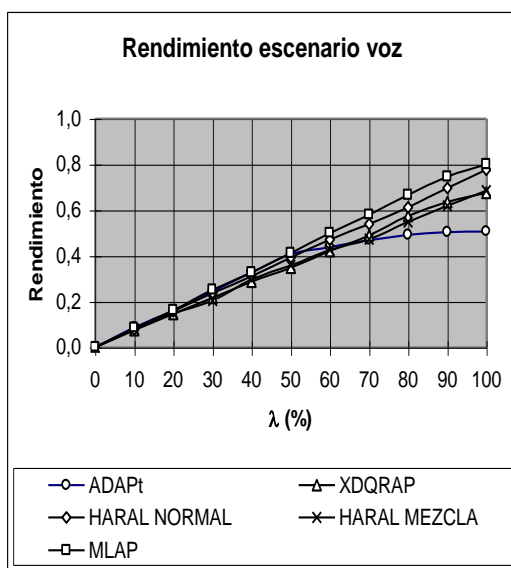


Figura 5. Comparación del rendimiento en escenario de voz

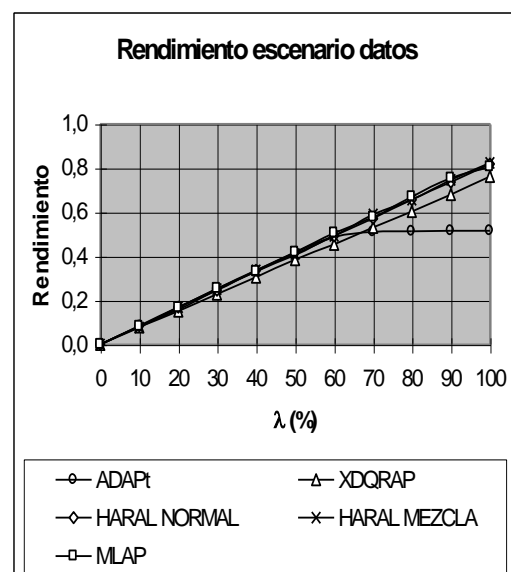


Figura 6. Comparación del rendimiento en escenario de datos

En las figuras 7 y 8 se analiza el retardo global (CTD, *Cell Transfer Delay*). El CTD en el escenario de voz es muy parecido en todos los protocolos hasta una λ del 90% donde en el protocolo Normal se dispara debido a que utiliza un menor número de minislots para resolver la contienda.

En el escenario de datos el protocolo Normal se mantiene igual que el XDQRAP hasta λ del 90%, donde empieza a aumentar, aunque sin dejar de cumplir con la QoS.

En el tráfico del tipo CBR para cumplir la QoS se debe tener muy en cuenta la varianza del retardo, (CDV, *Cell Delay Variation*). Observar en la figuras 8 y 9 este valor.

De las gráfica anteriores podemos extraer la conclusión de que el protocolo HARAL mejora las prestaciones respecto al DQRAP, y además como es muy versátil podemos configurarlo diferente según tengamos un escenario más restrictivo respecto a la QoS requerida.

El protocolo MLAP parece superar al resto en la mayoría de simulaciones mostradas, pero debe tenerse en cuenta que este protocolo utiliza mecanismos de reserva y por lo tanto su implementación es mucho más compleja.

También queremos remarcar, que los minislots se deben considerar como un *overhead* de control, es por ello que el hecho de ajustar su número de forma dinámica hace que el rendimiento de total del protocolo sea mayor.

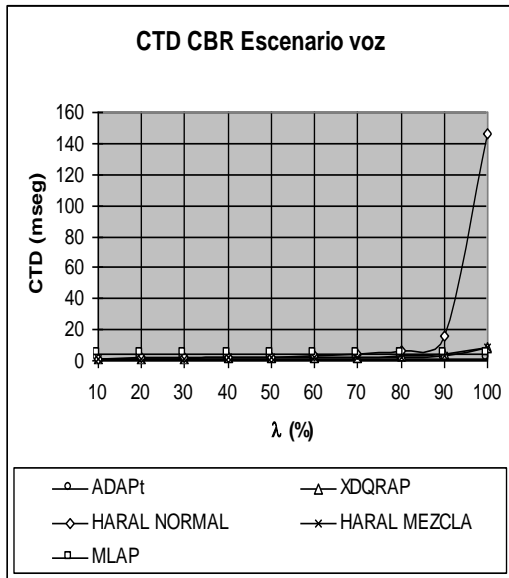


Figura 7. Comparación del retraso en escenario de voz

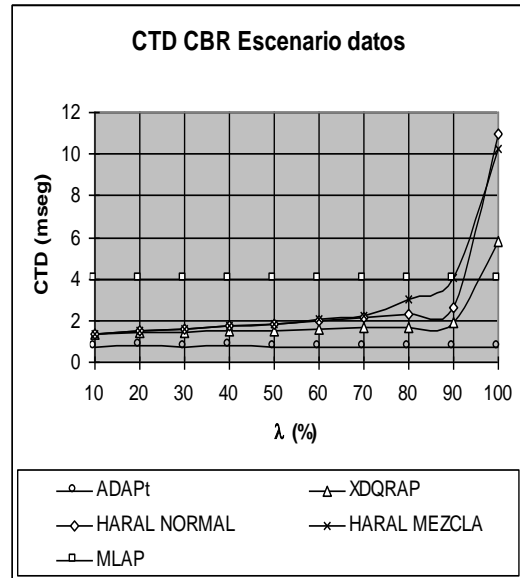


Figura 8. Comparación del retraso en escenario de datos

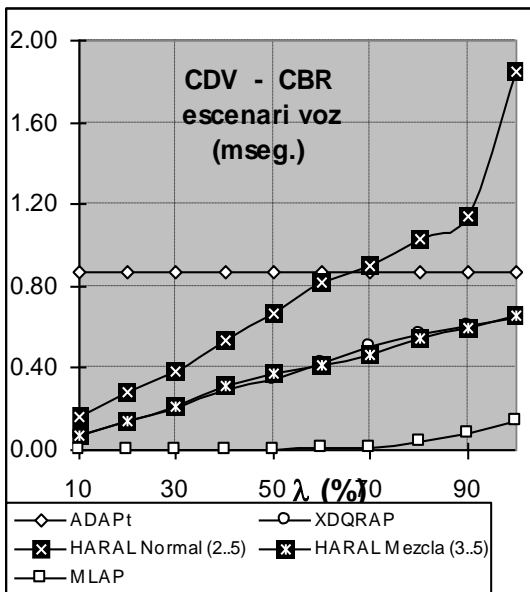


Figura 9. Comparación del CDV para tráfico CBR (escenario voz)

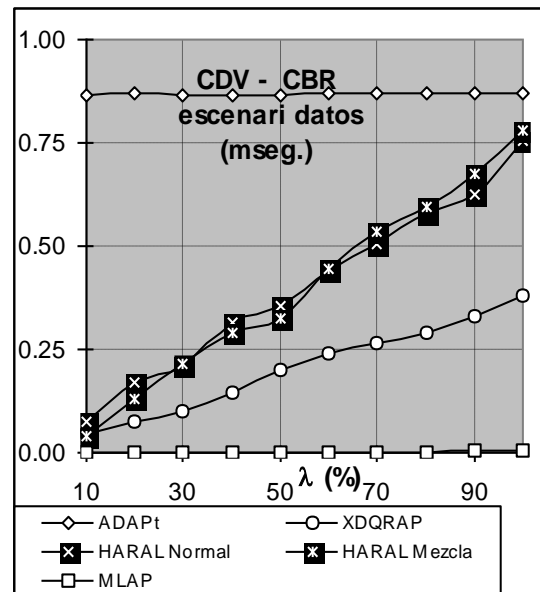


Figura 10. Comparación del CDV para tráfico CBR (escenario datos)

5. Análisis matemático de las prestaciones del protocolo HARAL

La evaluación analítica de las prestaciones de protocolos como HARAL acostumbra a ser una ardua tarea, afortunadamente nosotros podemos partir de la teoría ya desarrollada en anteriores trabajos [7][8].

En primer lugar suponemos un canal ideal, de modo que nuestro canal tiene las características siguientes: estable, bloqueado, tráfico de Poisson, libre de errores, ranurado, escucha continua, infinito número de estaciones, respuesta inmediata y sin almacenamiento. Nuestro objetivo principal será calcular las prestaciones que nos permiten conocer el comportamiento de nuestro protocolo, y que además permitan compararlo con otros similares.

El estudio analítico lo realizamos para la cola de petición (RQ) que es donde se producen las colisiones.

5.1. Escenarios de cálculo

En primer lugar fijaremos distintos escenarios para el cálculo analítico (tabla 1) , variando los valores: m_{min} , m_{max} , l_{inf} , y l_{sup} en las variantes Normal y mezcla (tabla 1).

Tabla 1. Escenarios para el cálculo analítico de HARAL

escenario	l_{inf} (%)	l_{sup} (%)	m_{min}	m_{max}	Nor mal	Mez cla
1	17	34	2	6	HN1	HB1
2	17	26	2	6	HN2	HB2
3	21	34	2	5	HN3	HB3
4	21	26	2	5	HN4	HB4
5	21	34	3	6	HN5	HB5
6	21	26	3	6	HN6	HB6

El cálculo analítico de los datos nos ha permitido en primer lugar conocer la incidencia en el

cálculo de L_n de los valores configurables, y como consecuencia de ello en el resto de parámetros para ambas variantes normal y mezcla.

5.2. Cálculo de los parámetros principales.

Para conocer el comportamiento de un protocolo de contienda debemos calcular en primer lugar la longitud media del intervalo de resolución de colisiones ($L_{n,m}$) cuando tenemos n paquetes colisionados y m minislots al inicio de la colisión. Estos valores son fundamentales para hallar el resto de parámetros.

El cálculo de $L_{n,m}$ debe realizarse utilizando probabilidades y recursividad, en el caso del DQRAP el valor de m es fijo (ver [1]), en este caso podemos hallar la fórmula cerrada para el cálculo de $L_{n,m}$ en función del valor de m i de los valores de $L_{j,m}$ ($j < n$). En nuestro caso el cálculo se complica algo más por el hecho de que m varía según haya sido el resultado de la colisión [9].

Tabla 2. Valores calculados analíticamente de $L_{n,2}$ del protocolo HARAL

$L_{n,2}$	HN3	HN4	HN1	HN2	HN6	HN5	HB4	HB3	HB2	HB1	HB6	HB5
0	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00
1	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00	1,00
2	1,7500	1,7222	1,7500	1,7222	1,4444	1,5000	1,7222	1,7500	1,7222	1,7500	1,4444	1,5000
3	2,6393	2,6043	2,6393	2,6043	2,0847	2,2500	2,6043	2,6393	2,6043	2,6393	2,0847	2,2500
4	3,6174	3,4793	3,6174	3,4793	2,8293	3,0385	3,4246	3,5738	3,4262	3,5751	2,7724	2,9812
5	4,5252	4,3140	4,5252	4,3139	3,5855	3,7548	4,2249	4,3942	4,2218	4,3907	3,4443	3,6125
6	5,3757	5,1139	5,3756	5,1136	4,3331	4,4328	4,9814	5,0863	4,9383	5,0444	4,1007	4,1999
7	6,1810	5,8908	6,1808	5,8901	5,0632	5,1112	5,7019	5,7530	5,5841	5,6381	4,7146	4,7643
8	6,9513	6,6534	6,9509	6,6520	5,7743	5,7956	6,4243	6,4467	6,2044	6,2285	5,3044	5,3277
9	7,6951	7,4065	7,6942	7,4042	6,4687	6,4778	7,1461	7,1558	6,8436	6,8541	5,9118	5,9221
10	8,4194	8,1525	8,4175	8,1488	7,1496	7,1535	8,1571	8,1613	7,5661	7,5705	6,8109	6,8154

En la tabla 2 se muestran los valores hallados de $L_{n,2}$ en los distintos escenarios y se comparan con el protocolo DQRAP.

Una vez conocidos los valores de $L_{n,m}$, se puede hallar el vector de probabilidad de estado del sistema ($\vec{\pi}$), y a continuación los valores esperados de: intervalo de resolución de colisiones (B), caudal (S, throughput), número medio de paquetes en el sistema (N), deriva (D,

drift), tasa límite de estabilidad (λ_{sty}) y una estimación de los retardos debidos a la cola inicial (Wq), a la resolución de las colisiones (Ws) y el retardo conjunto.

Finalmente, la tabla 3 muestra los valores hallados del resto de parámetros en el umbral de inestabilidad del protocolo (λ_{sty}).

6. Conclusiones

Hemos presentado el protocolo MAC HARAL, de propósito general en sus distintas versiones (Normal y Mezcla). Hemos podido observar que este protocolo ofrece mejoras en bastantes resultados respecto a otros protocolos ya existentes, tanto de forma simulada como analítica. Cabe destacar que además, la mejora de resultados viene acompañada de una mayor simplicidad en el funcionamiento, sobretodo en el protocolo HARAL mezcla donde las estaciones han de controlar una única cola.

Se ha visto que el protocolo es estable para cualquier tipo de cargas y que su comportamiento se adapta dinámicamente a las exigencias del tráfico, permitiendo dedicar el mayor ancho de

banda posible a la transmisión de datos y reduciendo al máximo la información de control que se “desperdicia”.

En el caso de tráfico ATM sus prestaciones son similares a las de otros protocolo, y donde de nuevo HARAL es destacable es por su sencillez, y facilidad de implementación.

Agradecimientos

Este trabajo no habría sido posible sin la colaboración de los estudiantes Jaume Bonet, Xavier Catalina, Victor Carceler, Felipe Edo y Jorge Salvo de la Escuela Politécnica de Vilanova i la Geltrú; a todos ellos mi agradecimiento por su interés y dedicación.

Tabla 3. Valores esperados del protocolo **HARAL** (cola RQ) en el límite de estabilidad (λ_{sty}).

	HN1 & HN3	HN2 & HN4	HN5	HN6	HB3	HB1	HB4	HB2	HB5	HB6
λ_{sty} (%)	75,397	76,249	81,403	83,287	75,671	75,705	76,434	76,466	81,637	83,512
B	1,3263	1,3115	1,2285	1,2007	1,3215	1,3209	1,3083	1,3078	1,2249	1,1974
N	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000	1,000
C	0,7540	0,7625	0,8140	0,8329	0,7567	0,7570	0,7643	0,7646	0,8164	0,8351
Wq	0,6632	0,6558	0,6142	0,6003	0,6607	0,6605	0,6542	0,6539	0,6125	0,5987
Ws)	1,7753	1,7507	1,6214	1,5786	1,7665	1,7654	1,7450	1,7440	1,6153	1,5731
Wg = Wq + Ws	2,4384	2,4064	2,2356	2,1789	2,4272	2,4258	2,3991	2,3979	2,2277	2,1718

Referencias

- [1] W. Xu and G. Campbell. *A Distribute Queueing Random Access Protocol for a Broadcast Channel*. DQRAP Research Group, Report 90-1, Computer Science Dept., Illinois Institute of Technology.
- [2] HUANG, J.-Ch., i BERGER, T. *Delay Analysis of Interval-Searching Contention Resolution Algorithms*. IEEE Trans. on Inf. Theory, vol. IT-31, n° 2, March 1985, pp: 264 - 273.
- [3] CURTIS, A. i SILLER, J. *Traffic Characteristics and Performance Parameters for Continous Rate Applications*. IEEE Project 802.14 Working Group, Doc. IEEE 802. 14-95/135, Nov. 1995.
- [4] Ch. Bisdikian, B. Neil, R. Mc. Norman y R. Zeisz, R. *MLAP: A MAC Level Access Protocol for the HFC 802.14 Network*. IEEE Comm. Magazine, vol. 34, n° 3, March 1996.
- [5] J. E. Dail y M. A. Dajer, y otros. *Adaptive Digital Access Protocol: A MAC Protocol for Multiservice Broadband Access Networks*. IEEE Comm. Magazine, vol. 34, n° 3, March 1996, pp: 104-112.
- [6] J. M. Ulm. *A MAC proposal for 802.14*. IEEE Project 802.14 Working Group, Doc. IEEE 802.14-95/134, Nov. 1995.
- [7] Ll. Gutierrez and S. Sallent. *A new method to analyze the performance of Multi-Access Protocols*. Proc. IPCCC'98, Feb. 98, Phoenix (USA), pp. 426- 434.
- [8] LL. Gutierrez, y S. Sallent. *Análisis y Simulación de protocolos de acceso a Redes de cable de Banda Ancha*. Proc. URSI'97, Sept. 1997, Bilbao (Spain), vol. II, pp: 425 – 428.
- [9] LL. Gutierrez, y S. Sallent. *Description and Performance Evaluation of HARAL, an efficient Protocol for Multi-access Channels*. Proc. EUNICE'99, Sept. 1999, Barcelona (Spain).

Control de acceso en entornos multicast: un enfoque a la autenticación de emisores

Antonio F. Gómez Skarmeta Angel L. Mateo Martínez Pedro M. Ruiz Martínez
Departamento de Informática, Electrónica e Inteligencia Artificial Universidad de Murcia
Facultad de Informática, Campus de Espinardo, s/n, 30001 Murcia
Telf: 968-364644, Fax: 968-364151
E-mail: { skarmeta,amateo,pedrom }@dif.um.es

Abstract

During the last years, there have been developed a lot of new interactive multimedia tools based on IP multicast over the Mbone. Although Mbone makes possible audio and video transmission over the Internet, it is used only by technicians related with the development of the Mbone because it has a lot of problems like badused bandwidth, no control over who is joined to a session, there is no way to control the scope of sent packets, ... These kinds of problems make Internet Service Providers (ISP) to think of Mbone like an immature technology. So we are trying to solve this kind of problems, especially those related with multicast senders control and authentication, by modifying multicast routing schemes.

1. Introducción

La gran popularidad que ha ganado Internet en los últimos años ha hecho posible el desarrollo de nuevos servicios interactivos como la videoconferencia. No obstante, dado el gran consumo de ancho de banda que necesitan este tipo de aplicaciones, es necesario poder disponer de una buena infraestructura que posibilite una buena utilización de los recursos.

Las soluciones actuales más populares, tales como NetMeeting y CU-SeeMe, se basan en la idea de los "reflectores" o nodos de la red que actúan como repetidores de información: todo lo que envía cada participante se reenvía al resto de participantes en la videoconferencia.

Frente a esta solución nos encontramos con aplicaciones basadas en el uso del IP multicast [2] como medio de transmisión en lugar del IP tradicional (unicast). El IP multicast se basa en utilizar direcciones de grupo en vez de direcciones que pertenezcan a un solo equipo. Así, un determinado equipo recibirá los paquetes dirigidos tanto a su dirección IP como a la correspondiente a alguno de los grupos a los que se haya unido. De este modo, se necesitan nuevos equipos de enrutamiento para poder trabajar con este tipo de direcciones destino recayendo la complejidad en las infraestructuras de red necesarias. Como contrapunto, se elimina la necesidad de utilizar equipos repetidores y se consigue una utilización óptima del ancho de banda en los enlaces.

Estos nuevos equipos de enrutamiento multicast llamados *mrouter*s, actúan del mismo modo que los routers normales sólo que empleando un algoritmo de enrutamiento multicast. Es decir, cuando les llega un paquete dirigido a un determinado grupo multicast por un determinado interfaz virtual, determinan a partir de su tabla de enrutamiento multicast el interfaz o interfaces de salida por el que

deben de reenviar este paquete. Básicamente se pueden distinguir dos tipos de *mrouter*s: los routers normales con soporte multicast y los hosts que ejecutan algún demonio de enrutamiento multicast (DEM).

Debido a que no todos los routers que existen actualmente en Internet incorporan soporte multicast, los equipos que hacen enrutamiento multicast se comunican mediante el establecimiento de túneles por los que se encapsulan los paquetes multicast. En concreto, esos paquetes multicast se encapsulan dentro de paquetes unicast dirigidos al *mrouter* situado al otro extremo del túnel. A toda esta infraestructura de túneles para soporte multicast es a lo que se ha llamado Multicast Backbone (Mbone [1]).

Durante los últimos años, los esfuerzos se han dirigido a la creación de nuevas aplicaciones, protocolos de tiempo real, etc. Sin embargo, no deben de descuidarse otra serie de aspectos que si bien no son tan vistosos, si que son igual de importantes y necesarios para hacer del multicast un servicio serio y maduro capaz de ser ofertado por los ISP's. Estos aspectos tienen que ver con el control de los usuarios que hay en un determinado grupo, el ámbito o Time To Live (TTL) con el que están emitiendo, etc.

Todo esto, convierte al IP Multicast en un servicio que los proveedores de servicios, ya sean estos proveedores comerciales, las universidades o cualquier otro tipo de proveedor de conexión a Internet, ven complejo, difícil de gestionar y difícil de utilizar, lo que provoca que se esté siendo infrutilizado.

Afortunadamente, los algoritmos y sistemas de enrutamiento han alcanzado un grado bastante alto de estabilidad, es decir, aunque el algoritmo de enrutamiento en sí no es el más eficiente que se puede utilizar, lo cierto es que una vez levantados correctamente los túneles necesarios, el sistema

logra un alto grado de estabilidad. En este sentido, los problemas aparecen cuando surge algún problema en la red multicast que hay que depurar, debido sobre todo a la falta de aplicaciones de depuración tales como las que puede haber para el IP Unicast.

Sin embargo, aunque la gestión de la red una vez que ésta se ha puesto en funcionamiento no requiere un gran esfuerzo, los proveedores continúan sin ofrecer masivamente el servicio, sobre todo debido a que no existe ningún sistema que permita controlar quién accede al Mbone y cómo. Así, los proveedores comerciales no lo pueden ofrecer como un servicio de valor añadido porque o lo ofrecen a todos los clientes o no lo ofrecen. Del mismo modo, las universidades si ofrecen el servicio a todos sus usuarios sufren el riesgo de ver considerablemente aumentado el tráfico en sus enlaces.

Así pues, para que el Mbone alcance el grado de popularidad necesario para que se aproveche toda la potencia que proporciona, es necesario definir un esquema que proporcione un cierto control sobre el uso que se hace de él. Con esto, se podría ofrecer el servicio a los clientes de los proveedores y alcanzaría un cierto grado de popularidad, que además provocaría el desarrollo de nuevas aplicaciones más amigables y vistosas de cara al usuario final.

Por lo tanto, el problema en el que nos centramos es el control de acceso de los usuarios a Mbone. Este sistema debería proporcionar un control sobre qué usuario recibe o envía sobre determinado grupo multicast. Sin embargo, esto constituye una solución definitiva y nosotros hemos optado por una solución intermedia que permita la autenticación de los emisores en los grupos multicast, pero que no hace ningún cambio en el sistema de recepción de información.

2. Funcionamiento del DEM

Aunque existen muchos algoritmos de enrutamiento multicast como PIM, DVMRP [3], etc., en todos los casos se trata de decidir el interfaz virtual (VIF) por la que se debe de reenviar cada paquete multicast que llegue al *mrouter*.

Para mejorar la escalabilidad y el consumo de recursos, algunos de estos algoritmos emplean el llamado *pruning*. El *pruning* consiste básicamente en evitar la retransmisión innecesaria de paquetes multicast a subredes en las que no hay ningún receptor interesado en esos paquetes. Pero, ¿Cómo sabe el DEM si hay algún host en alguna de sus interfaces interesado en un determinado grupo?. Para esto se emplea el Internet Group Management Protocol (IGMP [4]). Este protocolo permite que los equipos de una determinada subred puedan indicar al DEM que están interesados en un determinado grupo. Por lo tanto, es también tarea del DEM el

llevar este registro de los grupos asociados a cada interfaz virtual. Además veremos como el DEM está muy estrechamente ligado al soporte para reenvíos que da el Sistema Operativo (SO).

2.1. Soporte del SO al enrutamiento multicast

Como ya se ha mencionado antes, normalmente los paquetes multicast suelen formar parte de comunicaciones que se pretende que sean en tiempo real por lo que el tiempo de procesamiento de estos paquetes en cada uno de los nodos intermedios por los que va pasando debe de ser mínimo. Para ayudar a acortar en la medida de lo posible este tiempo de proceso, el SO ofrece al DEM, la posibilidad de reenviar los paquetes automáticamente sin necesidad de que el DEM tenga que andar examinando paquete por paquete.

Gracias a este soporte del SO, el DEM simplemente tendrá que atender a los mensajes IGMP. Una vez que con IGMP sabe que paquetes provenientes de un determinado origen deben de salir por que interfaz, indica al SO utilizando una llamada al sistema que reenvíe automáticamente ese tipo de paquetes por ese interfaz mientras no se le indique lo contrario. Del mismo modo, cuando todos los hosts de una subred accesible por un determinado VIF dejan de estar interesados en recibir paquetes de un determinado grupo multicast, el DEM comunicará al SO mediante otra llamada al sistema, que deje de reenviar los paquetes correspondientes por ese VIF.

El problema que aparece es que no existe ningún control sobre los emisores y por lo tanto cualquier usuario que tenga accesible un *mrouter* puede emitir el tráfico multicast ya que el *mrouter* simplemente retransmite los paquetes por los VIF's que corresponda. Es precisamente nuestra tarea el establecer un control sobre estos EM.

Sin embargo, esta forma de realizar los reenvíos de paquetes nos obliga a tener que bajar a un nivel inferior al DEM para poder tener cierto control sobre los paquetes multicast. El único control que podemos llevar a cabo a nivel de DEM, es el decidir si vamos a permitir que un determinado tráfico multicast que llegue por un VIF va a ser reenviado o no. En ningún caso podemos diferenciar entre usuarios o equipos. Es decir, o habilitamos a todos a emitir a un determinado grupo o no habilitamos a ninguno. Además, caso de habilitar a alguno, cualquiera de los hosts accesibles desde ese VIF puede emitir con el TTL que les apetezca sin que podamos hacer nada para evitarlo. Este enfoque tiene la ventaja de que es muy general y podría aplicarse a cualquier equipo que haga de DEM independientemente del SO que tenga. Sin embargo, tiene el inconveniente de que es muy poco elástico y prácticamente inservible. Necesitamos pues mejores soluciones que indudablemente van a pasar por equipos basados en SS. OO. cuyo código fuente del

kernel sea totalmente abierto tales como Linux [5] o FreeBSD.

Actualmente la gestión de los EM es un tema totalmente novedoso y por estudiar. De hecho, sólo existe una propuesta (que se ha convertido en Internet-Draft [8]) que describe una serie de extensiones al protocolo IGMPv2 para soportar la autenticación de EM.

3. Solución adoptada

Para resolver el problema descrito, pueden plantearse diferentes soluciones cada una de ellas con sus ventajas y sus inconvenientes. Todas estas soluciones tienen un denominador común que nos viene impuesto por la forma en la que se realiza el reenvío de los paquetes: la necesidad de modificar el soporte que da el SO a los DEM. Básicamente, se trata de controlar cuando un determinado paquete multicast debe de ser reenviado o no. En concreto, un datagrama multicast sólo se reenviará cuando cumpla una serie de condiciones dependientes del usuario o host que lo envía.

3.1. Modificaciones al Kernel del SO

Para poder llevar un cierto control sobre los paquetes multicast que están circulando realmente a través de nuestro *mrouter*, no nos queda otro remedio que trabajar a nivel del kernel y controlar qué paquetes se van a reenviar y cuales no.

Las modificaciones se han realizado sobre un PC con Linux RedHat 5.2 que hacía de *mrouter*. Inicialmente, las modificaciones se hicieron sobre el kernel 2.0.36, sin embargo, dado que este kernel presenta varios fallos en lo referente al enrutamiento multicast, decidimos pasarnos al kernel 2.2.2. Es importante destacar que las modificaciones realizadas en el kernel 2.0.36 no valen para el 2.2.2 ni al contrario. Esto se debe a que a partir del kernel 2.2.0 cambia la forma en la que se copian los datos entre el espacio de datos del usuario y el espacio de datos del kernel.

Nuestra modificación del kernel consiste en la adición de unas estructuras de datos que me permitan recoger los parámetros asociados a cada EM así como mantener un registro de los EM autorizados junto a los grupos a los que emiten. Del mismo modo se incorporan una serie de funciones que me permitan comprobar que realmente el tráfico generado por un EM corresponde a los parámetros asociados a ese usuario y me permitan descartar los paquetes que no se ajusten a los parámetros establecidos. Además se añaden nuevas llamadas al sistema para permitir la reconfiguración dinámica.

Independientemente de la versión del kernel empleada, se deben de definir una serie de estructuras internas de manejo de datos que nos permitan gestionar qué usuarios o hosts pueden enviar a qué grupos y con qué TTL máximo. Para ello se definió una nueva estructura denominada

msender_ctl. Esta estructura no almacena otra cosa sino la información necesaria para controlar el tráfico generado por un emisor multicast (EM).

El siguiente paso era pues, comprobar que realmente el tráfico generado por cada emisor cumplía los parámetros establecidos para ese emisor en su correspondiente estructura *msender_ctl*. Para ello, lo primero que debía definirse es una política de control. La política que hemos implementado consiste en descartar todo el tráfico multicast excepto el que provenga de un origen registrado en el kernel y no viole los parámetros definidos para ese origen en la estructura *msender_ctl* que le corresponde en el kernel.

Con esta política podemos estar tranquilos de que ningún usuario va a poder emitir tráfico a algún grupo multicast al que no le esté permitido. De este modo se evita que unos usuarios puedan emitir a sesiones a las que no tienen permiso y por lo tanto no puedan molestar o estropear esas sesiones de forma malintencionada.

Del mismo modo, nos aseguramos de que tampoco va a poder emitir con un TTL superior a su máximo permitido ya que en tal caso esos paquetes serían descartados. Esto nos permitiría por ejemplo, abrir una sala de libre acceso a Mbone estando tranquilos de que si damos a esas máquinas un TTL máximo menor al Threshold de su primer *mrouter*, nunca podrán sacar tráfico de esa sala. Esto les permitiría recibir sesiones desde fuera pero en ningún caso consumir recursos no locales o molestar a otros usuarios.

Una vez estaba definida la política había que implementarla. Para ello, se añadió una rutina al kernel que comprueba que cada paquete situado en la cola de paquetes IP del kernel cumple con la política anteriormente definida y en caso contrario se descarta.

Si bien, esto es un gran avance desde nuestra situación de partida, no debemos de quedarnos aquí. Aún nos falta algo muy importante por definir. ¿Cómo pasamos la información al kernel de los parámetros que corresponden a cada usuario?.

Para esto nos creamos tres nuevas llamadas al sistema: una para inicializar las estructuras de datos que almacenaban los parámetros para los diferentes emisores, otra para añadir un determinado EM con sus correspondientes parámetros y otra para eliminar un EM. Estas tres nuevas llamadas nos permiten una gran flexibilidad ya que podemos añadir y eliminar de forma totalmente dinámica a un EM sin necesidad de rearrancar el *mrouter*.

3.2. Una primera solución

Una vez construido el soporte básico para controlar el cumplimiento de la política definida, lo siguiente es integrar todo esto en un sistema real y ponerlo a

prueba. Es decir, obtener una primera solución al problema.

En concreto esta primera solución consistió en modificar el código fuente del DEM *mrouted* para que aceptase un fichero de configuración llamado *mauth.conf* que es el que incluiría los usuarios a añadir a las estructuras del kernel junto con sus correspondientes parámetros. Concretamente modificamos la versión 3.9-beta3 y la ejecutamos en el mismo Linux RedHat 5.2 con kernel 2.2.2 anteriormente mencionado.

Cuando este DEM arrancaba, leía los datos del fichero de configuración, utilizaba las nuevas llamadas al sistema para añadir EM al kernel y posteriormente comenzaba a enrutar multicast del modo usual. Lógicamente al terminar el demonio volvía a eliminar del kernel los EM añadidos. De este modo, actualizar la lista de emisores era tan simple como hacer un *kill -HUP* el DEM tras haber actualizado el fichero *mauth.conf*.

La ventaja que aporta esta solución es muy clara: simplifica mucho el proceso de gestión. Se trata simplemente de editar un fichero de configuración y rearrancar el DEM.

También es cierto que también tiene unos inconvenientes que se pueden resumir en que no aprovecha las ventajas que nos aporta el haber realizado las modificaciones en el núcleo del SO.

3.3. Una solución mejorada

Los inconvenientes que presenta la solución anterior se deben a la ligadura que establecemos entre el DEM y el proceso de definir los parámetros y los usuarios. Para evitar estos problemas planteamos un enfoque distribuido. Es decir, separamos la gestión de los EM de lo que es el funcionamiento del DEM. De este modo se consigue una infraestructura independiente del DEM.

Esta independencia se consigue gracias a las ventajas que aportan las llamadas al sistema que hemos creado:

- Nuestras llamadas al sistema nos permiten insertar y eliminar EM sin necesidad de rearrancar nada y aquí se nos obliga a rearrancar el DEM para cada actualización.
- Nuestras llamadas al sistema nos proporcionan independencia del DEM que se emplee y nos evitan el ir modificando el código fuente del algoritmo de enrutamiento que corresponda.

Para conseguir esta infraestructura, vamos a crearnos nuestro propio demonio de gestión de usuarios frente al kernel al que denominaremos en adelante *musersd* (Multicast users daemon).

La función del *musersd* va a permanecer indefinidamente escuchando los comandos que le

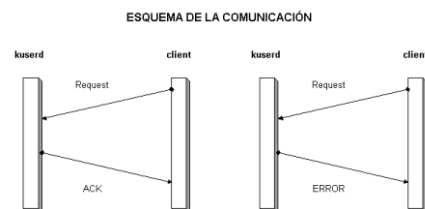
lleguen dirigidos a un determinado puerto UDP de la máquina y entonces procederá a ejecutarlos. Lógicamente, los comandos que puede recibir son añadir un EM junto con sus parámetros al kernel o eliminar un EM del kernel. No hemos añadido el comando de inicialización de estructuras del kernel porque ese lo llevará a cabo automáticamente *musersd* al arrancar.

En el otro extremo del socket establecido por el *musersd*, se situará un applet que hará de interfaz entre el administrador y el kernel del *mrouter* y que servirá para autenticar frente al kernel a un determinado emisor. A este applet le hemos llamado *mauthclient*.

Lógicamente el primer problema que se plantea es la definición de un protocolo de comunicación entre el *musersd* y *mauthclient*. Para ello definimos un protocolo del tipo *call/response* en el que el *musersd* hará de servidor y *mauthclient* será el cliente. El esquema de comunicación es el que se muestra en la Fig. 1.

El *mauthclient* enviará los comandos anteriormente mencionados y el *musersd* responderá con un mensaje de ACK o ERROR según si la ejecución de ese comando fue correcta o no.

Para evitar comportamientos anómalos del protocolo y, dado que se está trabajando sobre UDP, se tuvieron que definir una serie de campos extras en cada mensaje para permitir el control de posibles pérdidas paquetes, pérdidas de los propios



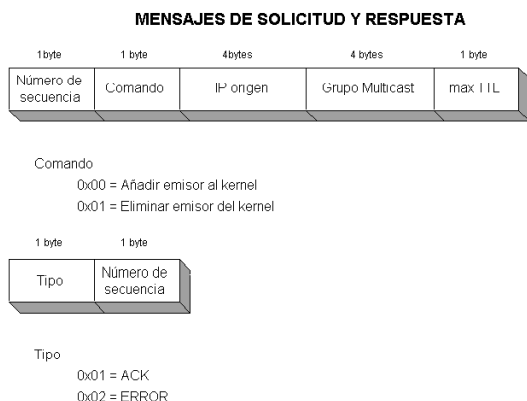
asentimientos así como establecer temporizadores para la gestión de los timeouts. En concreto, para evitar los problemas que acarrear las pérdidas de paquetes se emplea por un lado un *número de secuencia* asociado a cada mensaje para poder detectar la falta de algún paquete intermedio y por otro lado se emplean temporizadores para reenviar los mensajes en caso de no recibir respuesta en un determinado lapso de tiempo.

Además, debido a la existencia de posibles retransmisiones debe de implementarse en la parte del *musersd* algún mecanismo que asegure que se cumple la semántica de llamada *at most once* similar a la de los Remote Procedure Call (RPC). Es decir, se trata de que aunque lleguen varias solicitudes de un mismo comando, en realidad sólo se ejecute una vez en el servidor para evitar posibles efectos laterales. Para evitar este efecto indeseado, se ha recurrido a la típica ventana deslizante que en nuestro caso va a ser de tamaño uno. Es decir, el

usersd memoriza la última solicitud que recibió así como la respuesta que se obtuvo. De este modo se puede responder directamente a esa solicitud sin necesidad de ejecutar el comando más de una vez. El formato de los mensajes que se intercambian aparece reflejado en la Fig. 2.

Las ventajas que incorpora esta solución son las siguientes:

- Independencia total del DEM.
- Gestión totalmente distribuida basada en Web con la consiguiente comodidad y facilidad de uso que supone para el administrador. Del mismo modo, al ser la aplicación de gestión un applet nos asegura la suficiente independencia de la plataforma.
- Sólo se impone que la máquina que hace de *mrouter* use un SO abierto. El resto de las máquinas que participen en sesiones multicast e incluso la que usa el administrador pueden ser de cualquier plataforma.



Lógicamente, este enfoque también tiene un inconveniente. El manejo puede resultar poco intuitivo. Además, el administrador tendría que estar atento a los grupos multicast que existen actualmente empleando alguna de las herramientas típicas de Mbone como SDR o MAnnouncer y a partir de esa información añadir las autorizaciones al kernel del demonio de enrutamiento.

3.4. Solución definitiva

Para facilitar la gestión hemos optado por crear una herramienta global de control de EM que incorpore la posibilidad de que el administrador desde una sola pantalla sea capaz de controlar las sesiones que van creándose o eliminándose, así como los EM autorizados a participar en cada una de ellas junto con los parámetros de control asociados.

Para conseguir nuestro objetivo introducimos un nuevo elemento que llamaremos *mauthserver*. El *mauthserver* se va a colocar como un elemento intermedio entre el *usersd* y el *mauthclient* y va a tener varias funciones:

- Escuchar anuncios de sesiones multicast. El

Figura 2: Formato de los mensajes

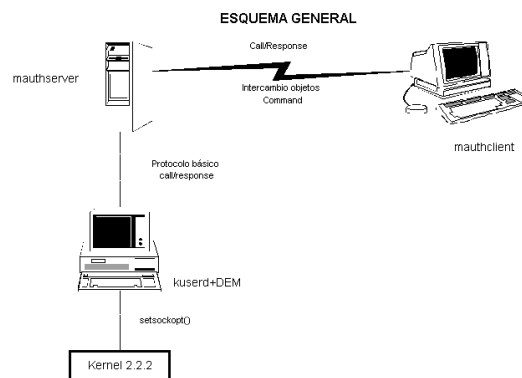


Figura 3: Esquema de la arquitectura

mauthserver va a implementar los protocolos SAP [6] (Session Announcement Protocol) y SDP [7] (Session Description Protocol). De este modo, capturarán todos los anuncios de sesiones que se vayan produciendo y los enviará al *mauthclient* para que éste pueda presentarlos al administrador.

- Dar una cierta tolerancia a fallos (TF) al sistema. La posición estratégica del *mauthserver* le permite controlar todo el flujo de información entre el *mrouter* y el *usersd*. De este modo el *mauthserver* va a ser el elemento homogeneizador que va a permitir la recuperación satisfactoria del sistema ante la caída de cualquiera de sus extremos. Para ello, el *mauthserver* mantendrá la información tanto de las sesiones como de los usuarios autorizados en disco. Ante cualquier caída, a partir de los datos del disco se puede restablecer la situación normal de funcionamiento.

El esquema general de esta solución aparece en la Fig. 3. Como vemos, esta figura plantea la idea de un enfoque totalmente descentralizado en el que aparecen una serie de entidades que se comunican pero sin establecer en ningún caso restricciones sobre la localización de ninguna de esas entidades. También se puede apreciar como aparecen una serie de esquemas de comunicación entre las diferentes entidades de la arquitectura que se deben de definir.

3.5. Comunicación ente el *usersd* y el *mauthserver*

Para la comunicación entre estas dos entidades se emplea el mismo mecanismo *call/response* que empleaba el *mauthclient* de la solución anterior para comunicarse con el *usersd*. El *mauthserver* enviará solicitudes que a su vez recibe del *mauthclient*, y recibirá respuestas del *usersd*. En función de estas respuestas se actualizará el nuevo estado del sistema en disco y se devolverá la respuesta al *mauthclient*.

El protocolo y los formatos de los mensajes siguen siendo los mismos que los del esquema de comunicación anterior.

3.6. Comunicación entre el *mauthserver* y el *mauthclient*

Para comunicar al *mauthserver* y al *mauthclient*, se van a necesitar un par de esquemas de comunicación: por un lado necesitamos que el *mauthserver* comunique al *mauthclient* para indicarle la creación o eliminación de una nueva sesión. Por otro lado, necesitamos que el *mauthserver* acepte las solicitudes que haga el administrador desde el *mauthclient* para registrar o eliminar EM y las transmita al *musersd* para que se hagan efectivas.

Para lo referente a la aceptación de las solicitudes que haga el administrador, la solución es reutilizar el mismo protocolo que definimos anteriormente. Para que el *mauthserver* informe del estado de las sesiones, vamos a establecer un esquema un tanto diferente.

Dado que tanto el *mauthserver* como el *mauthclient* se encuentran implementados en Java, vamos a aprovechar las facilidades de serialización de objetos que proporciona este lenguaje. Cada vez que el *mauthserver* descubra el anuncio de una nueva sesión, va a pasar a través de un socket al *mauthclient* un objeto de la clase *Comando*. Básicamente un objeto de la clase *Comando* contendrá un atributo *tipo* (que identificará si se trata de añadir una nueva sesión, eliminar alguna y existente...) y un atributo de la clase *Session*. En este atributo es precisamente donde se almacena toda la información referente a la sesión (los grupos multicast, los tipos de codificación, título, EM autorizados, etc.). De esta forma el *mauthclient*, puede estar permanentemente informado de las sesiones que se vayan creando y eliminando.

Las ventajas que tiene este enfoque tienen que ver sobre todo con:

- Ofrece una solución realmente operativa y manejable. De hecho, incorporamos una herramienta integral de gestión de EM que permite que un administrador pueda establecer de un modo sencillo, dinámico y vía WWW los parámetros que se van a asignar a cada uno de sus EM. De este modo podrá controlar totalmente el tráfico multicast que sale de su red evitando así todos los problemas que se plantearon en la introducción.
- Propone un enfoque totalmente flexible y distribuido a la vez que fácil de integrar y utilizar.
- El empleo del lenguaje Java para la realización de todos los elementos del sistema excepto para el *musersd*, nos aporta un independencia total de la plataforma que nos ayuda aún más a conseguir una mayor flexibilidad. De hecho el único elemento del sistema que no es multiplataforma es el *mrouter* al que sólo se le impone que el SO ofrezca los fuentes del kernel.

- Facilita la tarea del administrador ya que el *mauthclient* incorpora una interfaz totalmente amigable, intuitiva y sencilla de utilizar. La interfaz puede apreciarse en la Fig. 4.
- Por supuesto, sigue siendo independiente del DEM a utilizar. De este modo, es una solución fácilmente integrable en multitud de entornos y situaciones.

En cuanto a los inconvenientes, destacar la necesidad de la figura del administrador que establezca los parámetros asociados a cada EM. Como ya se verá en las vías futuras, entra dentro de nuestros planteamientos tratar de evitar la presencia de un administrador y automatizar en la medida de lo posible el proceso.

4. Conclusiones y vías futuras

La tecnología de IP Multicast, junto con su implementación en el Mbone, proporciona un esquema idóneo para la realización de videoconferencias sobre Internet, pero no sólo para videoconferencias, sino para la utilización de cualquier servicio que implique la comunicación simultánea entre más de dos equipos de la red.

Sin embargo, se trata aún de un esquema de comunicación en fase de maduración. Aunque se está haciendo un gran esfuerzo para el desarrollo y perfeccionamiento de los algoritmos de enrutamiento multicast, se están dejando de lado cuestiones como el control de usuarios, sistemas de monitorización y depuración de errores en la red multicast, privacidad de la información...

Como hemos podido ver a lo largo del documento, se puede utilizar varias alternativas para solucionar el problema, desde una solución a nivel de aplicación que consistiría en no dejar arrancar las aplicaciones a menos que el usuario se haya autenticado hasta una solución a nivel de Sistema Operativo totalmente integrada dentro del nivel IGMP.

Nosotros hemos optado por realizar una solución intermedia entre las dos, básicamente para poder probar primero la operatividad del sistema antes de implementar una solución más compleja.

Así, la solución que hemos implementado es

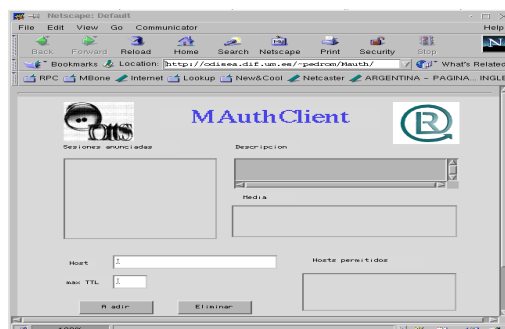


Figura 4: Applet de administración

totalmente operativa para los propósitos definidos, es decir, para poder controlar quién se conecta a MBone, quién transmite y con que ámbito lo hace, ya que cada usuario tiene definido un ttl máximo de transmisión. Y todo esto se realiza de un modo transparente para el usuario final.

Además, no es necesario realizar grandes cambios a nivel del DEM. De hecho, no hay que modificar propiamente este software (mrouted), sino que se añade funcionalidad al sistema operativo que lo ejecuta. Pero los cambios necesarios son mínimos y se limitan a añadir unas pocas llamadas.

No obstante, aunque se trata de un sistema totalmente operativo, esto no significa que no pueda ser objeto de nuevas mejoras, algunas de las cuales ya se está intentando introducir. Algunas mejoras que hay que introducir son:

- Puesto que es necesaria una comunicación entre distintos equipos de una red, es necesario establecer algún mecanismo para identificar los equipos reales involucrados en las comunicaciones y que nadie pueda suplantar el papel del otro. En este sentido, se están considerando varias posibilidades como pueden ser la utilización de SSL o bien la firma digital mediante PGP.
- Actualmente, la autorización a un usuario se hace mediante la figura del administrador, el cual tiene que definir qué usuarios pueden enviar información y con qué ttl. Sin embargo, se puede integrar todo el sistema dentro de un esquema de RADIUS para liberar de estas tareas al administrador.
- El sistema trata fundamentalmente de prohibir que los usuarios puedan saturar los enlaces mediante el envío masivo de información por grupos multicast. Sin embargo, no trata nada sobre la recepción de información. En un siguiente paso, junto con la integración con IGMP, se puede realizar también autenticación de usuario a nivel de recepción, lo cual implica también algún esquema de cifrado de la información.

Agradecimientos

Para finalizar el documento, nos gustaría agradecer su colaboración a todas las personas que nos han ayudado en la búsqueda de la solución e implementación de la misma.

Así nos gustaría agradecer Norihiro Ishikawa del NTT Information and Communication Systems Laboratories su ayuda al resolvernos algunas cuestiones relativas al draft [8] que ha elaborado su grupo de trabajo, así como por dejarnos una copia de la implementación experimental de su sistema sobre FreeBSD.

Además, nos gustaría extender el agradecimiento a Francisco Cruz, Jesús González Barahona y su

grupo de trabajo dentro de la Universidad Carlos III de Madrid, por la colaboración y el interés mostrado por buscar una solución dentro de las reuniones que se celebraron dentro del grupo de trabajo IRIS-MBone.

Finalmente, agradecemos a la Comisión Interministerial de Ciencia y Tecnología CICYT por soportar parcialmente este trabajo a través del proyecto TEL97-2001-E.

Referencias

- [1] Vinay Kumar. "MBone: Interactive Multimedia on the Internet". New Riders, 1996. ISBN 1-6205-397-3.
- [2] Steeve Deering. "IP Multicast Extensions for 4.3BSD UNIX and related systems". Junio 1989. Stanford University.
- [3] D. Waitzman, C. Partridge, S. Deering. "Distance Vector Multicast Routing Protocol". RFC 075.
- [4] W. Fenner. "Internet Group Management Protocol, Version 2". RFC 2236. Noviembre 1997.
- [5] David A. Rusling. "The Linux Kernel". REVIEW, Version 0.8-2. Marzo 1998
- [6] Mark Handley. "SAP: Session Announcement Protocol". INTERNET-DRAFT. Noviembre 1996.
- [7] M. Handley, V. Jacobson. "SDP: Session Description Protocol". RFC 2327. Abril 1998
- [8] N. Ishikawa, N. Yamanouchi, O. Takahashi. "IGMP Extensions for Authentication of IP Multicast Senders and Receivers". INTERNET-DRAFT. Agosto 1998.

Capítulo 2B: Plataformas Distribuidas y Gestión de Red

Modelo distribuido para la construcción de servicios de red multimedia programable.
J.A. Lozano, M.C. Oria, F. Sandoval (U. Málaga)

Arquitectura Composicional de Seguridad para Servicios Telemáticos.
L. Fuentes, J. López (U. Málaga)

Una experiencia de vídeo bajo demanda soportada en tecnología web, CORBA y SDL.
R. de Diego, J. Carracedo, C. Ramos, V. Hernández (U. Politécnica de Madrid)

El papel de los servicios de intermediación en el contexto del comercio electrónico.
V. Villagra, J.I. Moreno, J.I. Asensio, J. Berrocal (U. Politécnica de Madrid)

Mecanismos de comunicación y gestión de servicio de un broker de información multiagente.
F. Valera, J.I. Moreno, V. Villagrà, J. Berrocal (U. Carlos III de Madrid)

Sistema de Gestión de Servicios de Telecomunicación en un Entorno Cambiante.
R. Koehn, C. Guerrero, L. Bordon y G. Fernández (U. La Coruña)

Modelo distribuido para la construcción de servicios de red multimedia gestionables

J. A. Lozano, M. C. Oria y F. Sandoval
Departamento de Tecnología Electrónica, E.T.S.I. de Telecomunicación
Universidad de Málaga, Campus de Teatinos, 29071-Málaga
Telf: 95-2131424, Fax: 95-2132733
E-mail: jal@dte.uma.es

Abstract

Multimedia services are becoming an attractive formula for organizations as companies, universities, hospitals, etc. with their corporate networks (intranet), since these new services help them to return network investment. This paper addresses two problems that the multimedia applications developers have to faced with. On one hand, the complexity of building applications for heterogeneous and distributed environment, such as corporate networks are; on the other hand, the applications must be able to request to the network some requirements of quality, in order to guarantee the information that multimedia applications generate is correctly delivered. A generic object oriented model is proposed in order to help developers to solve these problems. The model has a set of classes to hide the networking technologies issues and to build manageable multimedia services

1. Introducción

Cada vez son más numerosas las organizaciones como empresas, universidades, hospitales, etc. que disponen de una red de datos propia. Estas redes corporativas o *intranets*, como se han denominado, estaban diseñadas para transportar datos entre las distintas localizaciones de la organización. El desarrollo de nuevos servicios multimedia está cambiando el uso al que estaban destinadas estas redes, ya que estos servicios permiten, por ejemplo, mantener videoconferencias entre miembros situados en varias localizaciones físicas diferentes, transmitir un seminario a varios puntos, reproducir un vídeo de una clase o sesión médica almacenado en otro lugar, etc.

Estos servicios se caracterizan por ser aplicaciones que se ejecutan en un ordenador o estación de trabajo que acceden y/o intercambian información con otras aplicaciones o sistemas de almacenamiento remotos a través de una red de comunicaciones. Las entidades que soportan los servicios multimedia están distribuidos por toda la red y a la vez pueden ser compartidos por otros servicios/usuarios.

Las redes corporativas son, generalmente, entornos heterogéneos donde pueden existir, distribuidos por toda la red, servidores y estaciones de trabajo UNIX junto con ordenadores personales con diferentes sistemas operativos como Windows, Mac o LINUX. Incluso las tecnologías de transporte utilizadas para las redes pueden ser distintas: Ethernet, ATM, FDDI, etc.

La creación de nuevos servicios, así como el control y la gestión de los mismos, en estos entornos puede resultar muy compleja ya que hay que considerar las particularidades de cada tecnología. Esto hace que el código que se genera sea poco flexible y deba ser modificado cada vez que se produce un cambio en la

red. Esta situación se agrava ya que las herramientas de ayuda al desarrollo de aplicaciones suelen estar diseñadas para entornos homogéneos.

La aparición de plataformas para el desarrollo de aplicaciones software distribuidas como CORBA (*Common Object Request Broker Architecture*) [1] y DCOM (*Distributed Common Object Model*) [2], ayuda en parte a solucionar este problema. Estas plataformas ofrecen una serie de servicios que pueden utilizar las aplicaciones y que facilitan la comunicación entre los objetos que componen la aplicación, con independencia de su localización e implementación.

Otro aspecto importante a destacar en estos servicios es el tipo de información que manejan ya que mezclan simultáneamente información tradicional (texto, imágenes, etc.) con información de audio y vídeo (multimedia), en muchos casos en tiempo real y con un volumen elevado de datos. La transmisión de información de varios tipos exige que la red disponga de mecanismos adecuados para diferenciarlos. Surge así el concepto de calidad de servicio (QoS *Quality of Service*) como una serie de parámetros cuantificables y que se especifican de forma independiente a como se representan o controlan [5].

Por ejemplo, en la Fig.1 se puede ver que para ofrecer un servicio de videoconferencia típico entre dos ordenadores ambos intercambian información de audio, vídeo y control. Para cada tipo de señal existen unos parámetros (por ejemplo el retardo, las pérdidas, etc.) que hay que garantizar. El valor de estos parámetros puede cambiar incluso para el

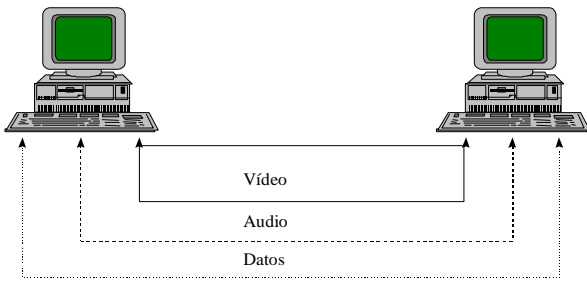


Figura 1.- Ejemplo de servicio multimedia

mismo servicio pues el usuario puede elegir la calidad con la que quiere recibir la información de audio y/o vídeo.

En este artículo se propone un modelo de objetos genérico para ayudar al desarrollo de servicios multimedia distribuidos. El modelo permite la creación, operación, gestión y programación de los servicios multimedia. Para ello, aumenta el nivel de abstracción necesario para desarrollar aplicaciones multimedia encapsulando aspectos tecnológicos específicos y reduciendo, por tanto, el conocimiento técnico requerido a los programadores. Permite, también, que las aplicaciones puedan indicar a la red los valores de QoS requeridos para el intercambio de información.

El resto de este artículo se va estructurar de la siguiente forma: en el apartado 2 se introduce CORBA como plataforma para la construcción de aplicaciones distribuidas y se especifican los parámetros que se van a utilizar para indicar la QoS, en el apartado 3 se va a presentar el modelo de objetos, en el apartado 4 se indica cómo se aplica este modelo para en un servicio real, y por último, el apartado 5 se presentan las conclusiones de este trabajo y las líneas futuras de investigación.

2. Los nuevos servicios multimedia

Para el desarrollo de servicios multimedia hay que resolver, entre otros, dos aspectos fundamentales: la utilización de recursos distribuidos por toda la red y la necesidad de disponer de servicios de red diferenciados sobre los que transportar los diferentes tipos de información.

Ambos aspectos plantean a los diseñadores de aplicaciones multimedia dos problemas nuevos que es necesario resolver: desarrollar aplicaciones distribuidas y encontrar la forma de indicarle a la red el nivel de calidad exigido para el transporte de la información. Para resolver el primer problema están apareciendo plataformas de desarrollo sobre las que construir aplicaciones distribuidas, y para resolver el segundo problema se están definiendo arquitecturas enfocadas al control y gestión de la QoS, como la que se presenta en [3], y que definen un modelo de servicios para clasificar el grado de calidad requerido por las aplicaciones.

2.1. Aplicaciones distribuidas

En un intento de solucionar los problemas asociados a la programación distribuida en 1989 se constituyó el consorcio OMG (*Object Management Group*), con el objetivo de desarrollar, adoptar y promover estándares para el desarrollo de aplicaciones en entornos distribuidos heterogéneos. OMG ha desarrollado un modelo conceptual, conocido como el modelo de objetos básico, y una arquitectura de referencia denominada OMA (*Object Management Architecture*), sobre la que se construyen las aplicaciones. Esta arquitectura define las facilidades necesarias para el procesamiento distribuido de aplicaciones orientadas a objetos, con un elevado grado de abstracción.

En esta arquitectura existe un componente, el ORB (*Object Request Broker*), que es un bus lógico por el cual se comunican los objetos que componen una aplicación. Alrededor de este bus se agrupan una serie de clases y servicios como se ve en la Fig. 2. Existe un conjunto de clases genéricas que agrupan funciones que suelen ser comunes a la mayoría de las aplicaciones, son las facilidades comunes. Las interfaces del dominio modelan entidades que son comunes en cierto ámbito de aplicación (medicina, telecomunicaciones, etc.). Además de estos dos grupos, están las interfaces específicas de la aplicación. Asociado al bus, también existen una serie de servicios genéricos para el manejo y control de los objetos.

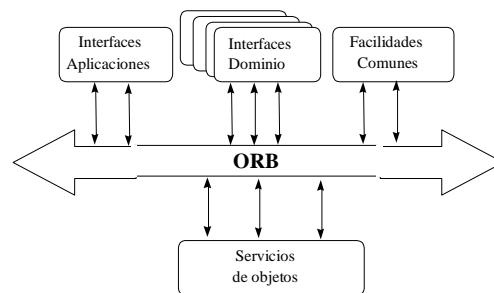


Figura 2.- Arquitectura OMA

El ORB se encarga de transportar las peticiones que hacen los clientes a los servidores y de devolver las respuestas que se generan en respuesta a las peticiones. La característica principal del ORB es la transparencia con la que se realizan las transacciones frente a:

- la localización de los objetos sobre los que se invocan los métodos, ya que estas pueden estar en distintos nodos,
- el lenguaje de programación con el que se han construido los objetos, pues permite invocar métodos sobre objetos codificados con diferentes lenguajes como C, C++, Java, etc.,

- el estado de ejecución del objeto destino, ya que el ORB se encarga de activarlo en caso necesario,
- los mecanismos de comunicaciones utilizados para llevar las peticiones y respuestas.

Una de las primeras especificaciones que adoptó el consorcio fue la de CORBA, en la que se detallan y especifican las características del ORB de la arquitectura OMA. CORBA constituye una plataforma sobre la que es posible desarrollar aplicaciones distribuidas. Ofrece una serie de servicios que pueden utilizar las aplicaciones y que facilitan la comunicación entre los objetos que componen una aplicación, con independencia de donde se encuentren localizados físicamente.

Los objetos se especifican en CORBA a través de su interfaz que se codifica según un lenguaje de especificación de interfaces, IDL (*Interface Description Language*). IDL declara los métodos que se pueden invocar sobre el objeto, y es similar a la declaración de clases en C++. La implementación de la clase se puede realizar en cualquier lenguaje de programación y es independiente de la declaración de la interfaz.

Los componentes principales de CORBA se pueden ver en la Fig. 3, y son:

- El ORB, que es el responsable de transportar las peticiones entre los objetos y llevar la respuesta al objeto que hizo la petición. A partir de CORBA 2.0 permite la comunicación con objetos en el dominio de otro ORB a través de los protocolos GIOP (*General Inter-ORB Protocol*) e IIOP (*Internet Inter-ORB Protocol*), ambos sobre TCP (*Transport Control Protocol*).
- Interfaces ORB. Ofrecen operaciones estándar que desacoplan las aplicaciones de los detalles de la implementación de un ORB.
- *Stubs* y *Skeleton* IDL. Se generan automáticamente a partir de la descripción IDL de las interfaces e implementan clases que ofrecen los servicios de comunicaciones entre los objetos. Sirven de unión entre los clientes y los servidores.
- Interfaz de llamadas dinámicas (DII) Permite al cliente acceder directamente a los mecanismos de transporte ofrecidos por el ORB. Se utiliza cuando una aplicación no se tiene conocimiento, en tiempo de compilación, de los métodos de la interfaz a la que está accediendo.
- Interfaz de *Skeleton* dinámicas (DSI). Es el equivalente de la DII para el servidor.
- Adaptador de objetos. Sus funciones principales son: asociar un cliente a un ORB, demultiplexar las peticiones que llegan al servidor e indicar la operación apropiada que tiene que ejecutar el servidor.

Aunque en la Fig. 3 aparece una arquitectura cliente/servidor, en CORBA, estos roles sólo tienen sentido en el ámbito de una transacción. El cliente en una operación puede ser el servidor en otra operación

2.2. Especificación de la QoS

La arquitectura de las redes corporativas está basada, al igual que Internet, en el protocolo IP (*Internet Protocol*). Estas redes sólo ofrecen un servicio de transporte *best effort*. Este servicio asegura que la red transportará los datos de forma fiable, pero no garantiza ningún otro parámetro de la transmisión. Cada paquete generado por una aplicación es tratado por la red de forma independiente y única. No existe posibilidad de indicar que ciertos paquetes pertenecen al mismo intercambio de información y por lo tanto están relacionados.

Las aplicaciones que se construyen sobre este tipo de redes están diseñadas de forma elástica, es decir, son insensibles al retardo. Las aplicaciones esperan a disponer localmente de todos los datos necesarios para ejecutarse. Así, por ejemplo, para reproducir una canción se obtienen los datos del servidor, se almacenan en memoria y se reproducen. Incluso, mientras se está transportando la información desde el servidor se pueden producir paros en la transmisión.

En los nuevos servicios multimedia, la información puede fluir de forma continua entre dos aplicaciones, por tanto, debe existir un mecanismo para poder indicarle a la red que ciertos paquetes forman parte de un mismo servicio y que deben recibir el mismo tratamiento (encaminamiento, prioridad, etc.). Surge así el concepto de flujo para agrupar la información que viaja por la red, y que en [3] se define para caracterizar la generación, transmisión y consumo de datos asociados con un tipo de información y con vistas a la gestión y control de la QoS.

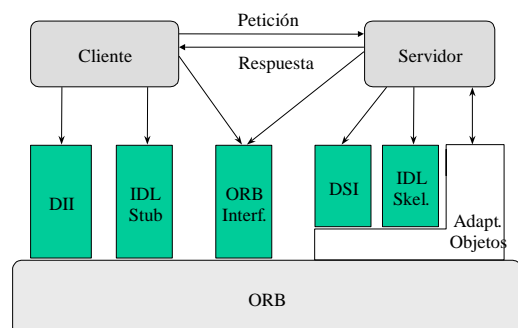


Figura 3.- Componentes de CORBA

El concepto de flujo permite separar individualmente cada uno de los tipos de información que genera una aplicación y asociar a cada flujo una cierta QoS. Es necesario que las aplicaciones y la red dispongan de una visión común

de los aspectos de calidad. Esta visión común es el modelo de servicios que especifica los parámetros que se van a utilizar para medir la calidad y permite clasificar los flujos de información según estos parámetros.

Un modelo de servicio establece una serie de clases de servicio que sirven para clasificar, en función de sus parámetros de calidad, los flujos de información. El IETF ha definido un modelo de servicios integrados definido para la futura expansión de las capacidades de Internet que contempla los parámetros de retardo y *jitter* de la conexión. Este modelo presenta tres clases de servicio: garantizado, estadístico y elástico [4]. La clase de servicio garantizado establece unas garantías absolutas en los parámetros, el estadístico asegura ciertos valores estadísticos de los parámetros como valor medio, valor de pico, etc., y por último, la clase de servicio elástica no ofrece ningún tipo de garantía sobre ningún parámetro.

Este modelo sólo tiene en cuenta los aspectos temporales de la transmisión de los paquetes. Sin embargo, si se quiere realizar una gestión integrada de los servicios y los recursos compartidos de la red, hay que tener en cuenta otros aspectos. Estos aspectos incluyen datos sobre volumen de información (tamaño de trama, de ráfaga, velocidad de transmisión, etc), calidad de la transmisión (pérdidas, errores), e incluso, aspectos como la sincronización entre flujos. El modelo de servicios, también debe incluir aspectos de gestión como la posibilidad de monitorización, alarmas, etc. si se quiere poder ejercer cierto control y gestión sobre los servicios.

Las arquitecturas para la prestación de servicios multimedia suelen estar organizadas en niveles (aplicación, transporte, red, sistema, etc.), y los aspectos de QoS se localizan en todos los niveles, aunque en cada nivel adopten una visión diferente. Por tanto, la gestión de la calidad supone desarrollar arquitecturas que coordinen la visión de los aspectos de calidad de cada nivel. Una arquitectura de este tipo es la QoS-A presentada en [3]. En [6] puede verse una arquitectura para garantizar la QoS en redes IP mediante el protocolo de reserva de recursos RSVP [8].

En la Tabla I, se indican los parámetros que se van a utilizar para especificar la QoS de los flujos de información. Se ha incluido el mínimo número de parámetros, que tiene en cuenta todos los aspectos de la transmisión sin complicar excesivamente el diseño.

La primera columna indica los aspectos relacionados con el tiempo necesario para transportar la información entre los distintos nodos. Para cada parámetro se pueden utilizar valores absolutos (valor máximo de retardo o *jitter*) y/o valores estadísticos (valor medio). Por ejemplo, se puede especificar el

retardo medio para el transporte de los paquetes que forman el flujo y el máximo retardo que puede sufrir un paquete individual. De esta forma se le indica a la red el retardo de una agrupación de información, por ejemplo una trama, y el retardo máximo de cada paquete dentro de la trama.

Los parámetros de volumen indican la cantidad de datos que va a generar la aplicación. Permiten calcular los recursos de red (ancho de banda, espacio en las colas, etc.) que se van a necesitar para transportar la información y son utilizados por los algoritmos de CaC (*Call Admission Control*) y las funciones de vigilancia UPC (*Use Policy Control*) que se implementen en los nodos.

La fiabilidad indica la sensibilidad de la aplicación a la pérdida de datos. La transmisión de información de control no admite errores, en cambio, la reconstrucción de una señal de vídeo puede tolerar algunos errores en los datos.

En cuanto a los otros parámetros, el control indica el grado de monitorización que es necesario aplicar al flujo de información (continua, paso de umbrales, etc.), y el tipo de información indica aspectos de codificación de la señal y sincronismo en flujos individuales o entre flujos. La sincronización puede ser para un flujo de información individual, o entre varios flujos. Una arquitectura para servicios multimedia que tiene en cuenta todos los aspectos de sincronismo se puede ver en [7].

3. El modelo de objetos

El modelo que se propone se localiza entre las aplicaciones y los recursos de red que se utilizan para soportar los servicios multimedia, como se puede ver en la Fig. 4. Los programadores de las aplicaciones multimedia, sólo necesitan conocer los métodos que les ofrecen las clases del modelo. Cualquier conocimiento específico de las tecnologías y protocolos subyacentes queda abstraído por el modelo. Los recursos pueden ser físicos o lógicos.

La funcionalidad que ofrece el modelo puede ser utilizada tanto por las aplicaciones, como por sistemas de gestión. Por lo tanto, cuando existan varias entidades que puedan acceder a los métodos del objeto se debe habilitar un mecanismo de control.

Tabla I.- Aspectos de QoS

Aspectos	Parámetros
Temporales	Retardo Jitter
Volumen	Velocidad media

	Velocidad de pico Tamaño de ráfaga
Fiabilidad	Pérdidas
Control	Monitorización
Tipo de información	Codificación Sincronización

Conceptualmente, el modelo se basa en la creación de flujos de información, por parte de las aplicaciones, y la asociación de estos flujos a conexiones de red. Las clases se clasifican en dos grupos: clases de transporte y clases de control. Las clases de transporte son las que intervienen directamente en el transporte de los datos y modelan la funcionalidad de las distintas entidades de red. Las clases de control sirven para el control y la gestión de los servicios que se construyan sobre el modelo.

Los objetos de transporte se localizan en los nodos que intervienen en el servicio y los flujos de información fluyen a través suyo, en cambio, los objetos de control se localizan en el nodo que controla el servicio. Si el sistema es centralizado, existe un único nodo de control para todas las instancias de servicio. Si es distribuido, el control puede realizarlo cualquiera de los nodos finales que participan en el servicio (normalmente, el sistema que inicia el servicio). La comunicación entre los objetos se hace utilizando los servicios que proporciona CORBA.

La gestión de los servicios multimedia es importante [11], puesto que facilita aspectos como la provisión de los servicios, la monitorización, el mantenimiento, la compartición de recursos, etc. Especialmente, cuando se prestan sobre entornos heterogéneos, como puede ser la Intranet de una empresa o universidad, etc.

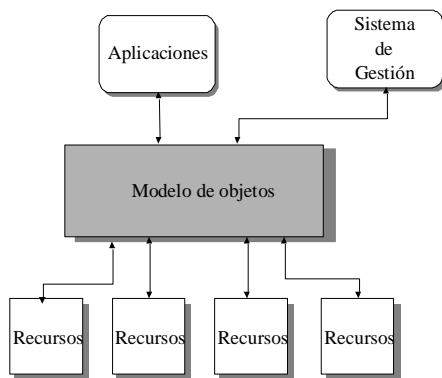


Figura 4.- Localización del modelo

Las redes que soporten los nuevos servicios multimedia deben poder ofrecer varios tipos de servicios de transporte, con diferentes grados de QoS, y deben poder manejar grandes cantidades de información. En este sentido, la UIT-T ha seleccionado la tecnología ATM como base de la futura red de servicios integrados de banda ancha

(RDSI-BA). Por su parte, el IETF está desarrollando nuevos protocolos que puedan ser incluidos en la arquitectura actual de Internet, basada en el protocolo IP. Estos protocolos permiten reservar recursos (espacio en las colas, ancho de banda, etc.) en la red (RSVP *Resource Reservation Protocol*) e incluir información de control de temporización que puedan utilizar las aplicaciones (RTP/RTCP *Real Time Protocol/Real Time Control Protocol* [9]).

En las redes corporativas actuales pueden convivir ambas tecnologías y por lo tanto un modelo genérico debe ser capaz de abstraer y ocultar las peculiaridades de cada una de ellas, de forma que el diseñador de servicios no tenga que considerarlos en sus desarrollos.

Para organizar las funciones de gestión se han propuesto varias arquitecturas como por ejemplo, el modelo TMN de la UIT-T [10], para la gestión de redes y servicios de telecomunicaciones. Esta arquitectura introduce una serie de niveles superpuestos, cada uno con un nivel de abstracción creciente (nivel de elemento de red, nivel de red, nivel de servicio y nivel de negocio). El modelo que se propone permite su inclusión en una arquitectura de gestión de este tipo, se localiza en el nivel de gestión de red y haría de interfaz para el nivel de gestión del servicio.

En la Fig. 5 se puede ver el árbol de contención del modelo de clases. Los círculos al final de las líneas indican una cardinalidad "1 a n" los vacíos y "1 a 1" los rellenos. Los objetos de transporte, que aparecen sombreados, presentan dos interfaces, una de transporte y una gestión. Las demás clases del modelo tienen propósitos de gestión y presentan sólo interfaces de gestión.

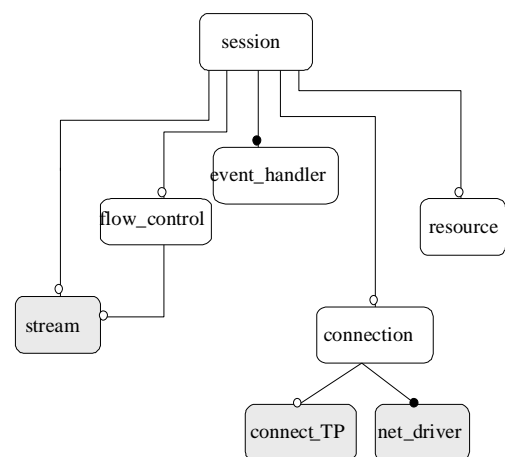


Figura 5.- Árbol de contención

Cuando una aplicación necesita transmitir información por la red genera tantos objetos *stream* como necesite. Esta clase implementa el concepto de flujo individual y tiene asociada información sobre la localización (cola, fichero, etc.) donde se encuentra la información. Se encarga también del control de la temporización de cada flujo individual.

A continuación, la aplicación inicia una sesión, es decir, crea una instancia de la clase *session* donde registra los flujos de datos e indica la QoS de cada flujo. Esta instancia se encarga de contactar con los nodos a los que va destinada la comunicación y de controlar el establecimiento y liberación de las conexiones, así como del acceso de los usuarios al servicio. En servicios de red punto a multipunto o multipunto a multipunto, por ejemplo en una sesión de difusión de un seminario, si un usuario solicita conectarse, comprueba que tiene permiso para hacerlo y lo conecta. El procedimiento así como algunas de las primitivas empleadas se puede ver en la Fig. 6.

Si el servicio es distribuido, esta instancia se crea en el nodo que controle la sesión y si es centralizada, en el nodo servidor del servicio. Los métodos principales de esta clase permiten a los usuarios abrir y cerrar sesiones, unirse a una sesión, abandonarla y renegociar la QoS.

En esta clase se implementan también funciones para localizar recursos en la red. Por ejemplo, cuando un nodo intenta transmitir una secuencia de vídeo codificado en formato MPEG-2 a otro nodo que no dispone del descodificador adecuado. En este caso el nodo receptor rechaza la llamada e indica la causa, entonces se intenta localizar en la red un descodificador que se pueda utilizar. Esta función hace uso de los servicios que ofrece CORBA para localizar los recursos e importar la interfaz que ofrecen estos recursos para su utilización.

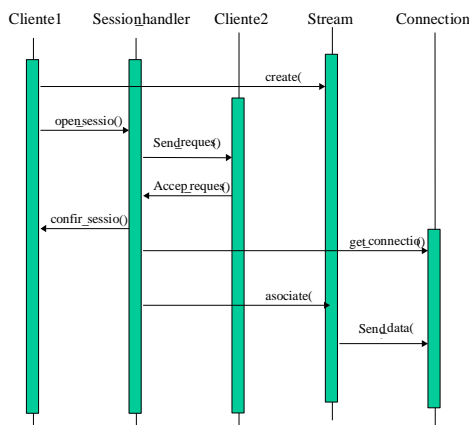


Figura 6.- Establecimiento de un servicio.

La clase *flow_control* controla los aspectos entre flujos. Permite la fijación de las prioridades entre flujos, en caso de no poder transmitir la información de todos los flujos. También gestiona las asociaciones de flujos indicando, en aplicaciones con varios flujos, cual es el flujo principal que se tomará como referencia para, por ejemplo, aspectos de sincronización. El usuario puede señalar un flujo de audio con prioridad sobre los otros y además que las imágenes se sincronicen con él.

Cada sesión tiene asociada un objeto *event_handler* que se encarga de monitorizar y procesar la información de control y gestión que se genera durante la prestación del servicio. En esta clase se implementan parte de las políticas de control de la QoS de las conexiones, puesto que ante un evento que ocurre en la red debe decidir si hay que generar una alarma, si hay que notificarlo al control de la sesión, etc. Básicamente, consiste en una serie de filtros para seleccionar los eventos de la red que interesen. A cada filtro hay que indicarle los parámetros de calidad sobre los que se va a ejercer la vigilancia. Cuando se detecta alguna anomalía en la red se comprueba si afecta a la QoS establecida y las acciones que hay que emprender.

Para cada flujo de información se crea un objeto *connection*. Este objeto se encarga de la coordinación de la comunicación entre los nodos que intervienen en el servicio. Su funcionalidad es similar al gestor de conexiones de una red ATM. Selecciona los recursos necesarios para establecer la conexión, fija aspectos como la selección de *TCP* (*Transport Control Protocol*) o *UDP* (*User Datagram Protocol*) sobre IP si se usa una red de datagramas, la capa *AAL* (*ATM Adaptation Layer*) que se va a utilizar, etc.

Otra función es el control de la QoS requerida, para ello realiza por un lado la traducción de los parámetros de calidad en conexiones de red, y por otro lado, en cooperación con otros objetos, establece la monitorización de la QoS. Recibe cualquier evento que afecte a la calidad de la conexión y lo notifica al gestor de eventos (*event_handler*) para que decida como le afecta al servicio que soporta la conexión.

Esta clase permite cierta jerarquía agrupando varias conexiones en una sola conexión. Por ejemplo, una conexión de vídeo extremo a extremo puede requerir una conexión del extremo emisor hasta un nodo que incluya un descodificador y otra conexión desde éste hasta el extremo receptor. Ambas conexiones se presentan hacia un nivel superior como una única conexión.

Para cada conexión se crean una serie de instancias que representan los recursos de red. Estas clases son las que se encargan de abstraer las peculiaridades de cada tecnología. Implementan los métodos genéricos de la clase *connection* sobre cada tecnología y le ayudan a realizar sus funciones. En la Fig. 5 sólo aparecen las clases asociadas a una conexión.

La clase *control_TP* ofrece las funciones de control sobre los protocolos a nivel de transporte y de red. En redes IP, por ejemplo, implementa la funcionalidad requerida para utilizar los protocolos TCP/IP o UDP/IP. El programador no necesita conocer los mecanismos concretos de estos protocolos sino los métodos de esta clase.

Los protocolos de los niveles más bajos y las funciones de control de los dispositivos de red se modelan en la clase *net_driver*. Las capacidades de control se realizan, normalmente, mediante llamadas al sistema o a una librería que se comunica con los controladores de la tarjeta de red. El modelo se puede adaptar para la utilización de modelos de red programables como el que se muestra en [12].

4. Ejemplo de aplicación

Para demostrar la flexibilidad del modelo se va a aplicar a la realización de un servicio de videoconferencia punto a punto para miembros de un departamento. El nodo que inicia el servicio se encarga también de su control y gestión.

Cada interlocutor genera tres flujos de información, una señal de audio, una de vídeo e información de control. Se va a utilizar como red de transporte una red ATM formada por un conmutador al que están conectados los dos nodos participantes.

Como se puede apreciar en la Fig. 7, los objetos de control se localizan en el nodo que solicita el servicio, el ordenador de la izquierda. En una arquitectura centralizada, estos objetos se crean en el nodo servidor. En este caso cada vez que un cliente quiere iniciar una sesión debe comunicárselo al servidor.

Si por ejemplo el extremo receptor no puede recibir el flujo de vídeo con la codificación que genera el servidor y viceversa, el objeto *session* localiza en la

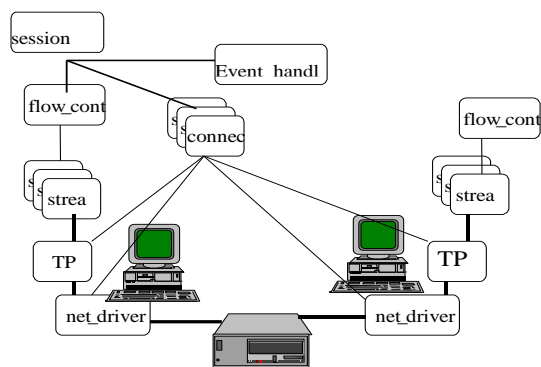


Figura 7.- Arquitectura distribuida

red los recursos adecuados utilizando los servicios que proporciona CORBA. El flujo de vídeo se envía de un nodo al decodificador y de éste al otro nodo. En este caso la conexión de vídeo tiene la forma que se muestra en la Fig. 7. El flujo de vídeo está representado en línea continua y la información de control entre clases por línea discontinua.

Como se puede ver cada sistema que participa en la conexión tiene una instancia de la clase *control_TP*, que en este caso se encarga de decidir la capa AAL que se va a utilizar en la transmisión, y otra instancia de *net_driver* para controlar las tarjetas de conexión a la red.

Cuando es necesario utilizar ciertos recursos compartidos como decodificadores o recursos para establecer flujos *multicast* el proceso es algo más complicado. Primero hay que localizar los nodos donde están y establecer dos conexiones, como se puede ver en la Fig. 7. El control de algunos objetos puede estar compartido por lo cual hay que tomar ciertas precauciones cuando se invocan sus métodos.

Cuando se quiere realizar la gestión de los recursos compartidos es más adecuada una arquitectura centralizada, ya que en este caso sólo es necesario que el servidor conozca las políticas de control que se quieren aplicar. Este nodo podría también realizar las funciones de gestión de los servicios, permitiendo en cada momento conocer las sesiones que hay abiertas y el estado de las mismas.

La ampliación a un servicio en el que puedan participar más de dos usuarios supone la inclusión de un recurso de control sobre conexiones *multicast*, que sería controlado por la *session*.

5. Conclusiones

El modelo genérico que se ha propuesto, permite la programación de distintos servicios multimedia sin necesidad de conocer las particularidades de la tecnología de red que se va a utilizar, simplemente necesitan conocer los métodos de las clases. Así, por ejemplo, no sería necesario conocer las particularidades de la programación con *sockets* UNIX.

El código generado es más robusto puesto que se trabaja a varios niveles de abstracción, dejando los aspectos particulares en los niveles bajos. Además, la utilización de la programación orientada a objetos, aumenta considerablemente la reutilización del software. La utilización de los mecanismos de herencia y polimorfismo de la programación orientada a objetos permite que el modelo se pueda especializar para adaptarse a distintos entornos.

La aplicación de tecnologías como CORBA en el campo de las telecomunicaciones, puede ayudar bastante, tanto en la creación de aplicaciones distribuidas que soporten los nuevos servicios multimedia, como en el control y la gestión de estos servicios.

El modelo puede irse ampliando hasta constituir una arquitectura global que abarque todos los procesos necesarios para la gestión correcta de todos los servicios que se prestan en una red multimedia. Esto permitiría el desarrollo integrado de los nuevos servicios y optimizaría la utilización de los recursos, haciendo los servicios más baratos y por lo tanto más rentables para las empresas.

En los nuevos servicios multimedia es importante considerar los aspectos de la QoS. Se ha incluido un

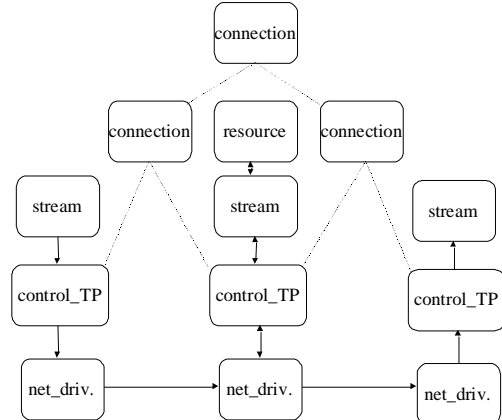


Fig. 7.- Flujo de vídeo en dos tramos

modelo de servicios para especificar la calidad que requieren las aplicaciones que extiende el definido por el IETF. Además, el proceso de fijar y vigilar la calidad está distribuido entre varios objetos.

Agradecimientos

Este trabajo ha sido parcialmente financiado por la Comisión Interministerial de Ciencia y Tecnología (CICYT), proyecto n° TIC96-0743.

Referencias

- [1] Object Management Group. *The Common Object Request Broker: Architecture and Specification*. 2.0 ed., May 1996.
- [2] D. Rogerson. *Inside COM*. Redmond, WA: Microsoft, 1997.
- [3] A. Campbell, G. Coulson and D. Hutchison. "A Quality of Service Architecture". *ACM SIGCOMM, Computer Communications Review*, Vol. 24, pp 6-27. April 1994.
- [4] R. Braden, D. Clark and S. Shenker. "Integrated Services in the Internet Architecture :an Overview". *IETF RFC 1633*. Jun 1994.
- [5] ISO/IEC, "QoS - Basic Framework - CD Text", *Joint ISO/IEC-ITU-T Interim Meeting*, Toronto, January 1995.
- [6] T.P. Barzilia, D. D. Kandlur, A. Mehra and D. Saha. "Design and Implementation of a RSVP-Based Quality of Service Architecture for a Integrated Service Internet". *IEEE Journal on Selected Areas in Communications*. Vol. 16, No. 3, pp 397-413. April 1998.
- [7] P. Owerzarski, M. Diaz and C. Chassot. "A Time-Efficient Architecture for Multimedia Applications". *IEEE Journal on Selected Areas in Communications*, vol 16, No 3, pp 383-395, April 1998.
- [8] R. Braden, L. Zhang, S. Berson, S. Herzog and S Jamin. "Resource reservation protocol (RSVP)-Version 1, functional specification". *RFC 2205*. Sept. 1997.
- [9] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: A transport protocol for real-time applications." Internet draft (work-in-progress) draft-ietf-avt-rtp-*.txt, Mar. 1995.
- [10] ITU-T. "Principles for a Telecommunications Management Network". ITU Recommendation M.3010, 1992.
- [11] C. Aurrecoechea, A. Lazar and R. Stadler. "Toward Building Manageable Multimedia Network Services". *Proceedings of the*

IFIP/IEEE International Conference on Management of Multimedia Network and Services, MMNS'97, Chapman&Hall Publisher, Montreal, Canada, July 1997.

- [12] J. Huard and A. Lazar. "A Programmable Transport Architecture with QoS Guarantees". *IEEE Communications Magazine*. Pp. 54-62. October 1998.

Arquitectura Composicional de Seguridad para Servicios Telemáticos

Lidia Fuentes y Javier López
Depto. de Lenguajes y Ciencias de la Computación
Universidad de Málaga. Campus de Teatinos. E29071- Málaga (ESPAÑA)
Teléfono: 952-132810, Fax: 952-131397
E-mail: {lff,jlm}@lcc.uma.es

Abstract

An important growth of telematic services provision is presently observed. Multimedia and electronic commerce services are outstanding ones. In order to extend the use of such applications, mechanisms that provide communications authenticity and privacy are needed. Moreover, it is necessary to make use of a new software technology that assist the development of applications as open, distributed and evolving systems. This paper introduces an application framework (MultiTEL) designed to allow development and reusability of applications based on the secure and trustful interchange of documents through the Internet. MultiTEL is a compositional framework that encapsulates a multimedia service architecture. In this paper, we propose an extension of MultiTEL for supporting security issues.

1. Introducción

Los recientes avances en el campo de las redes de telecomunicación y en los sistemas distribuidos han proporcionado la aparición de tecnologías de alta velocidad para la propagación de datos a través de sistemas digitales. Las redes abiertas están siendo utilizadas cada vez más como una plataforma para la comunicación, pues permiten rápidos y eficientes intercambios de información con un bajo coste económico asociado y con una fácil accesibilidad.

En el ámbito de las organizaciones privadas, las redes abiertas ofrecen la posibilidad de crear nuevos negocios, nuevos canales de distribución y nuevos métodos de alcanzar al cliente, promoviendo el desarrollo del comercio electrónico. Por otro lado, en el ámbito de las organizaciones públicas, estas redes contribuyen a facilitar el acceso de los ciudadanos a los servicios públicos de las administraciones gubernamentales.

Así, el desarrollo actual y las perspectivas de futuro de las superautopistas de datos y de una infraestructura global de información, es decir, de Internet y de la World Wide Web (WWW), dan paso a toda una variedad de nuevas posibilidades y servicios para las que es necesario tener un marco común de trabajo que permita un fácil desarrollo de nuevas aplicaciones, así como la integración del software ya existente.

Sin embargo, la realización efectiva de esas nuevas posibilidades está influenciada por las inseguridades típicas de las redes abiertas: los mensajes pueden ser interceptados y manipulados, la validez de los documentos se puede negar, o los datos personales pueden ser recolectados de forma ilícita, por lo que los documentos electrónicos importantes son, con frecuencia, intercambiados en

redes cerradas donde hay relaciones de confianza mutua entre los usuarios, y no en redes abiertas donde el acceso es libre y generalmente se realiza de forma anónima. Por lo tanto, también en este sentido es necesario hacer uso de nuevos medios, como las técnicas criptográficas, que ayuden a los agentes involucrados a crear entornos de confianza que satisfagan los niveles de seguridad requeridos y donde tanto los emisores como los receptores de información privada importante y significativa dispongan de medios que aseguren la privacidad y la autenticación de la información digital transmitida.

Los servicios de seguridad, al igual que otras propiedades no funcionales, tradicionalmente se han incluido siempre como parte integrante de los módulos funcionales de una aplicación, o más recientemente como servicios comunes dentro de plataformas de componentes distribuidos tipo CORBA [1]. Ambos enfoques implican que los componentes van a contener además de su comportamiento interno métodos relativos a gestionar la seguridad, lo que complica el diseño interno del componente y dificulta enormemente su reutilización. Por otro lado, dado que los sistemas abiertos sobre la Web evolucionan para incrementar su nivel de seguridad, es necesario definir modelos que permitan incorporar a las aplicaciones y a los componentes ya desarrollados propiedades de seguridad introduciendo mínimos cambios en el código.

Recientemente, a raíz del creciente interés que ha despertado el estudio de la arquitectura software han aparecido diferentes paradigmas para el diseño de aplicaciones distribuidas abiertas, como la orientación a componentes [2], los marcos de trabajo (*frameworks*) [3], la programación reflexiva [4] o los patrones de diseño [5] los cuales proporcionan mecanismos que permiten incluir propiedades no funcionales, entre ellas la de seguridad, de forma

modular e independiente. Nosotros hemos diseñado un marco de trabajo basado en componentes que incorpora muchas de las características de los diferentes paradigmas antes mencionados. Este marco de trabajo denominado MultiTEL permite adaptar y derivar a partir de una arquitectura base servicios avanzados de telecomunicación concretos como el vídeo a la carta, videoconferencias, aplicaciones de trabajo colaborativo, etc. [6].

En el presente trabajo se pretende extender a MultiTEL con componentes de seguridad e incorporar estas características a las aplicaciones ya desarrolladas. En primer lugar definiremos aquellas propiedades de seguridad que deben ser incorporadas además de todos los procesos necesarios para, por ejemplo, gestionar y garantizar la confidencialidad de los datos y la autenticidad de las claves. Definiremos una arquitectura basada en el modelo de componentes propuesto por MultiTEL como un conjunto de componentes básicos que pueden ser especializados.

Uno de nuestros objetivos es garantizar la interoperabilidad de las aplicaciones con usuarios que soporten o exijan diferentes niveles de seguridad. MultiTEL soporta la composición dinámica de componentes propios de un servicio o externos (*plug-ins*) por lo que se facilita el uso por parte de la aplicación de aquellos componentes de seguridad que sean propiedad de los clientes. Por otro lado, el marco de trabajo MultiTEL se extiende mediante un mecanismo de caja negra que consiste en agregar nueva funcionalidad a las aplicaciones derivadas mediante composición y delegación. Esto significa que para incorporar servicios de seguridad a una aplicación sólo es necesario agregar los nuevos componentes, extendiendo de esta forma su arquitectura, sin necesidad de modificar ningún otro componente.

2. MultiTEL: Plataforma de Componentes Distribuidos

Actualmente existen diversas plataformas composicionales como DCOM y JavaBeans que permiten la construcción de componentes software independientes (llamados *commercial-off-the-shelf-components* o COTS), los cuales deberán ser posteriormente ensamblados de acuerdo a algún marco de trabajo (*framework*) composicional. En esta línea, nosotros hemos diseñado e implementado el marco de trabajo MultiTEL [7] que define la composición dinámica de componentes Java de acuerdo a una arquitectura marco para servicios avanzados de telecomunicación. Los componentes, implementados como objetos Java, se ejecutarán en una plataforma distribuida que utiliza Java/RMI como mecanismo de comunicación básico y cuyo uso y configuración se realizará a través de la Web.

El objetivo inicial de nuestro trabajo en MultiTEL fue el desarrollo de un modelo de servicios abierto que reflejase, a nivel de diseño, la colaboración y sincronización necesaria entre los componentes del servicio. Para conseguir estos objetivos nos inspiramos en los modelos de componentes basados en la separación de los aspectos de computación y coordinación en dos entidades diferentes: los componentes y los conectores [8]. El modelo de componentes que hemos propuesto [9] e implementado como parte de la plataforma de MultiTEL, nos permite diseñar componentes software como entidades independientes que podrán ser insertados en diferentes aplicaciones. Concretamente, nos ha permitido realizar la composición dinámica de un servicio con dispositivos multimedia, modelados como componentes, y que han sido desarrollados e instalados por diferentes fabricantes (COTS) [10], consiguiendo que usuarios con recursos heterogéneos pudiesen participar en el mismo servicio.

De la misma forma la flexibilidad del modelo de componentes y conectores nos ha permitido incorporar a las aplicaciones requisitos no funcionales, concretamente de seguridad como el que nos ocupa en el presente trabajo. La principal contribución es el diseño de componentes que incorporan aspectos de seguridad, a una aplicación ya diseñada y también de forma dinámica en tiempo de ejecución, si la aplicación así lo requiere.

2.1. Modelo de Componentes

El marco de trabajo MultiTEL incluye la implementación de un modelo composicional en el lenguaje Java. Este modelo proporciona la base para la implementación de la plataforma distribuida y de la arquitectura específica de los servicios multimedia. Los componentes y los conectores son entidades de primer nivel.

Los componentes son entidades pasivas y los conectores implementan protocolos de comunicación entre dos o más componentes. Dado que el objetivo principal del modelo es la integración de COTS dentro de una misma aplicación, éstos informarán de sus cambios de estado mediante la simple propagación de un evento hacia el entorno. Los componentes no poseen ningún tipo de información acerca de cómo sus computaciones internas afectarán al resto de la aplicación. Este enfoque permite implementar la colaboración entre componentes distribuidos de forma transparente a los propios componentes, dado que será un conector el que sincronizará, mediante la captura de eventos, las diferentes instancias de componentes de acuerdo a un protocolo específico de la aplicación.

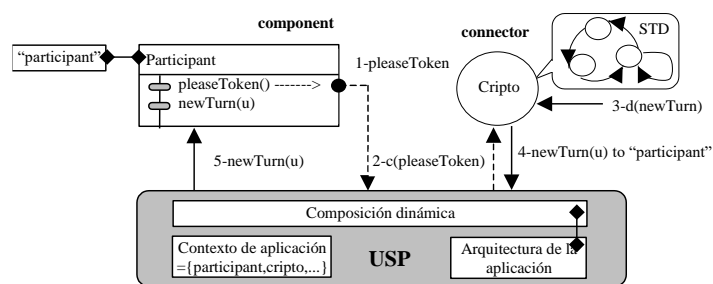


Fig. 1. Modelo de componentes y conectores. Composición dinámica por parte de la USP.

Por lo tanto, podemos ver a los conectores como abstracciones de patrones de coordinación, donde su implementación será dependiente de la aplicación. Mediante la gestión selectiva de eventos propagados por componentes externos, los conectores son capaces de controlar la ejecución de una parte de la aplicación (varios componentes) constituyendo en sí mismos una microarquitectura. El protocolo interno de un conector se define proporcionando el diagrama de transición de estados (STD) que irá evolucionando a medida que lleguen eventos del exterior. Su implementación se ha realizado aplicando el patrón de diseño *State* propuesto en [5]. En la figura 1 se muestra un componente que modela a un participante de una conferencia y que propaga el evento *pleaseToken* el cuál es gestionado por el conector de seguridad *Cripto* que codifica el mensaje antes de ser enviado al resto de los participantes.

La composición entre componentes y conectores la lleva a cabo el componente denominado USP (parte del servicio de usuario o User-Service Part). Existe un componente USP en la máquina de cada usuario participante de un servicio conteniendo la configuración global de la aplicación que personaliza para dicho usuario. La función principal de la USP es realizar la composición de componentes y conectores de forma dinámica, para lo cual tiene en cuenta la arquitectura de la aplicación que almacena internamente. Se entiende por arquitectura el conjunto de conexiones en base a métodos y a eventos que se establecen entre componentes y conectores. De esta forma MultiTEL suple las carencias que suelen encontrarse en otras plataformas de componentes tipo CORBA, en las cuales no se aborda adecuadamente los aspectos de arquitectura software [11]. Tal y como podemos ver en la figura 1 la USP además incluye el contexto de aplicación, que comprende el conjunto de componentes y conectores que han sido instanciados para un usuario determinado. Este contexto se actualizará de forma dinámica y se tendrá en cuenta, junto con la arquitectura de la aplicación, durante la composición dinámica.

La USP también implementa métodos para la difusión de mensajes y eventos y métodos que gestionan el paso de mensajes de conectores a

componentes. Hemos extendido el paso de mensajes tradicional de la programación orientada a objetos para aumentar su flexibilidad. Por ejemplo, uno de los tipos de direccionamientos de componentes permiten a un conector identificar al destinatario de un mensaje indicando el rol que desempeña dentro de una determinada arquitectura. En la figura 1 podemos ver como la USP decide cuál es el conector apropiado (o conectores) que deben gestionar el evento *pleaseToken* el cual es respondido por el conector enviando el mensaje *newTurn* a los componentes que desempeñan el rol "participant" dentro de dicha aplicación. Debido a que los conectores sólo necesitan conocer el rol del componente destino serán capaces de enviar mensajes a componentes externos (ej: componentes que modelen recursos locales, multimedia o de seguridad), con el único requisito de que deben registrarse para desempeñar el rol indicado. Los roles de los componentes destinatarios de los mensajes de un conector pueden pasarse como parámetros, lo que facilita e incrementa mucho la reutilización de los propios conectores como patrones de diseño. Veremos la utilidad de esta característica en el ejemplo de la sección 4.

2.2. Estructura en capas

La plataforma MultiTEL se estructura en tres capas diferentes: el nivel de aplicación, la USP y el núcleo. La figura 2 muestra la estructura de una aplicación para un determinado cliente, donde puede verse que los dos primeros niveles están formados por clases Java transferidas a través de la red mientras que el núcleo es la parte residente de la plataforma.

- *Nivel de Aplicación.* Los servicios multimedia se ejecutan mediante la composición dinámica de componentes de servicio, multimedia y de red. Los componentes de servicio modelan las interacciones con el usuario, esto es la lógica del servicio. Los componentes multimedia modelan los recursos multimedia del cliente (cámaras, micrófonos, etc.) y los protocolos de gestión y control de dispositivos multimedia. La gestión y reserva de canales de banda ancha multipunto se modelan como parte de la arquitectura de red. La mayoría de los componentes multimedia y de red son dependientes de la máquina del cliente por lo que se instancian de forma local.
- *USP.* Es una applet Java, por lo que los usuarios se unen a un servicio mediante la ejecución de la correspondiente USP desde un navegador. El usuario interactúa con el servicio a través de su USP local que representa el punto de acceso al servicio. Esta se encarga de conectar entre sí los componentes y conectores locales y éstos con el

resto de los componentes pertenecientes a los

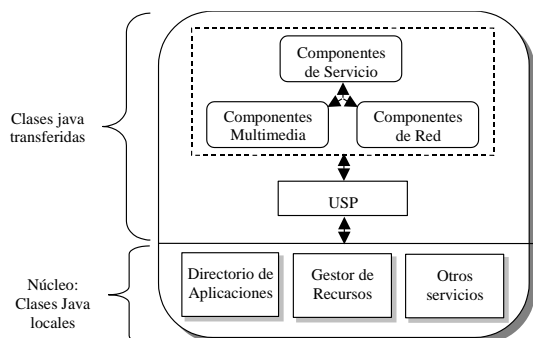


Fig. 2. Arquitectura de la plataforma MultiTEL.

usuarios participantes del servicio.

- **Núcleo.** Incluye herramientas y servicios relacionados con la compartición y ejecución de aplicaciones sobre la Web. Uno de sus componentes es el gestor de recursos encargado de crear los componentes locales del cliente. Este gestor instanciará componentes que modelen dispositivos multimedia, conexiones virtuales y componentes de seguridad adquiridos previamente por el usuario. De esta forma la USP compondrá estos componentes externos mediante un mecanismo de *plug-and-play*. Los servicios de localización de componentes remotos también pertenecen a esta capa.

Aparte de estos servicios básicos para la ejecución de servicios, MultiTEL ofrece herramientas para la edición (Directorio de Componentes) y composición de componentes (Generador de Servicios) que soportan el proceso de derivación de servicios. Se incluye también una herramienta para la configuración y ejecución de aplicaciones distribuidas denominada Directorio de Aplicaciones.

3. Servicios de Seguridad

El buen resultado de los procedimientos tradicionales basados en soporte papel depende en gran medida de que las partes intervinientes tengan confianza en los medios que utilizan para realizar las funciones típicas de privacidad, prueba de identidad, autorización, firma, certificación, etc. Como resultado, se han desarrollado un conjunto de procedimientos y protocolos físicos que especifican cómo crear registros de tal forma que una tercera parte pueda arbitrar disputas posteriores. El punto esencial es que la información tiene una existencia física en forma de documento en papel que ha sido firmado, o autorizado, o certificado, etc.

Sin embargo, en la sociedad de la información en la que la posesión, el control, la transferencia o el acceso están basados en información no-física, es decir, información cuya existencia no está esencialmente ligada a un registro físico, es esencial encontrar medios para realizar todas las funciones

asociadas con el establecimiento de la integridad de esa información, basados únicamente en la evidencia interna presente en la información en sí misma. Es decir, algunas de las funciones comunes tradicionalmente asociadas con los documentos requieren alguna función análoga para los documentos electrónicos. Estas funciones pueden contemplarse como servicios de seguridad, cada uno con unos requerimientos propios y particulares.

No hay un único medio técnico para solucionar estos problemas, pero sí hay un elemento común, y este es el de la transformación criptográfica de aquella información que requiere de seguridad. Estas transformaciones posibilitan que una o más entidades que conozcan un dato secreto puedan realizar una operación en la información para que ésta quede protegida de amenazas externas. De esta forma, se previene el fraude, o bien se detecta, en los sistemas donde la información en sí misma no tiene una existencia física.

Son muchas las razones por las que algunos usuarios, externos o no, pueden desear comportarse de forma fraudulenta. Por ejemplo:

- Obtener acceso no autorizado para acceder a la información;
- Impersonar a algún otro usuario;
- Evitar la responsabilidad de haber originado cierta información;
- Alegar falsamente haber recibido información de algún otro usuario;
- Alegar falsamente no haber enviado nunca cierta información;
- Alegar falsamente no haber recibido información que efectivamente fue recibida;
- Modificar los privilegios de acceso de otros usuarios o los suyos propios;
- Insertarse en el canal de comunicación de otros usuarios;
- Comprobar quién accede a cierta información;
- etc.

Debido a esta multitud de razones, la problemática de la integridad de la información y de la aplicación de los principios criptográficos es muy amplia. Mediante la utilización de los servicios de seguridad a los que antes hacíamos referencia, se pueden evitar estas amenazas. Así, atendiendo a los posibles tipos de amenazas, los servicios necesarios podrían ser tan variados como identificación, autorización, firma, notariación, testificación, responsabilidad, acuse de recibo, certificación de origen y/o destino, derecho de acceso, validación, tiempo de ocurrencia, endoso, autenticación, voto, pertenencia, registro, aprobación/desaprobación, privacidad, etc. [12].

Es obvio que la lista es demasiado larga y, por lo tanto, no resulta una guía útil para organizar los servicios de seguridad. Por el contrario, la investigación y el desarrollo de la seguridad (especialmente aquella referida esencialmente a las

redes de ordenadores) se centra en algunos servicios generales que engloban las distintas funciones requeridas [14]. La clasificación más útil es la siguiente:

- **Confidencialidad:** Requiere que la información en un sistema de computación y la información transmitida sea accesible sólo por las partes autorizadas.
- **Autenticación:** Requiere que el origen del mensaje sea identificado correctamente, con la seguridad de que la identidad no es falsa.
- **Integridad:** Requiere que la información almacenada en el sistema de computación y la información transmitida sólo pueda ser modificada por las partes autorizadas.
- **No repudio:** Requiere que ni el emisor ni el receptor de un mensaje puedan negar la transmisión.
- **Control de acceso.** Requiere que el acceso a los recursos de información sea controlado por el sistema destino.
- **Disponibilidad:** Requiere que la información del sistema de computación esté disponible para las partes autorizadas cuando sea necesario.

Estos servicios se ven afectados por la diversidad de algoritmos de cifrado/descifrado que se pueden utilizar, por los diferentes protocolos de petición o de intercambio de claves y las posibilidades que esto implica en cuanto a la utilización de entidades intermediarias, así como por los posibles bloques de construcción (funciones hash, códigos de autenticación de mensajes, etc.) utilizados en los protocolos donde estos servicios están involucrados.

4. Extensión de aplicaciones con componentes de seguridad

En este apartado partiremos de una aplicación sencilla de videoconferencia ya diseñada y la extenderemos con componentes de seguridad que ofrezcan alguno de los servicios presentados en la sección anterior. A modo de ejemplo se describe sólo el servicio de confidencialidad en términos generales y posteriormente se diseña su arquitectura utilizando el modelo de componentes y conectores. A continuación veremos como introducirlos provocando el mínimo impacto en una videoconferencia ya diseñada. En este ejemplo, nos centraremos únicamente en ver cómo se puede incrementar la seguridad a la hora de intercambiar mensajes entre un conector y un componente que residen en distintos sistemas.

4.1. Componentes de seguridad

Como ya se ha comentado en el apartado anterior, el servicio de confidencialidad proporciona la protección de los datos transmitidos. La protección es eficiente ante ataques pasivos, es decir, ante aquellos en los que el objetivo del oponente es obtener la información transmitida sin realizar modificación alguna sobre el flujo de datos. Se

pueden identificar varios niveles de protección. El servicio más amplio protege a todos los datos transmitidos entre dos usuarios durante un periodo de tiempo. Pero también se pueden definir formas más recortadas de este servicio, incluyendo la protección de un único mensaje o incluso de campos específicos dentro de un mensaje. Estos recortes son, por lo general, menos útiles que la opción más extensa porque resultan más complicados y caros de implementar.

Por lo tanto, nos referimos, a continuación, al servicio de confidencialidad en su forma más general. El esquema correspondiente al uso de ese servicio se muestran en la figura 3. En esta figura se muestra la utilización del servicio de confidencialidad bajo la operación de criptografía simétrica y de criptografía asimétrica (o de clave pública). Se puede observar que los bloques principales de ambos procesos son iguales; es decir, el usuario *A*, a partir de un mensaje *M*, construye un mensaje cifrado aplicándole la función (o algoritmo) de cifrado *E*. El destinatario del mensaje, el usuario *B*, recibe el texto cifrado y mediante la aplicación de la función inversa, es decir, la función de descifrado *D* es capaz de obtener nuevamente el texto original. Ambas funciones de cifrado y descifrado están complementadas con la utilización de una clave. Es precisamente la obtención de esta clave la que diferencia el modelo de operación entre ambos métodos criptográficos (además de la diferencia natural en los algoritmos a utilizar).

En la utilización del servicio de confidencialidad mediante un criptosistema simétrico la clave K_{AB} , denominada clave de sesión, es una clave secreta que puede ser originada por el propio usuario *A*, o bien puede ser obtenida a través de una entidad intermediaria, el Centro de Distribución de Claves (o KDC). Por el contrario, la utilización del servicio de confidencialidad mediante un criptosistema asimétrico requiere de la utilización de un par de claves matemáticamente relacionadas, K_{Bpu} y K_{Bpr} , que están asociadas al usuario destino *B*. La clave utilizada por el usuario *A* para cifrar, es decir, K_{Bpu} , se obtiene normalmente a partir de un certificado de clave pública, emitido también por otra entidad intermediaria, la Autoridad de Certificación. Dicho certificado puede estar en algún repositorio de claves [15], o puede ser proporcionado al usuario *A* por la Autoridad de Certificación [16] o por el propio usuario *B* [17].

Cualquiera de los servicios de confidencialidad nombrados anteriormente pueden diseñarse de acuerdo al modelo de componentes y conectores de MultiTEL. En la figura 4 se puede ver la arquitectura de un servicio de confidencialidad simétrico el cuál mediante parametrización y sustitución de componentes puede transformarse fácilmente en uno asimétrico.

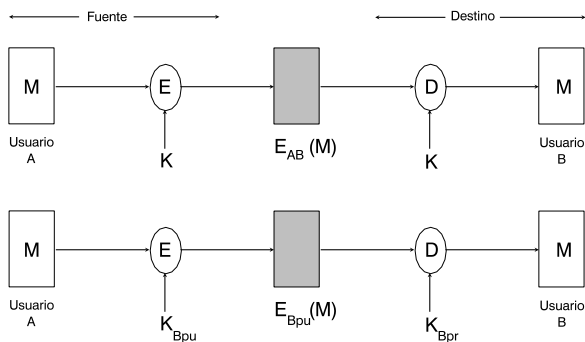


Fig. 3. Servicio de Confidencialidad mediante Criptosistemas Simétrico y Asimétrico

La entrada a este subsistema será la petición de cifrar un mensaje enviada por un conector de una aplicación y como resultado se devolverá el mensaje original al componente destinatario. El componente *Confid* será el encargado de recibir dichas peticiones, propaga el evento que permite iniciar el cifrado del mensaje con un algoritmo determinado (encapsulado por dicho componente). En el receptor, cuando se recibe el mensaje cifrado se lo hace llegar al conector encargado de iniciar el descifrado correspondiente. La misión de los conectores *SConfidC* y *RConfidC* es la de encargarse a componentes específicos las diferentes sub tareas que conllevan el cifrado/descifrado de mensajes. Estos conectores podrán ser reutilizados en el diseño de otros servicios de seguridad que incluyan las mismas funciones de *encrypt()* y *decrypt()*.

4.2. Caso de estudio

En este apartado partimos de una aplicación ya diseñada de videoconferencia que extendemos con los componentes y conectores de confidencialidad que definimos anteriormente. Nuestro objetivo es reutilizar íntegramente los componentes y conectores de la videoconferencia y modificar únicamente la arquitectura del servicio definida dentro de la USP. Por motivos de claridad sólo incluiremos una parte pequeña de la planificación de la videoconferencia, y que consideramos suficiente para ilustrar el mecanismo propuesto. Tal y como se muestra en la figura 5 tenemos un componente que modela a un participante (*Participant*), su conector *SSchedPP* que controla su participación dentro del servicio, el componente gestor (*Manager*) y su conector *SSchedMP* encargado de controlar la planificación de la conferencia. Ambos conectores incluyen como parámetros el rol que desempeñan, dentro de este servicio concreto, los componentes

destinatarios de los mensajes.

El participante podrá emitir eventos típicos para realizar peticiones de palabra, solicitar intervenciones más extensas y recibirá mensajes de respuesta, de finalización de intervención, entre otros. Este intercambio de mensajes envía el texto en claro sin realizar ningún tipo de cifrado o de autenticación sobre los datos o la identidad del usuario emisor.

Como hemos comentado anteriormente la arquitectura de la aplicación se define como parte de la USP. La clase base que se debe especializar define métodos para dar de alta componentes, conectores, lectura de parámetros de configuración y para definir las conexiones entre ellos en base a eventos y a mensajes. A continuación puede verse la arquitectura del servicio implementada como una clase heredera de la clase base USP (clase *ServiceUSP*) que como vemos sólo necesita redefinir los métodos para la instanciación de la arquitectura: *defComponents()*, *defConnectors()*, *loadOrganizationParameters()* y *loadConnections()*.

```
public class VideoConference1
    extends ServiceUSP {

    public void defComponents(){
        component("participant", "Participant",
            "local", "mtsb.vc.VCParticipant");
        component("manager", "Manager", "remote",
            "mtsb.vc.VCManager");
        ...
    }

    public void defConnectors(){
        connector("sschedpp", "Participant",
            "local", "mtsb.vc.SSchedPP");
        connector("sschedmp", "Manager",
            "remote", "mtsb.vc.SSchedMP");
        ...
    }

    public void loadOrganizationParameters(){
        /* Lectura de parámetros de
        organización: algorithm */
        /* Ej: Proporcionar "manager" como
        parámetro al conector "sschedpp" */
    }

    public void loadConnections(){
        String[] events1={"pleaseToken"};
        handlesEventsFrom("sschedpp",events1,
            "participant");
        ...
    }
}
```

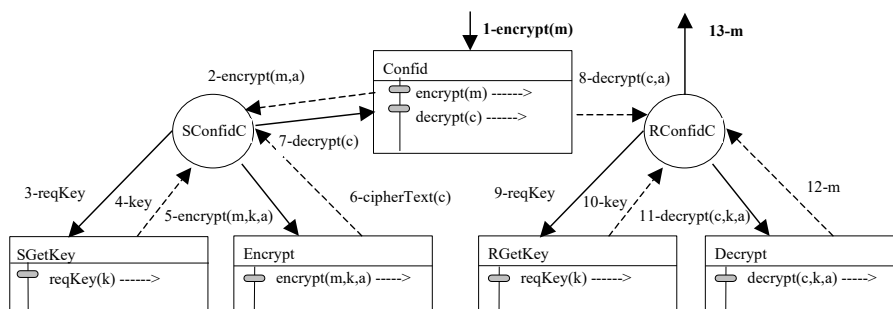


Fig. 4. Componentes y conectores para un servicio de confidencialidad mediante criptosistema simétrico.

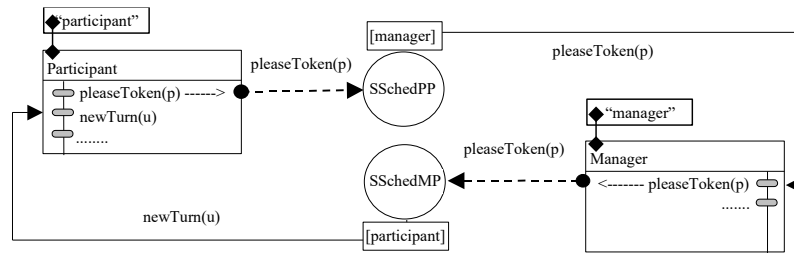


Fig. 5. Arquitectura de planificación para un servicio de videoconferencia

Para registrar componentes y conectores dentro de la USP se debe proporcionar en primer lugar el rol que desempeña a nivel de arquitectura representado por un identificador único. Este identificador de rol será utilizado en la definición de las conexiones en las que el componente o conector tomará parte. En el ejemplo vemos que al componente participante se le asigna el rol "participant" y al gestor del rol "manager". El segundo parámetro designa el tipo de usuario al cuál se asocia; en el caso de una videoconferencia simple podrá ser un participante normal (*Participant*) o el gestor del servicio (*Manager*). En tercer lugar se indicará alguna información acerca de la localización del componente o conector que podrá ser local, remota o incluso en caso de tratarse de componentes que modelen recursos físicos podría proporcionarse la URL correspondiente. Por último, se indica el nombre específico de la clase Java que implementa al componente o al conector, la cual ha sido definida independientemente del rol que ésta desempeña dentro de esta arquitectura concreta.

Al igual que se registran los componentes del servicio, que serán transferidos como parte del applet USP, se pueden registrar componentes externos proporcionados por el cliente. En este caso, su participación estaría prevista dentro del servicio y posteriormente en tiempo de ejecución se necesitaría proporcionar únicamente el nombre de la clase que lo implementa. De esta forma se realizaría el enlazado dinámico del servicio con *plug-ins* que modelan recursos específicos del cliente como, por ejemplo, las tarjetas inteligentes, que se utilizan como soporte de las claves privadas de los usuarios y en ocasiones realizan internamente el proceso de firma digital.

Toda esta información proporcionada a la USP acerca de los componentes y conectores es configurable por parte de cualquier usuario que desee organizar un servicio, esto es, que necesite adaptar la definición de un servicio a sus preferencias. Dentro de la configuración de un servicio se proporcionan valores a los parámetros de los componentes y conectores. Por ejemplo, el conector de confidencialidad del emisor (*SConfidC*) presentado en la sección anterior (figura 4) tiene como parámetro un algoritmo de cifrado que será indicado por el cliente organizador según sus propios requisitos.

En relación a la transferencia segura de información, ésta se puede aplicar a dos niveles: eventos y mensajes. Si quisiésemos propagar eventos codificados bastaría con modificar las conexiones que aparecen en el método *loadConnections()*. El objetivo sería sustituir la conexión del ejemplo por otra que permita codificar el evento enviado, y de esta forma reconfigurar la arquitectura del servicio. En cuanto a la codificación de los mensajes se resuelve de forma más sencilla. Además de dar de alta los nuevos componentes y conectores del subsistema de confidencialidad definiremos una nueva versión del método de establecer conexiones:

```
public void loadConnections(){
    String[] message1={"pleaseToken"};
    String[] renamed1={"encrypt"};

    receiveMessageFrom("confid",message1,
        "sconfidc",renamed1);
    ...
}
```

Simplemente necesitamos definir que el mensaje *pleaseToken* que anteriormente se enviaba al componente *manager* debe ser enviado al nuevo componente *confid* pero renombrado. El renombrado de mensajes se define únicamente para el componente *confid* con el método *receiveMessageFrom()*. Dentro del método *loadOrganizationParameters()* se indica que el conector *sschedpp* en esta versión enviará sus mensajes al nuevo componente *confid* en vez de realizarlo al componente *manager* como hacía anteriormente. Igualmente habrá que parametrizar el conector de confidencialidad del receptor (registrado con el rol "rconfidc") indicando el rol del componente destinatario final de dicho mensaje *m* (el rol "manager").

```
public void loadOrganizationParameters(){
    /* Proporcionar "confid" como parámetro
    al conector "sschedpp" */
    /* Proporcionar "manager" como
    parámetro al conector "rconfidc" */
}
```

Como vemos no hemos tenido que modificar ninguno de los componentes del servicio, simplemente extendimos y reconfiguramos su arquitectura. Como hemos visto en la sección anterior, mediante parametrización y sustitución de componentes podremos obtener los mismos

resultados para el servicio de autenticación. De esta forma hemos conseguido simplificar notablemente el diseño tanto de las aplicaciones como de los componentes, y lo que es más importante hemos definido un modelo que permite incorporar de forma incremental a las aplicaciones nuevos requisitos como el de confidencialidad facilitando enormemente su evolución.

5. Conclusiones y Trabajos futuros

El desarrollo actual y las perspectivas de futuro de Internet y de la World Wide Web dan paso a toda una serie de servicios para los que es necesario tener un marco común de trabajo que permita tanto un fácil desarrollo de nuevas aplicaciones como la integración del software ya existente. La realización efectiva de esas nuevas posibilidades están influenciadas por las inseguridades típicas de las redes abiertas. Por lo tanto, es necesario hacer uso de técnicas criptográficas que creen entornos de confianza que aseguren la privacidad y la autenticación de la información digital transmitida. En consecuencia, es necesario definir modelos que permitan incorporar a las aplicaciones ya desarrolladas esas propiedades de seguridad introduciendo mínimos cambios en el código.

Así, en el presente artículo, se ha extendido el marco de trabajo MultiTEL con componentes para incorporar las características de seguridad a las aplicaciones ya desarrolladas. Más concretamente, se han incorporado aquellos que se han considerado como servicios básicos de seguridad y hemos presentado un ejemplo basado en el servicio de confidencialidad, el cual ha sido utilizado para extender una aplicación de videoconferencia previamente diseñada.

La flexibilidad del modelo y de la plataforma de componentes definida en MultiTEL nos ha permitido introducir características de confidencialidad a los servicios, simplemente mediante la reconfiguración y extensión de su arquitectura. Mediante este mecanismo se podrá seguir agregando otras propiedades de forma incremental, según lo vayan requiriendo las aplicaciones. Por otro lado, la arquitectura de seguridad propuesta representa un patrón de diseño que puede ser reutilizado en cualquier servicio telemático. Esta arquitectura se ha definido en base a componentes de seguridad básicos, pero podrá ser instanciada con otros componentes para obtener nuevos servicios de seguridad.

Como trabajo futuro se pretende incorporar a MultiTEL esos nuevos servicios de seguridad que ayuden a solucionar algunos de los requisitos que son propios de recientes campos de actuación como el comercio electrónico, donde se precisan funciones de certificación de transacciones, notarización de documentos, anonimidad, auditoría, pagos electrónicos, etc.

Referencias

- [1] OMG, *CORBA 2.0 "Interoperability"*. Informe Técnico Núm. 95-03-10, Universal Networked Objects, 1995.
- [2] C. Szyperski. *"Component Software. Beyond Object-Oriented Programming"*. Addison Wesley, 1998.
- [3] M.E. Fayad y D.C. Schmidt. *"Object-Oriented Applications Frameworks"*. Communications of the ACM, 40(10): 32-38, Octubre 1997.
- [4] Kiczales, G. et al., *"Aspect-Oriented Programming"*. Actas del ECOOP'97, No. 1241 de LNCS, pp. 220-242, 1997.
- [5] E. Gamma et al., *"Design Patterns"*. Addison Wesley, Reading Mass, 1995.
- [6] L. Fuentes y J.M. Troya, *"MultiTEL : A Component-Oriented Framework in the Domain of Multimedia Telecommunication Services"*. Capítulo del libro *Object-Oriented Application Frameworks*. Wiley & Sons. 24 páginas, próxima publicación en Julio 1999.
- [7] MultiTEL. <http://www.lcc.uma.es/~lff/MultiTEL>
- [8] R. Allen y D. Garlan, *"Formal Connectors"*. Informe Técnico CMU-CS-94-115, Carnegie Mellon University, Marzo 1994.
- [9] L. Fuentes, and J.M. Troya, *"A Component-Oriented Architecture to Design Multimedia Services on a Distributed Platform"*. *Worldwide Computing and Its Applications*, LNCS No. 1274, pp. 90-105, Agosto 1997.
- [10] L. Fuentes and J.M. Troya, *"A Java Framework for Web-Based Multimedia and Collaborative Applications"*. *IEEE Internet Computing*, 3(2): 55-64, Marzo/Abril 1999.
- [11] D. Krieger and R.M. Adler, *"The Emergence of Distributed Component Platforms"*. *IEEE Computer*, pp. 43-53 March 1998.
- [13] G. Simmons, *"Contemporary Cryptology. The Science of Information Integrity"*. IEEE Press, 1992
- [14] W. Stallings, *"Network and Internetwork Security. Principles and Practice"*. Prentice-Hall, 1996
- [15] ISO/IEC JTC1/SC 21. *Draft Amendments DAM 4 to ISO/IEC 9594-2, DAM 2 to ISO/IEC 9594-6, DAM 1 to ISO/IEC 9594-7, and DAM 1 to ISO/IEC 9594-8 on Certificate Extensions*, 1996.
- [16] J. López et al., *"A System For Public Keys Service In the Spanish Research & Academic Network"*. VII Annual Conference of the Internet Society, 1997.
- [17] P.Zimmerman, *"The Official PGP User's Guide"*. MIT Press, 1995.

Una experiencia de vídeo bajo demanda soportada en tecnología web, CORBA y SDL.

Rubén de Diego, Justo Carracedo, Carlos Ramos, Vicente Hernández
Departamento de Ingeniería y Arquitecturas Telemáticas
EUIT Telecomunicación, UPM. Crtra de Valencia Km 7, 28031 MADRID
Telf: 91-3367823, Fax: 91-3367817
Correo-e: (rdiego.carracedo.cramos.vhernandez@diatel.upm.es)

Abstract

This paper presents a practical experience based on the use of Web, CORBA and SDL technologies to implement a platform provided with interactive multimedia services. The system is focused to a Video On Demand (VoD) service. In this proposal Web technology is useful not only for user interfaces but also as a nexus among the other technologies, in such a way that in co-operation with CORBA it permits that each of the system elements can be developed separately by using different technologies (Java, C++ and SDL). This experience has been developed in the framework of the DIATEL department, as a pilot project, and is considered as a continuation of the work carried out by the authors in MEDAS and OTELSO projects.

1. Introducción

El área de los servicios multimedia interactivos es en la actualidad uno de los más activos en el mundo de la telemática. Investigadores, organismos normalizadores, fabricantes, operadores y proveedores de servicios están inmersos en un proceso de elaboración de propuestas innovadoras que deberán dar satisfacción a las necesidades que plantean los usuarios en la actualidad y, sobre todo, a las que aparecerán dentro de unos pocos años.

La experiencia presentada en este artículo describe una propuesta en la que se apuesta por Internet/ATM como soporte de los futuros servicios. En la propuesta no se entra a analizar los medios de comunicación en los que, a su vez, se apoyarán estos protocolos, dando por hecho que con tecnologías basadas en modems de alta velocidad y fibra o mixtas fibra-coaxial, en un plazo breve, sino en el presente, se tendrán satisfechos los requisitos básicos exigibles a la red.

La propuesta apuesta igualmente por una interfaz de usuario unificada mediante la tecnología web, lo que permitirá ofrecer mecanismos de acceso amables para el usuario y de fácil desarrollo para el proveedor de servicios.

El objetivo fundamental de la experiencia ha sido combinar distintas tecnologías software en el desarrollo de aplicaciones distribuidas de alta complejidad y exigencia, como es el caso de las aplicaciones multimedia interactivas. Así, se ha tenido la ocasión de continuar las experiencias llevadas a cabo en proyectos anteriores (MEDAS [5] y OTELSO [6] fundamentalmente), en los que se trabajó con técnicas de análisis y diseño orientado a objetos, técnicas de especificación formal y arquitecturas distribuidas orientadas a objetos (CORBA). En esta ocasión los citados trabajos se han complementado con el uso de tecnología web, que ha demostrado ser un interesante “compañero de viaje” en el desarrollo de este tipo de aplicaciones, dado que desde el primer momento se ha conseguido apreciar la

relación sinérgica existente entre las tecnologías mencionadas.

Los servicios interactivos de distribución de vídeo representan un cambio fundamental en el mundo de la televisión actual. En un sistema tradicional de televisión por difusión, múltiples estaciones emiten sus programas simultáneamente y el usuario selecciona un sólo canal que le ofrece una programación determinada. Esto trae como resultado que el usuario vea afectada su capacidad de elección debido a la difusión paralela de programas diferentes en diferentes canales. Por el contrario, un sistema interactivo permite que toda la programación esté disponible para los usuarios en cualquier momento. No existen restricciones temporales.

En el servicio de vídeo bajo demanda (VoD) cada usuario elige el vídeo deseado y puede utilizar funciones de control mientras lo visualiza, sin necesidad de almacenamiento local. El usuario puede solicitar y visualizar un vídeo, dentro de los ofertados, en cualquier momento. Las funciones de control de las que se dispone son: parar, continuar, adelantar, rebobinar o atrasar y finalizar el vídeo, las mismas opciones que actualmente ofrecen los reproductores de vídeo doméstico.

La primera conclusión a la que se llega cuando se analiza el problema del VoD [1][2] es que su implantación requiere de equipos caros, por lo que sólo resultaría viable si existiese una demanda considerable, circunstancia que no se da en la actualidad en nuestro entorno. No obstante, se mantuvo en esta experiencia el VoD como caso de estudio, dado que su alto grado de exigencia, en todos los aspectos, permitiría obtener un buen conocimiento de los sistemas multimedia interactivos de cara a futuros proyectos. Igualmente se juzgó como muy interesante el hecho de que, para la elaboración de prototipos y demos, la naturaleza de la información que se ha de manejar (vídeos MPEG y AVI) permite ser manipulada de forma cómoda usando componentes comerciales.

La propuesta continúa mejorándose en la actualidad, por lo que ha de considerarse como una propuesta abierta y en evolución. Se han desarrollado, sobre las actuales redes IP, varios prototipos que han demostrado de manera parcial su viabilidad. Algunos de los elementos propuestos están pendientes de un estudio más riguroso a través de modelos teóricos y/o de implementaciones de prototipos sobre redes ATM [3].

2. El Adaptador Multimedia Interactivo (AMI)

Uno de los elementos fundamentales en la arquitectura propuesta es el Adaptador Multimedia Interactivo o AMI¹.

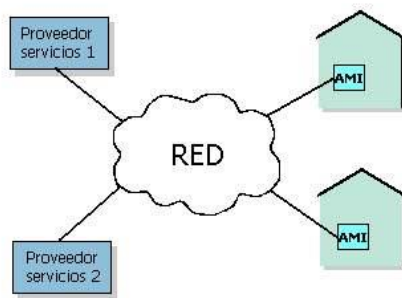


Figura 1. Estructura general de la red

La Figura 1 ilustra la situación del AMI como elemento intermediario entre la red y el usuario. Es un elemento de trascendental importancia, dado que será el soporte y en ocasiones la interfaz de los servicios ofertados al cliente.

Aunque el estudio de la infraestructura de red no ha sido un objetivo fundamental del proyecto, se han tratado de explorar las diversas alternativas que la técnica actual puede proporcionar para la implantación del servicio previsto. En los últimos sondeos realizados se han considerado como más viables las redes con un transporte ATM/SDH y un acceso basado en ADSL y/o FTTC/VHDL (*Fiber To The Curb/Very High Rate Digital Subscriber Line*). pudiendo proporcionar este último un acceso ATM a 25 o 52 Mb/s.

La Figura 2 representa los posibles servicios del AMI y algunas de sus características. Debido a la gran diversidad existente en los servicios multimedia, el AMI que se propone tiene una característica fundamental: la modularidad, tanto en los componentes *hardware* como

en los *software*. Será un dispositivo configurable en función de los servicios contratados por el cliente.

Figura 2.- Servicios y estructura del AMI

De cara a la red de acceso, se comporta como un módem xDSL (*x-Digital Subscriber Line*), más un conmutador ATM, mientras que de cara al usuario ofrecerá un cierto número de interfaces, cada una de ellas para un servicio determinado.

Entre otras funcionalidades pensadas para el AMI está la de actuar como elemento de una HAN (*Home Area Network*), bien únicamente como elemento encaminador/puente hacia la red externa, bien como elemento central a partir del que se construye la red doméstica.

El AMI deberá ser un elemento basado en un microprocesador o similar, con un sistema operativo capaz de integrar nuevas aplicaciones o módulos software, bien de forma local o bien de forma remota a través de, por ejemplo, la utilización de Java. En la propuesta se incluye un AMI que integra un ORB (*Object Request Broker*) de CORBA, un servidor Web y, por supuesto, un navegador de Internet que servirá de interfaz de usuario para los monitores y/o televisores a él conectados.

Posibles módulos *hardware* que se pueden incorporar al AMI son:

- Decodificadores MPEG de audio y vídeo.
- Receptor de TV analógica.
- Lector de tarjeta inteligente.
- Tarjeta de red *ethernet*

Posibles servicios soportados por este tipo de AMI son:

- Telefonía básica y digital.
- Televisión por cable digital – SDVB (*Switched Digital Video Broadcasting*).
- Videoconferencia y videotelefonía.
- Acceso a Internet.
- Servicio de Red Privada Virtual.
- Vídeo bajo demanda.

3. La experiencia de VoD realizada.

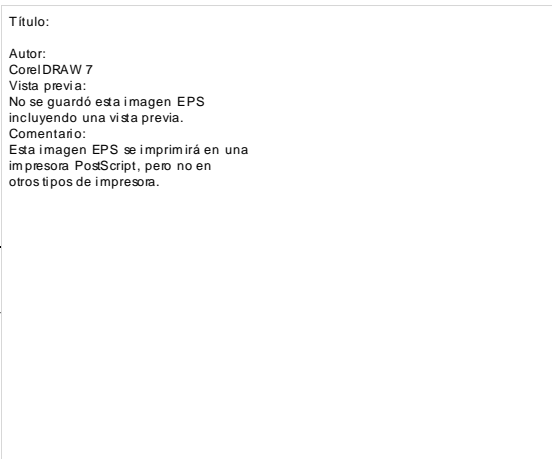
Para la realización de la experiencia se ha contado con una infraestructura muy básica que tendría que ser reforzada en el caso de una implantación comercial. No obstante, se han cubierto los objetivos planteados, que como se ha mencionado, no eran los de ofrecer un servicio real, sino comprobar que las tecnologías empleadas pueden resolver los requerimientos de este tipo de servicios.

3.1. Requisitos del servicio de VoD realizado.

A continuación se resumen las principales características y funcionalidades de la experiencia realizada.

3.1.1. Interfaz de usuario.

Estará basada en tecnología web. Para la navegación por las distintas opciones que permite el sistema, se generará un sitio web, donde se ubicará toda la información de los distintos servicios, organizada atendien-



do a diversos criterios. Se podrá consultar la cartelera por medio de un índice de películas, por una tabla con las más vistas, por sección, por género, etc.

3.1.2. Gestión de usuarios.

El sistema ofrecerá las interfaces adecuadas para que los usuarios se den de alta de forma semiautomática. El sistema recogerá los datos suficientes para identificar de manera adecuada al usuario y poder realizar las tareas de cobro de forma automática. El usuario podrá en todo momento acceder a la información que el sistema guarda sobre él. A tal fin, dentro del espacio web del sistema, existirán una serie de páginas por medio de las cuales se podrán realizar las consultas y modificaciones oportunas.

3.1.3. Adquisición y visualización del vídeo.

El servicio de VoD en sí comienza en el momento en que el usuario desea ver una película. En ese instante, consultará en su televisor la oferta de contenidos publicada por el sistema, y elegirá aquella que desee ver.

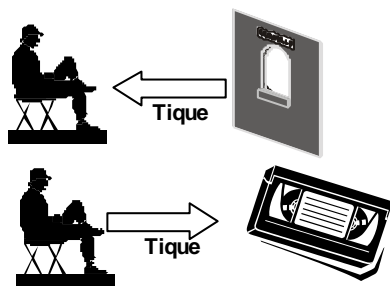


Figura 3.- Compra y utilización del tique.

Una vez elegido el título, deberá adquirir el derecho a verlo. Para ello, solicitará la compra de un tique al sistema (Figura 3), el cual consultará en su base de datos interna si el usuario puede ver esa película (limitaciones de censura, posibilidad de abono a un grupo de servicios, etc.).

A continuación se realizará una consulta interna al servidor de vídeo del que depende el usuario que solicitó el tique, para que determine el tiempo máximo que tardará el usuario en poder recibir el servicio solicitado. Finalmente se generará el tique, en el que se indicarán todos los datos del servicio a prestar, desde el precio, hasta el tiempo estimado que tardará el sistema, en función del estado de la red, en poder ofrecer el servicio. El tique contiene además un código que lo identifica ante el sistema.

El usuario recibe el tique, y comprueba que todo es de su agrado. En caso de no estar de acuerdo con algún aspecto del tique, como el precio o el tiempo que tardará en recibir la película, podrá decidir anularlo, con lo cual el servicio no será cobrado. Si decide cursarlo, el servidor de vídeo deberá encargarse de la reproducción de la película.

3.1.4. El servidor de vídeo.

Se implementará un sistema jerárquico distribuido de servidores de vídeo, de forma que se puedan hacer estudios y experimentaciones sobre la adecuada disposición de los elementos que lo componen en función de la ubicación geográfica y socioeconómica de los usuarios, a fin de lograr un adecuado dimensionado del sistema.

En el caso de no tener almacenado el servicio a reproducir, el servidor de vídeo lo solicitará al servidor de ficheros del que dependa, que se encargará de transferirlo al servidor de vídeo que lo solicitó, realizando para ello las transferencias oportunas.

Una vez que el servidor de vídeo tenga en su poder el fichero que almacena la película, se le indicará al cliente, que a partir de ese instante podrá controlar la reproducción del mismo modo que lo haría con un reproductor de vídeo doméstico.

En un caso real los requisitos exigibles a los distintos elementos dependerán del nivel que ocupen en la jerarquía. Un servidor de ficheros situado en la parte baja del escalafón, es decir, cerca del usuario, no deberá tener una capacidad de almacenamiento muy elevada, aunque por el contrario deberá ofrecer tiempos de respuesta y de transferencia mínimos. Los dispositivos de almacenamiento internos que utilizará serán rápidos, aunque de no muy elevada capacidad. En el caso del servidor de ficheros raíz no importará tanto la velocidad con que pueda recuperar la información, sino que primará el criterio de maximizar la capacidad de almacenamiento, dado que contendrá todos los contenidos que pueda ofrecer el sistema. En el caso tratado en la experimentación todas las máquinas servidoras de vídeo podrán ser iguales.

3.1.5. Gestión del sistema

Se dispondrá de un sistema de gestión integrado que permita administrar los servicios, los elementos y los usuarios.

En lo relativo a la gestión de servicios, se podrán insertar nuevas películas, modificar datos como la sección o el precio de películas ya ofertadas, o bien retirar alguna película.

En lo relativo a la gestión de los elementos, se dispondrá de un panel de gestión que recoja las distintas alarmas generadas por los equipos que conforman el sistema, desde los servidores de ficheros al equipamiento del usuario. Cada uno de los componentes software del sistema comunicará al panel de alarmas cualquier anomalía detectada.

3.2. Plataformas y herramientas.

Para la realización de la experiencia se ha contado con:

- PCs con windows95-98 / NT.
- Internet Explorer y Netscape.
- Orbix y Orbix web de Iona Technologies.
- TAU (SDT y ORCA) de Telelogic.

- JDK1.1 de Sun Microsystems.
- Visual C++ 5.0 de Microsoft.
- Borland Jbuilder.
- Microsoft Access.
- Servidores web:
 - IIS de Microsoft.
 - Java WebServer 1.1 de Sun.

3.3. Metodología.

Se ha utilizado el siguiente ciclo de vida:

- **Análisis Orientado a Objetos.** Se han empleado las herramientas de TAU/ORCA, las cuales permiten realizar un modelado parcial del sistema en UML.
- **Diseño arquitectural.** La funcionalidad del sistema se ha descompuesto en distintos subcomponentes distribuidos, en su mayoría materializables como objetos, que se integran gracias a tecnología web, Java y CORBA. Se definen las interfaces IDL de los componentes CORBA y los escenarios de interoperación.
- **Diseño detallado.** En función de la naturaleza del subcomponente se elige la tecnología de implementación adecuada y se procede a su diseño pormenorizado.
- **Implementación.** La mayor parte de los componentes CORBA se han desarrollado usando Java y SDL (el código C generado automáticamente se ha compilado con Visual C++). El acceso a las bases de datos se ha realizado con JDBC. Los componentes que no se integran a través de CORBA se han desarrollado básicamente en Java. Sobre el *Java WebServer* de Sun se han desarrollado varios *javaservlets*.
- **Pruebas.** Se han seguido, excepto para los componentes desarrollados mediante técnicas formales, métodos tradicionales. Para los componentes desarrollados en SDT se han usado las herramientas de validación y verificación de SDT.

3.4. Diseño arquitectural del sistema de VoD.

Se han identificado tres planos de aplicación:

- **Plano de usuario.** En este plano se engloban las interfaces de aquellos componentes necesarios para la visualización de una película por parte del usuario.
- **Plano de control.** En este plano se incluyen las interfaces de los objetos necesarios para supervisar el funcionamiento del sistema. Se encargan de monitorizar y responder, en la medida de lo posible, a las alarmas generadas. Otra función importante dentro del sistema y que se engloba en este plano es el llamado *control de 'caching'*, es decir, la distribución automática de las películas en los diferentes servidores.
- **Plano de gestión.** En este plano se encuentran los objetos encargados de facilitar la gestión del sistema. Se contemplan operaciones como el cambio de parámetros en los objetos, obtener las estadísticas de la red, parar y arrancar objetos. También se encuentran en este plano las interfaces que permi-

ten realizar la gestión de los usuarios y la tarificación.

3.4.1. Plano de usuario

La Figura 4 representa una posible distribución de los objetos más significativos con interfaces en el plano de usuario. Esta distribución, en la que se ha centralizado el sistema de gestión de bases de datos, se realiza a modo de ejemplo, debiéndose elegir la más adecuada a cada implementación. La estructura de componentes elegida permite distribuir con facilidad la base de datos.



Figura 4. Objetos del plano de usuario.

Grosso modo el funcionamiento de este plano es el siguiente:

- El usuario del servicio de VoD, utilizando el *browser* del TV-PC accede al servidor web ubicado en la *máquina información*, para elegir y comprar el tique de la película.
- Los objetos *Autenticación usuario* y *Atención usuario* realizan la funcionalidad anterior consultando a los objetos de la *máquina de gestión*. La máquina de gestión contiene al *sistema de gestión de bases de datos*, que, a su vez, contiene metainformación de las películas y de la información relativa a los usuarios.
- Los objetos de la máquina de gestión se encargan de informar de si el usuario está autorizado a realizar la petición y de dar toda la información relativa a la película (disponibilidad, tiempo estimado para el visionado...).
- Una vez comprado el tique el usuario se dirige de nuevo al objeto *Atención usuario* para comenzar el visionado de la película. Este objeto se pone en contacto con el objeto *Servidor AMI*, indicándole que puede iniciar la transferencia de la película, de un determinado servidor de películas.
- La transferencia del vídeo codificado es responsabilidad del objeto *Servidor AMI* y se realiza a través de *sockets*. En la actualidad se está a la espera de poder probar las funcionalidades de nuevas herramientas como son JavaMedia y OrbixMX para la realización de este tipo de transferencias de flujos de datos.

3.4.2. Plano de control

En este plano (Figura 5) se han identificado los objetos correspondientes a la infraestructura de servidores de

vídeo jerárquicos y a la gestión del panel de alarmas del sistema. Los objetos de control de la distribución de las películas en los servidores de vídeo se han ubicado, por sencillez, en la *máquina gestión*. En una aplicación real, de cara a mejorar la eficiencia y fiabilidad del sistema, lo más adecuado es que se incluyan en una máquina del subsistema de los servidores de vídeo.

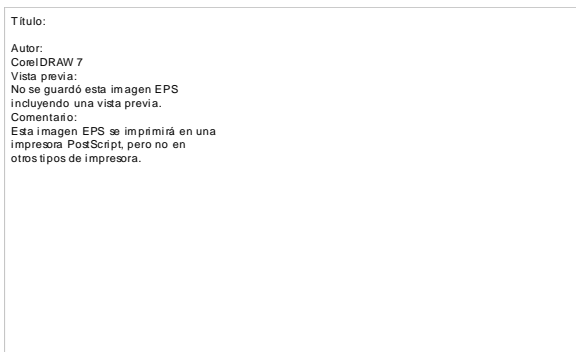


Figura 5.- Objetos del plano de control.

Las transferencias de películas entre servidores, al igual que en el plano de usuario, no se realizan utilizando CORBA, sino que se hacen a través de *sockets*.

3.4.3. Plano de gestión

Uno de los objetivos que se ha impuesto para la gestión del sistema es que ésta se realice mediante interfaces web. Así, en la máquina de gestión (Figura 6), aparece un servidor web que se encargará de soportar las citadas tareas. De esta manera la gestión se podrá hacer desde un navegador de cualquier máquina autorizada.

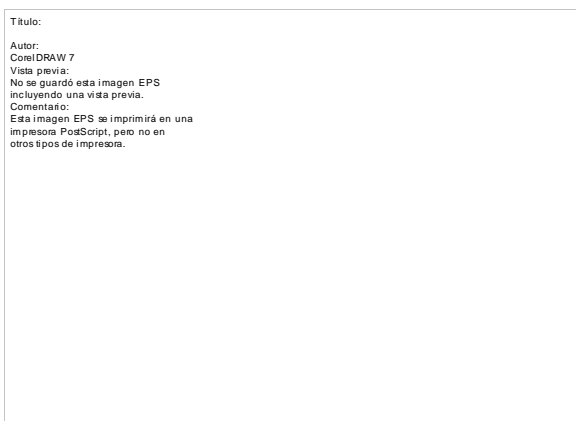


Figura 6.- Objetos del plano de gestión.

La tarificación no se ha incluido en la máquina de gestión de forma meramente testimonial. Su ubicación en un sistema real dependerá de los requisitos no funcionales impuestos.

4. Integración con SDL.

Algunos de los componentes CORBA identificados anteriormente (Gestión servidor, Control servidor y Servidor AMI) se han realizado en SDL [4]. Evidentemente se han escogido aquellos que más se adecuan a ser resuelto con este lenguaje.

En el proyecto OTELSO [6], se realizó una experiencia que sirvió para que Telelogic incorporara parcialmente en la herramienta SDT (*SDL Development Tool*), la funcionalidad de integrar de manera automática aplicaciones desarrolladas con SDL sobre el ORB de Orbix.

En la actualidad con SDT se pueden desarrollar aplicaciones CORBA que sean sólo servidores, no admitiendo el desarrollo de la parte cliente. La citada funcionalidad está presente únicamente en la versión UNIX de las herramientas y no en la de Windows. Esta circunstancia ha obligado a migrar a Windows, al tiempo que se han completado ligeramente, las aplicaciones de integración manual SDT-CORBA desarrolladas para UNIX en OTELSO a Windows.

4.1. Integración SDL-CORBA

La integración puede realizarse de dos maneras:

- **Aproximación orientada a CORBA.** A partir de la interfaz IDL se desarrolla una aplicación SDL que implementa el servicio definido.
- **Aproximación orientada a SDL.** Se arranca de una especificación del sistema en SDL, y se identifican un conjunto de interfaces IDL. El número de interfaces dependerá de la distribución deseada para el sistema. A partir de los subsistemas identificados se generan sus implementaciones CORBA.

En el caso de estudio, en base a la metodología empleada, se ha utilizado la aproximación orientada a CORBA.

4.2. Correspondencia IDL-SDL

El primer paso del proceso arranca de las definiciones IDL obtenidas en la fase de diseño arquitectural como parte del *contrato* que debe cumplir la aplicación SDL que se quiere construir.

Con objeto de automatizar el proceso, en OTELSO se estableció una correspondencia IDL-SDL siguiendo la recomendación de OMG [7]. Se ha utilizado la siguiente correspondencia (*mapping*) [8][9] entre los dos lenguajes:

Tabla 1

IDL	SDL
<i>Module</i>	<i>Block Type</i>
<i>Interface</i>	<i>Process Type</i>
<i>Operation</i>	
Síncrona	<i>External RPC</i>
Asíncrona	<i>Signal</i>
<i>Parameter</i>	
<i>IN</i>	<i>IN</i>
<i>OUT</i>	<i>OUT</i>
<i>IN/OUT</i>	<i>OUT</i>
<i>Attribute</i>	<i>Variable</i>
	<i>External RPC</i>
<i>Constant</i>	<i>Synonym</i>
<i>Typedef</i>	<i>Synotype</i>
<i>Enums</i>	<i>Newtype</i>
<i>Struct</i>	<i>Struct</i>
<i>Union</i>	<i>Struct</i>
<i>Sequence</i>	<i>String</i>
<i>String</i>	<i>Charstring</i>
<i>Array</i>	<i>Array</i>

Las correspondencias entre los tipos básicos de IDL y SDL son las de la Tabla 2.

Tabla 2

IDL	SDL	Syntype of
Long	CORBA_long	Integer
Short	CORBA_short	Integer
unsigned long	CORBA_unsigned_long	Integer
unsigned short	CORBA_unsigned_short	Integer
Double	CORBA_double	Real
Float	CORBA_float	Real
Char	CORBA_char	Character

El resto de conceptos pendientes (*any*, *octet*, *exceptions*, *raises* y *context expressions*) de establecer una correspondencia están bajo estudio hasta que se logre automatizar su proceso de traducción.

4.3. Creación de una aplicación SDL

Como ya se ha descrito, el desarrollo de una aplicación en SDL usando la alternativa que se ha denominado aproximación CORBA tiene como punto de partida la especificación IDL de la aplicación. Por tanto, el primer paso debe ser convertir dicha especificación a un sistema SDL equivalente, usando la correspondencia establecida en el apartado 4.2.

Mediante esta correspondencia, se obtiene un sistema SDL que contiene los siguientes elementos:

- **Un *package* con la interfaz de servidor** que contiene las definiciones para los tipos básicos, *synonyms*, *newtypes*, señales y procedimientos remotos, y los tipos de los parámetros asociados correspondientes a las definiciones de los tipos básicos, constantes, tipos compuestos, atributos y operaciones de la interfaz IDL.
- **Un sistema SDL** que contiene la definición de los *block type*, los *process type*, y referencias a los procedimientos remotos, que se corresponden a los módulos, interfaces y operaciones síncronas.

Si el sistema SDL es una aplicación cliente/servidor, es necesario generar también **un *package* con la interfaz de cliente** a partir de la especificación IDL del componente al que se van a solicitar servicios.

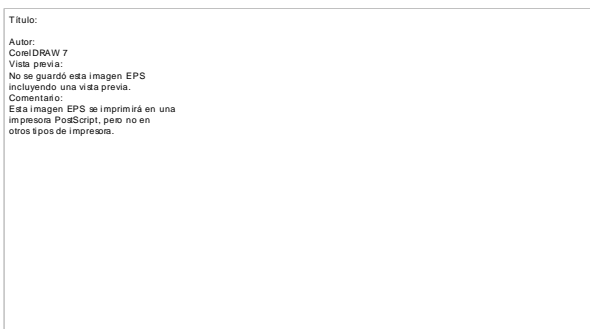


Figura 7.- Equivalencia IDL-SDL

En el sistema SDL, a través de la *use clause* del lenguaje, deben incluirse los *packages* correspondientes con las interfaces servidor ofrecidas y con las interfaces cliente que usa.

El sistema SDL (Figura 7) creado a partir de las interfaces cliente/servidor IDL es solamente un esqueleto, ya que la información relativa a la interacción entre bloques e instancias de procesos no puede ser deducida a partir de la especificación IDL, ni, por supuesto, la información relativa al comportamiento deseado. Por lo tanto, el siguiente paso es completar la estructura estática de la especificación SDL y describir el comportamiento de los tipos proceso y de los procedimientos.

El sistema SDL completo, incluyendo el acceso a otros servidores, debe ser analizado y simulado para verificar si satisface la funcionalidad requerida. Finalmente, se crea un ejecutable que pueda integrarse con la plataforma CORBA. Este ejecutable se obtiene compilando y encadenando el código de las diferentes partes implícadas en la integración de ambos mundos (Figura 8).

De un lado, usando la generación automática de código del entorno de desarrollo para SDL, se obtiene el código que implementa el servicio ofrecido por el componente.



Figura 8.- Generación del código del objeto

De otro lado, es preciso generar el código necesario para que el ORB pueda gestionar la interacción entre los clientes y la aplicación, como también entre esta aplicación y los servidores a los cuales ella tenga que acceder (lado servidor y lado cliente respectivamente). Este código es generado de modo automático por el compilador IDL del ORB que se utilice, generalmente proporcionado por la propia implementación de la plataforma CORBA.

El código generado por el sistema SDL no puede ser directamente invocado por el ORB; es necesario codificar un adaptador o *wrapper* que permita la integración de ambas partes. Este adaptador contiene, desde el punto de vista del ORB, la implementación del componente, aunque realmente sólo es una interfaz del sistema SDL en el que se ha realizado la implementación. La estructura del adaptador, así como las características relativas al modo concreto en que la integración se lleva a cabo se describen en el siguiente apartado.

4.4. Implementación de la integración

Las operaciones concretas para llevar a cabo la integración CORBA-SDL dependerán, en una gran medida, del entorno y las herramientas que se utilicen. En nues-

tro proyecto, como ya se ha mencionado, se usaron las herramientas Orbix para Windows y SDT 3.2.

Uniendo todas las piezas, la aplicación SDL desarrollada usando SDT esta formada por las siguientes partes (Figura 9):

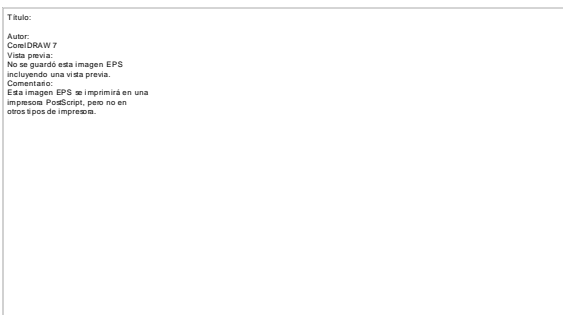


Figura 9.- Componentes de un objeto

- El sistema SDL. Contiene la implementación del componente que se ha desarrollado. Estará formado por un conjunto de instancias de procesos, algunos de los cuales interactuarán con el entorno para ofrecer ciertos servicios (lado servidor), y que solicitan ciertos servicios a otros servidores (lado cliente).
- Un adaptador (*wrapper*) C++. Contiene la implementación del componente desde el punto de vista del ORB. Desde el punto de vista de SDL constituye su entorno. También incluye el código necesario para solicitar servicios a otros componentes.
- Las funciones de entorno. Proporcionan la funcionalidad requerida para la interacción entre el sistema SDL y el C++ *wrapper*.
- El código específico del ORB. En unión al ORB, gestiona la interacción entre los clientes y los servidores.

El *kernel* que se usa para generar la aplicación SDL crea un solo proceso en Windows que contiene al sistema SDL, a las funciones del entorno, y al C++ *wrapper*. Esto genera algunos problemas al tratar las operaciones síncronas solicitadas por los clientes: cuando se recibe la invocación de una operación, el adaptador C++ debe enviar una señal al sistema SDL y no puede retornar el control de la ejecución hasta haber recibido la correspondiente respuesta; pero, como no hay procesamiento paralelo, el sistema SDL no puede progresar y, por tanto, no puede ejecutar la transición que implementa la operación.

Para resolver este problema se ha utilizado una implementación con *threads*. Para cada operación solicitada, sea síncrona o asíncrona, desde o hacia la aplicación SDL, se crea un nuevo *thread* que se ejecuta en paralelo con el sistema SDL, que es el *thread* principal. La

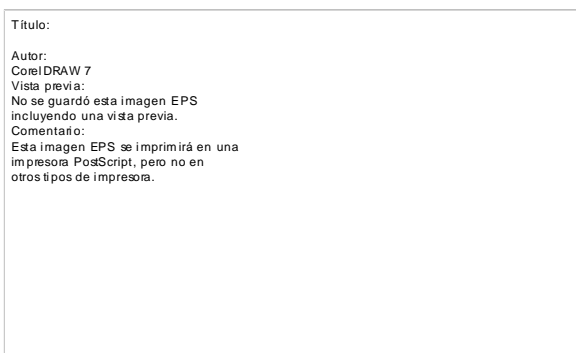


Figura 10 muestra la estructura de la solución utilizada.

Figura 10. La interfaz SDT-Orbix

Cuando un cliente solicita una operación a una aplicación SDL, el ORB invoca al método correspondiente del C++ *wrapper*, creándose un nuevo *thread* que se ocupa de introducir en el sistema SDL una señal con los parámetros correspondientes. Si es una operación síncrona se queda esperando a que el sistema SDL le envíe la señal de respuesta, de la que extrae los parámetros que porta, entregándoselos al ORB. Una vez hecho esto el *thread* muere.

En muchos casos, será necesario que el sistema SDL que implemente una aplicación actúe también como cliente, solicitando servicios a otros componentes en el entorno distribuido. El sistema SDL invoca una operación enviando una señal al entorno. Por cada señal enviada se crea un nuevo *thread*. Tanto en el caso de las operaciones síncronas como de las asíncronas, el *thread* invoca el método que implementa la operación. Si la operación es asíncrona no son necesarias más acciones y se termina el *thread*. Si se trata de una operación síncrona, los parámetros obtenidos por la respuesta recibida desde el ORB, se envían al sistema SDL, y se termina el *thread*.

Si diferentes clientes solicitan la misma operación síncrona sobre el mismo objeto de la aplicación SDL, serán varios *thread*, uno por cada operación invocada, los que se ejecutarán en paralelo. Para que cada *thread* reciba su correspondiente señal de respuesta y no se vea afectado por “condiciones de carrera”, es necesario enviar la identificación del *thread* que trata cada operación como un parámetro adicional de la señal que se envía al sistema SDL.

Finalmente, las funciones proporcionadas por SDT 3.2 para manejar las instancias de las señales *-xGetSignal*, *xReleaseSignal*, *SDLOutput*, etc- no son *multithread safe*. Por todo ello se hace necesario implementar fuertes medidas de sincronización entre el C++ *wrapper* y las funciones del entorno para evitar que diferentes *threads* ejecuten la misma función simultáneamente.

5. Conclusiones

La experiencia demuestra la viabilidad de la integración de las tecnologías Web, CORBA y SDL en el desarrollo de proyectos telemáticos y en particular de servicios multimedia interactivos. La sinergia entre las tres tecnologías ha sido puesta de manifiesto a lo largo de la realización del proyecto.

Integrar SDL como herramienta de desarrollo de componentes CORBA se hace en la actualidad, con ayuda de SDT, de una forma natural y, en sistemas UNIX, automática. Telelogic tiene previsto extender la integración automática a todos los sistemas operativos que soporta en breve plazo.

Existen hoy en día soluciones técnicas para abordar la realización de servicios como el de VoD, sólo es nece-

sario esperar a que la demanda social sea suficiente para que puedan ser implantados de manera masiva.

Broadband Services and Networks. LNCS 998, Springer-Verlag 1995.

Agradecimientos

A Gonzalo León Serrano que nos brindó la posibilidad de colaborar en los proyectos MEDAS y OTELSO. A todos los compañeros que participaron en el desarrollo de estos proyectos por sus opiniones, consejos y sugerencias. A la Comisión Interministerial de Ciencia y Tecnología (CICYT) y al Centro para el Desarrollo Tecnológico Industrial (CDTI) que financiaron parcialmente estos proyectos. A Telelogic AB (de Suecia) y a Telefónica I+D por la confianza depositada en el grupo de trabajo de DIATEL, que nos ha permitido, en los últimos años, continuar con el desarrollo de servicios para Red Inteligente, mediante el empleo de tecnologías SDL y CORBA, colaboración que ha estimulado el interés por aplicar lo aprendido a otras áreas como la de los servicios multimedia interactivos.

A todos los estudiantes, hoy graduados, que colaboraron en la realización de este proyecto.

Referencias

- [1] L.A. Rowe, D.A. Berger, and J.E. Baldeschieler. “*The Berkeley Distributed Video-on-Demand System*”, Multimedia Computing - Proceedings of the Sixth NEC Research Symposium, Ed. T. Ishiguro, SIAM, 1996
- [2] M. Podgorny, G.C. Fox. “*Video on demand Technologies and demonstrations. Final report*”. Northeast Parallel Architectures Center. Syracuse University. <http://trurl.npac.syr.edu/rlvod/report/report.html>. 1997.
- [3] ATM Forum. “*Audiovisual Multimedia Services: Video on Demand Specification*”. v1.1, af-saa-0049.000. Mar, 1997.
- [4] ITU-T. “*SDL Specification and Description Language. Z.100*”. 1996.
- [5] J. Carracedo, G. León, C. Ramos, R. de Diego et al. “*An industrial experience on SDL introduction in a conventional development life cycle*”. SDLForum93. Darmstadt. Alemania. 1993.
- [6] J. Carracedo, C.Ramos, R. De Diego, C.González et al. “*Introducing SDL in the development of CORBA-compliant applications*”. SDLForum97. Paris. 1997.
- [7] OMG-X/OPEN. “*Common Object Request Broker: Architecture and Specification*”. Working Paper OMG TC Document 91.12.1, Object Management Group, Framingham, MA (USA), September 1992.
- [8] Humboldt University, Berlin. “*Formal language mapping from CORBA IDL to SDL-92*”. Contribution to ITU Q6 Expert Meeting, Lutterworth. 1995.
- [9] A. Olsen and B.M. Jorgensen. “*Using SDL for targeting services to CORBA*”. In A. Clarke, M. Campolargo and N. Karatzas, editors, Bringing Telecommunication Services to the people 3rd International Conference on Intelligence in

El papel de los servicios de intermediación en el contexto del comercio electrónico.

Víctor A. Villagrà[®], José I. Moreno[®], Julio Berrocal[®], Juan I. Asensio[®]
{villagra,berrocal,jasensio}@dit.upm.es, jmoreno@it.uc3m.es

[®] Dpto. Ing. Telemática (DIT-UPM) ETSI Telecomunicación. Madrid, ESPAÑA. [®] Investigador visitante de la Univ. de Valladolid	[®] Area de Ingeniería Telemática Universidad Carlos III Leganés, ESPAÑA
---	---

Abstract

This paper presents the so-called electronic brokerage systems, which are supposed to play a key role in the context of the future electronic commerce marketplace, where the huge number of offers and demands will difficult the matching between them. The paper describes the main features of this electronic brokerage service, and the most important conclusions gained from the study and development of a brokerage system prototype made within the ABS (Architecture for Information Brokerage Service) project, which belongs to the ACTS European Research Program.

1. Introducción

El rápido desarrollo y disponibilidad de aplicaciones de comercio electrónico y servicios de información sobre las infraestructuras de comunicación existentes ha llevado a la necesidad de unos nuevos servicios, denominados servicios de intermediación electrónica (más conocidos por su nombre original en inglés, *electronic brokerage*), que tratan de acercar los dominios de los clientes y los suministradores en el ámbito del comercio electrónico. Estos servicios deberán evitar, o al menos disminuir grandemente, el papel proactivo de los usuarios a la hora de realizar las búsquedas de la información o servicios solicitados.

Estos nuevos servicios de intermediación electrónica deberán ocupar el papel que tienen los actuales mercados en el comercio tradicional, es decir, un lugar/servicio que permita de una manera eficiente, escalable e integrada:

- Que los clientes encuentren a los suministradores
- Que los suministradores encuentren a los clientes
- Que se puedan realizar transacciones comerciales completas, con pago incluido y de una manera segura.

En este artículo se expone la especificación realizada de un servicio de intermediación electrónica realizada en el ámbito del proyecto ABS, perteneciente al programa ACTS, el cual ha sido prototipado y validado en pruebas piloto internacionales durante 1997.

2. Servicio de Intermediación Electrónica

La base del comercio actual es el intercambio de bienes y servicios. Cuando el cliente necesita algo, crea una demanda en el mercado. Por otro lado, los proveedores proporcionan bienes y servicios, creando ofertas en el mercado. Para facilitar el encuentro de ambas partes, se requiere una "plataforma" que actúe como el lugar de encuentro entre ofertas y demandas.

Actualmente existen servicios similares aplicados a la información en Internet. Son los denominados "Portales de Internet", basados en servidores WWW que ofrecen las siguientes funcionalidades típicas:

- Catalogación de la información: en base a una estructura del conocimiento determinada, el portal ofrece enlaces a los proveedores de contenido relacionados con dicho tema.
- Servicio de Búsqueda: ya sea en el catálogo de títulos de ofertas disponibles en el portal, o una búsqueda a lo largo de toda la Internet.
- Servicios personalizados: es posible personalizar la estructura del portal para que incluya los servicios de información que el usuario elija, desde la previsión del tiempo en un determinado área, hasta la información de deportes de un determinado equipo de fútbol.

Si bien estos portales han evolucionado en los últimos años para ofertar unos servicios más

sofisticados, existen todavía deficiencias en dichos servicios. Entre ellas podemos citar las siguientes:

- ❑ El intermediario no asume ninguna responsabilidad por la información que proporciona ni asegura la calidad. Una consecuencia de ello es que gran parte de los resultados de los servicios de búsqueda son erróneos y no se corresponden a la información requerida.
- ❑ El intermediario no es capaz de proporcionar un valor añadido sobre la información de sus proveedores de contenidos, como por ejemplo puede ser la estructuración de información de diversos proveedores en base a una determinada estructura de conocimiento. De esta manera, el intermediario puede ofertar y proporcionar conceptos que necesitan una combinación de distintos proveedores de contenido, como puede ser un paquete turístico (billetes de avión más reserva de hotel más guías turísticas sobre el destino más propuestas de excursiones relacionadas).
- ❑ Los portales actuales son de aplicación universal, sin existir el concepto de portal especializado, dándose una situación de competencia entre portales. En un ámbito en el que la información puede llegar a tener volúmenes gigantescos, es necesario la cooperación federada entre portales especializados en subconjuntos (semánticos, geográficos o administrativos) de la información.
- ❑ Los portales se limitan casi exclusivamente a ofertar información, pero no servicios proporcionados por terceros. Esto lleva a que el usuario siempre tenga que encontrar al proveedor de servicios para poder conectarse a él para obtener el servicio deseado.

Es por ello por lo que surge el concepto de Intermediación Electrónica (o “*Electronic Brokerage*” en terminología anglosajona). Basándose en la utilización de las nuevas tecnologías de la información, deben ser capaces de proporcionar un servicio que facilite y organice la relación entre las ofertas y las demandas. Contará con dos clases de usuarios (usuarios finales y proveedores de bienes) y ofrecerá las siguientes funcionalidades a dichos usuarios:

- ❑ Bajos costes de búsqueda para ambos actores. La tarea principal de los intermediarios electrónicos será la búsqueda de una o varias ofertas que satisfagan una demanda o viceversa. Es necesario limitar la complejidad de dichas búsquedas haciéndolas lo más transparentes posibles para los usuarios.

- ❑ Estructuración de las fuentes de información: la eficiencia en las búsquedas de emparejamientos entre las ofertas y las demandas se basará en una estructuración adecuada de éstas en el intermediario. Esta estructuración permitirá a su vez una capacidad de navegación a través de las ofertas disponibles para los usuarios.
- ❑ Combinación de la información: el intermediario será capaz de combinar distintas ofertas o demandas para atender con una mayor calidad una petición de un usuario.
- ❑ Presentación: cuando las búsquedas nos proporcionen múltiples posibles soluciones, será necesario categorizar éstas para facilitar al usuario la toma de decisión.
- ❑ Acceso uniforme: independiente del tipo de la información, el intermediario proporcionará una visión uniforme de distintas ofertas y demandas heterogéneas.
- ❑ Fiabilidad: el intermediario velará por la calidad de las ofertas y demandas que maneja internamente, manteniéndolas actualizadas y preservando sus aspectos legales.
- ❑ Confidencialidad: cuando sea necesario el intermediario no desvelará la identidad de los distintos usuarios que intervienen en el servicio.

En las secciones siguientes, se presenta las características más importantes de un servicio de intermediación electrónica desarrollado en el ámbito del proyecto ABS (*Architecture of a Brokerage Information Service*) del programa europea ACTS, dónde se ha realizado un profundo estudio del servicio de intermediación electrónica, tanto desde el punto de vista del modelo de negocio en el que se sitúa este servicio, como de la especificación en detalle de los servicios que se deben prestar. Estos conceptos fueron validados con la realización de un prototipo utilizado por un conjunto de usuarios y proveedores de contenidos y servicios seleccionados que permitieron valorar los resultados del trabajo.

3. Definición del servicio de intermediación electrónica

Los principales servicios proporcionados por el intermediario electrónico son los siguientes:

- ❑ Para usuarios finales:
 - ◆ Capacidad de **suscribirse y acceder** al sistema, pudiendo especificar un perfil personal que el intermediario pueda consultar para mejorar el servicio proporcionado a ese usuario, en base al

filtrado de resultados o en base al refinamiento de las peticiones de ese usuario en base a su perfil personal.

- ◆ Capacidad de definir una **petición de búsqueda** de información. El papel del intermediario para ofertar este servicio será seleccionar los proveedores de contenido que sean capaces de ofrecer resultados totales o parciales a esa petición, y trasladar esa petición (total o parcialmente) a dichos proveedores de contenido. El valor añadido que ofrece el intermediario en este servicio es garantizar que los resultados son adecuados a la petición de los usuarios. Para ello, el intermediario debe conocer previamente el tipo de información manejada por los proveedores de información para ser capaz de identificar de la manera más exacta posible aquellos que pueden proporcionar respuestas útiles a la pregunta.
 - ◆ Capacidad de **navegar** en la estructura del conocimiento del intermediario. Este servicio se pone a disposición de los usuarios como una ayuda para realizar las peticiones de búsqueda de una manera más precisa. De esta manera, el usuario podrá seleccionar los conceptos de conocimiento relacionados con su petición de manera que el intermediario pueda seleccionar de una manera más precisa los proveedores de información involucrados en su petición.
 - ◆ Capacidad de visualizar y acceder a **ofertas** almacenadas directamente por los suministradores en el propio intermediario. Esta funcionalidad permite que los usuarios puedan ver las denominadas “ofertas” que los proveedores de información han registrado previamente asociadas a algún concepto de conocimiento. Este servicio es el típico servicio de paginas amarillas que es ofertado actualmente por muchos portales de Internet.
- Para los suministradores.
- ◆ Capacidad de **suscribirse y acceder** al sistema, pudiendo registrar un perfil personal del suministrador con datos globales acerca de la información que está proporcionando (por ejemplo, localización geográfica, idioma de la información, capacidades de comercio electrónico, etc.)
 - ◆ Capacidad de definir en el intermediario los denominados “**recursos**” del suministrador. Estos “recursos” representan el conocimiento del suministrador que éste registra en el intermediario, de manera que el intermediario pueda seleccionar a este

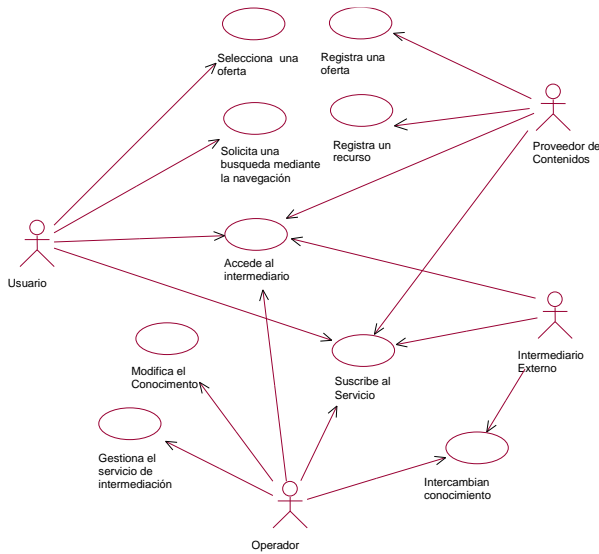
proveedor de información como candidato a resolver una (o parte) determinada petición de búsqueda de los usuarios que les será trasladada. De esta manera, un suministrador puede registrarse en el intermediario como manejador de información relacionada con “Turismo”, “Guías de Viaje de América del Sur” o “Suministros Textiles”.

- ◆ Capacidad de **navegar** en la representación del conocimiento del intermediario. Este servicio se pone a disposición de los suministradores como una ayuda para que puedan encontrar de una manera rápida aquellas las áreas de conocimiento del intermediario en las que puedan registrar un recurso o una determinada oferta.
- ◆ Capacidad de almacenar en el intermediario **ofertas** específicas de este suministrador. Estas ofertas son informaciones específicas relacionadas con un área de conocimiento que el suministrador ha decidido almacenar en el intermediario para que los usuarios puedan acceder a ellas sin tener que definir una petición de búsqueda. De esta manera, un usuario que navegue por el área de conocimiento “Paquetes turísticos”, si mira en el concepto “País Destino = Grecia”, podrá visualizar a su vez determinadas ofertas que proveedores de información pueden haber registrado sobre próximos paquetes turísticos a Grecia.

Adicionalmente, deben existir funcionalidades para gestionar y administrar este servicio de intermediación por parte del operador del servicio, que incluso debe ser capaz de modificar el conocimiento del intermediario, debido a la adición de algún proveedor de información en un área de conocimiento no contemplada hasta el momento por el intermediario.

Un aspecto importante a resaltar que no se traduce en un servicio directo a usuarios o suministradores es la capacidad de **federación** de intermediarios. El nivel de federación definido es el de compartición de conocimiento, de manera que un intermediario pueda conocer el conocimiento manejado por otro intermediario (es decir, el otro intermediario tiene suministradores capaces de proporcionar respuestas en dichas áreas conocimiento). Así, si a un intermediario le llega una petición de búsqueda de sus usuarios relacionada con un área de conocimiento que él no maneja, puede redirigirla a otros intermediarios que manejen dicha área de conocimiento y que puedan proporcionar resultados a esa búsqueda. Para que esta funcionalidad de federación pueda tener lugar, es necesario facilitar un servicio entre intermediarios para que intercambien su conocimiento.

Como resumen, en la siguiente figura se muestran los distintos servicios ofertados por el intermediario, utilizando la notación de “Use Cases” del lenguaje de modelado UML:



4. Representación del Conocimiento

Una de las características más importantes del servicio de intermediación es la representación del conocimiento que éste maneja. La adecuación de la información de los suministradores a esta representación del conocimiento es la clave para que las preguntas realizadas por el usuario puedan tener respuestas adecuadas a sus requisitos. Es responsabilidad del servicio de intermediación garantizar que las declaraciones de “recursos” (tipo de información que proporcionan) sean coherentes con la representación del conocimiento que posee el intermediario, de manera que asegure la calidad de las respuestas proporcionadas a un usuario ante una petición de búsqueda.

La representación del conocimiento que maneja el intermediario debe tener las siguientes características:

- ❑ Adecuada a los distintos campos de conocimiento de los proveedores de información, para que éstos puedan encontrar áreas de conocimiento adecuadas para registrar su información.
- ❑ Fácil de visualizar por los usuarios. Los usuarios podrán navegar por esta estructura de conocimiento para encontrar las áreas a las que se aplica sus peticiones, y de esta manera definir más precisamente sus peticiones al intermediario.
- ❑ Fácil de extender para que pueda acomodar a nuevos proveedores de información que manejen información en áreas de conocimiento

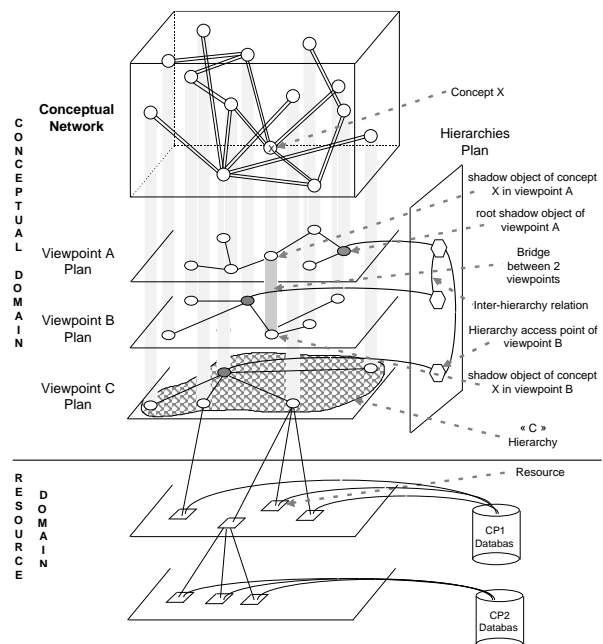
no incluidas inicialmente en la estructura de conocimiento del intermediario.

- ❑ Especificable de alguna manera que permita su intercambio entre intermediarios para que un intermediario pueda “importar” la estructura de conocimiento manejada por otros intermediarios y así redirigirles peticiones de sus usuarios en un entorno de intermediarios federados.

La estructuración del conocimiento es el campo que estudian las denominadas “ontologías”, que tratan de especificar la estructura del conocimiento de diversas áreas. En el proyecto ABS se utilizan una simplificación de estas ontologías denominadas “viewpoints” o proyecciones de las entidades de conocimientos global (“Conceptual Network”) sobre un determinado área de conocimiento. De esta manera, por ejemplo, el concepto “Ciudad” puede formar parte del “viewpoint” turismo, en el sentido de ciudad objeto de un viaje, o del “viewpoint” geografía, en el sentido de ciudad situada en un país. Estas proyecciones de un concepto sobre un determinado área de conocimiento o “viewpoint” se denominan “shadow object”.

Esta estructuración permite ofrecer a los usuarios la capacidad de poder navegar en un determinado “viewpoint”, que seleccionará previamente de acuerdo al área de conocimiento que él crea implicada en su búsqueda, hasta encontrar el concepto o conceptos de este “viewpoint” que represente su petición de búsqueda de información. Existe también la posibilidad de cambiar de “viewpoint” durante la navegación para trasladarse a otra área de conocimiento, utilizando puentes (“bridge”) entre objetos de distintos “viewpoints”.

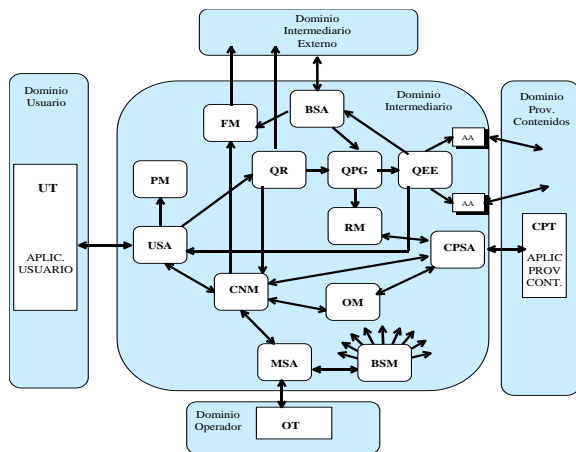
La siguiente figura muestra las capacidades de representación del conocimiento que se utilizó en el proyecto ABS:



5. Prototipo de intermediación del proyecto ABS

Las ideas presentadas en las secciones anteriores han sido llevadas a la práctica en el ámbito del proyecto ABS. Para ello, se diseñó una arquitectura de un intermediario genérico que fuera capaz de proporcionar los servicios y la representación del conocimiento especificados en las secciones anteriores.

Para ello, se utilizó una arquitectura basada en bloques funcionales, que se presenta en la siguiente figura:



Los componentes principales de esta arquitectura son los siguientes:

- Servicio de Acceso: Existen módulos de acceso basados en la arquitectura de servicios de TINA para cada uno de los posibles actores que quieran acceder al intermediario:
 - ◆ *USA: User Service Agent.* Módulo de Acceso de Usuarios.
 - ◆ *CPSA: Content Provider Service Agent.* Módulo de Acceso de los proveedores de contenido.
 - ◆ *BSA: Broker Service Agent.* Módulo de acceso de intermediarios externos para acometer actividades de federación
 - ◆ *MSA: Management Service Agent.* Módulo de acceso del operador del servicio.

Adicionalmente, el módulo PM (*Profile Manager*) es el encargado de gestionar los perfiles de cada uno de los actores que participa en el sistema, y que puede ser consultado por el resto de módulos que necesiten acceder a los datos de un perfil.

- Servicios de Búsqueda: Son los módulos encargados de recoger una petición de búsqueda del usuario y encontrar los proveedores de contenido más adecuados para esa petición, trasladándoles la petición total o parcialmente. Los módulos involucrados son:

- ◆ *QR: Query Redirector.* Decide si el intermediario tiene las áreas de conocimiento de dicha búsqueda o debe redirigirla a otro intermediario federado.
- ◆ *QPG: Query Plan Generator.* Consulta los recursos registrados por los suministradores y diseña un plan de búsqueda de acuerdo a los suministradores seleccionados para satisfacer dicha búsqueda. Este plan puede contener una descomposición de la petición en búsquedas parciales a distintos proveedores de contenidos.
- ◆ *QEE: Query Execution Engine.* Es el encargado de coordinar la redirección de la petición a los distintos proveedores de contenidos, y de componer las respuestas provenientes de cada uno de ellos
- ◆ *AA: Access Agent.* Agente de Acceso a cada Proveedor de Información. Estos agentes permiten respetar la forma de acceso de cada suministrador de información, de manera que el intermediario es el que se adapta al formato utilizado por cada suministrador de contenidos, tales como acceso vía Web, acceso SQL remoto a base de datos, etc.

- ◆ *RM: Resource Manager.* Módulo encargado de gestionar la base de datos de recursos, que han registrado previamente los suministradores de contenidos declarando sus áreas de conocimiento.

- Servicios de Navegación:

- ◆ *CNM: Conceptual Network Manager.* Módulo destinado a gestionar la base de datos donde están almacenadas las distintas áreas de conocimiento que maneja el gestor, y que ofrece a su vez la capacidad de navegar a través de dichas áreas a los usuarios y a los proveedores de contenidos.

- Servicios de Oferta:

- ◆ *OM: Offer Manager.* Módulo encargado de gestionar la base de datos de ofertas que puede ser utilizados por los proveedores de contenido para almacenar ofertas relacionados con un determinado concepto,

o por los usuarios finales para visualizar dichas ofertas.

- ❑ Servicios de Federación:
 - ◆ *FM: Federation Manager*. Módulo encargado de gestionar las importaciones y exportaciones del conocimiento del intermediario a otros intermediarios federados.
- ❑ Servicios de Gestión:
 - ◆ *BSM: Broker Service Manager*. Módulo encargado de coordinar la gestión del servicio y sistema, accediendo a lo distintos módulos del sistema para satisfacer las peticiones remotas procedentes del operador.

Esta arquitectura fue prototipada utilizando una plataforma de objetos distribuidos CORBA (*OrbixWeb*), desarrollando los módulos en el lenguaje JAVA, y fue validada mediante diversas pruebas de campo:

- ❑ Intermediario de Turismo: El intermediario era capaz de resolver peticiones relacionadas con paquetes turísticos mediante la conexión con suministradores de contenidos (las agencias de viaje francesas Degriffour y Reductour).
- ❑ Intermediario de Agricultura: El intermediario era capaz de ofertar un servicio de paginas amarillas (ofertas) a distintos agricultores con información relevante para ellos. Estas pruebas fueron realizadas en Finlandia por Telecom Finland (actualmente SONERA).

6. Conclusiones

El servicio de intermediación electrónica se está mostrando como una de las claves de la futura utilización del acceso a los contenidos y servicios en redes abiertas de comunicación electrónica, como lo es actualmente Internet. Para su desarrollo, es preciso una coordinación entre los proveedores de contenidos y servicios y los intermediarios para tener una representación común de dichos contenidos y servicios que permitan al intermediario ser capaz de procesarlos para ofertar servicios de navegación, acceso a ofertas y consultas a sus usuarios.

Otro aspecto importante es la normalización de las transferencias de información entre intermediarios que les permitan ser capaces de trabajar en unos entornos federado de intermediarios especializados, redirigiendo las peticiones de los usuarios a los intermediarios que se identifiquen como capaces de atender a dichas peticiones.

Se ha presentado a su vez los resultados del proyecto de investigación ABS (del programa ACTS de la Unión Europea), en el que se ha prototipado un intermediario, permitiendo validar dicho servicio en un entorno real.

Referencias

- [1] Juan I. Asensio, José I. Moreno, Víctor A. Villagrà "Modelado del Servicio de Intermediación Electrónica (brokerage) según el modelo de referencia de ODP: perspectiva de negocio". En "Comunicaciones de las Jornadas de Ingeniería Telemática JITEL 97". Bilbao, Septiembre 1997.
- [2] Juan I. Asensio, José I. Moreno, Víctor A. Villagrà, Jesús Redondo, Alexandre Nolle, Ionel Tothezan, George T. Karetos. "Application of TINA-C Computing and Service Architecture Concepts to the development of an Advanced Information Brokerage Service in the context of EC". Telecommunications Information Networking Architecture Conference, TINA'97. Santiago de Chile (Chile). Noviembre 1997.
- [3] Juan I. Asensio, Víctor A. Villagrà, José I. Moreno, Julio Berrocal. "An Approach to Electronic Brokerage in TINA Environments". Fifth International Conference On Intelligence in Services and Networks (IS&N 98). Amberes (Belgica). Mayo 98.
- [4] Proyecto ABS. Documento público D32. "Broker version 2 specification"
- [5] Proyecto ABS. Documento público D23: "Broker Business Model".

MECANISMOS DE COMUNICACIÓN Y GESTIÓN DE SERVICIO DE UN BROKER DE INFORMACIÓN MULTIAGENTE

Francisco Valera ¹, Jose Ignacio Moreno ¹, Víctor A. Villagrà ², Julio Berrocal ²

¹ *Area de Ingeniería Telemática, Universidad Carlos III de Madrid*

Avda. de la Universidad 30, 28911 Leganés (MADRID)

Email: fvalera@it.uc3m.es, jmoreno@it.uc3m.es

² *Departamento de Ingeniería Telemática, Universidad Politécnica de Madrid*

Ciudad Universitaria s/n, 28040 MADRID

Email: villagra@dit.upm.es, berrocal@dit.upm.es

Abstract

In this paper, a framework for Electronic Brokerage Service based on the emerging agent technology is presented. The first part is an introduction to brokerage and intermediation. In the second part the agent based brokerage service architecture is detailed, presenting the different modules in order to fully understand the whole system functionality. The third part covers in detail the communication scheme, giving information about inter-domain and intra-domain communication, as well as some tips about communication problems presented and the corresponding implemented solution. In the last part, a management system for this brokerage service is presented, again from architectural and functional point of view.

Main conclusions are taken from ACTS ABROSE (AC 316) project and ACTS ABS (AC 206).

1. Introducción

El Comercio Electrónico está principalmente basado en un intercambio de información eficiente y preciso entre las diferentes entidades implicadas en la transacción (típicamente cliente y proveedor), usando la infraestructura de comunicaciones existente. Gracias a las facilidades de comunicación proporcionadas por Internet y las redes intranet en general, la cantidad de información disponible es muy grande y además, se va incrementando continuamente (de manera análoga, también es habitual el aumento de la cantidad de información demandada).

Así, es necesario algún tipo de funcionalidad o mecanismo que permita a los clientes y a los servidores intercambiar información de manera fluida y precisa, tratando de minimizar el tiempo que normalmente se emplea en encontrar información verdaderamente útil que suele ser bastante alto.

1.1. Servicio de Brokerage

El servicio de brokerage (implementado por un broker de información), en el contexto del comercio electrónico, constituye una herramienta que ayuda tremendamente a potenciar la simetría de la tarea de intermediación, permitiendo que ambas partes se comuniquen fácilmente. Evidentemente el mayor problema con el que se enfrenta, es la búsqueda y posterior entrega de la información, específicamente requerida por los usuarios.

Dicho servicio de brokerage, de manera genérica, suele habilitar al usuario, la navegación a través de una clasificación o índice por contenidos, por el que puede ir buscando hasta que encuentra lo que necesita. Entonces se le provee de un conjunto de información con el formato atributo/valor, hasta que finalmente localiza los datos deseados.

Hay, sin embargo, aproximaciones más flexibles y transparentes basadas en el uso de recomendaciones personales (perfiles de usuario), que no obligan a tener que realizar la mencionada navegación, que en ocasiones, también puede llegar a ser un poco tediosa [1] y [2]. Concretamente vamos a distinguir dos tipos de sistemas:

- El primer modelo, utiliza un esquema de peticiones basado en una aproximación del lenguaje natural, de cara a que los usuarios puedan seleccionar la información que desean de una manera mucho más cómoda. Los usuarios pueden ir mejorando sus perfiles continuamente, basándose en las evaluaciones que ellos mismos hacen de la información recibida. Así, el mecanismo de búsqueda va refinándose continuamente.
- La segunda opción, consiste en utilizar filtros sintácticos colaborativos. La idea básica, es la de crear conjuntos de conocimientos comunes, de manera que los usuarios puedan obtener conocimientos unos de otros y aprovecharse mutuamente del proceso de refinamiento del mecanismo de aprendizaje anteriormente mencionado.

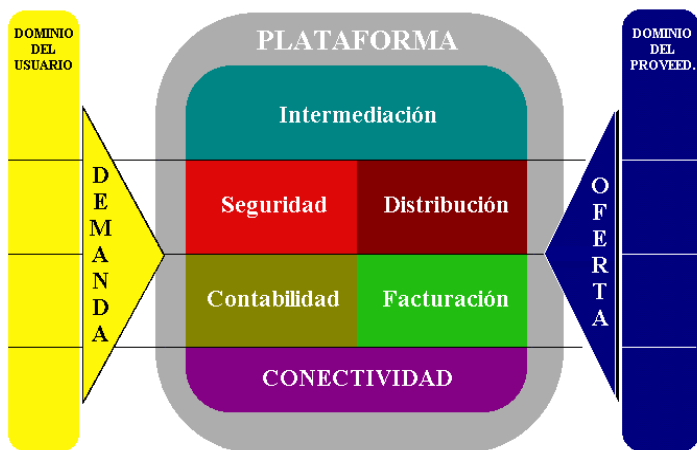


Figura 1. Plataforma de intermediación en el marco del comercio electrónico.

1.2. Intermediación

Un servicio de brokerage del estilo del que acabamos de introducir, estaría considerado como parte integrante de lo que se denomina, plataforma de intermediación [3].

Dicha plataforma, trata de englobar el conjunto de tareas que, se supone deben regir toda transacción que tenga la forma de la del comercio electrónico.

En la Fig. 1, podemos ver esbozadas las distintas tareas que constituyen la plataforma: *intermediación (brokerage)*, *seguridad*, *distribución*, *contabilidad*, *facturación* y *conectividad*. Todas ellas, como ya se ha dicho, están orientadas a disminuir las distancias entre la oferta y la demanda desde el punto de vista comercial.

1.3. ABROSE *

Centrados en el marco fijado por este entorno, lo que introduce este artículo, es precisamente, un Servicio de Intermediación Electrónica basado en tecnología de agentes, implementado utilizando Java (JDK1.1.7) y CORBA (OrbixWeb3.0) e incorporando un completo sistema de gestión. El servicio se desarrolla en el proyecto ABROSE (AC 316), dentro del programa ACTS la Comisión Europea.

ABROSE, formalmente especificado utilizando UML (con Rational™ Rose98™), hace un especial énfasis en la tarea de brokerage, incluyendo los dos enfoques anteriormente citados (navegación y colaboración).

* En el proyecto ACTS ABROSE (AC 316), participan las siguientes entidades: DT-Berkom, FT-CNET, Univ. of Toulouse, Univ. of Athens, Univ. of Berlin, Infomures from Romania, Onyx Ltd England, Dégriftour sa France, Univ. Politécnica de Madrid, Univ. Carlos III de Madrid

2. ABROSE

2.1 Objetivos

Los principales objetivos que se plantean en ABROSE (*Agent Based Brokerage Services in Electronic Commerce*) son:

- Captura dinámica de conocimientos.
- Incorporación de un sistema multiagente para la representación de la base de conocimientos.
- Interfaz gráfica para la navegación y petición de resultados (apoyada también en la tecnología de agentes).
- Interfaz gráfica y apoyo de agentes en el registro y propagación de ofertas por parte del proveedor.
- Utilización de Java y CORBA como lenguajes de implementación del sistema.
- Utilización de agentes colaborativos para optimizar la tarea de intermediación.
- Evaluar la incidencia de la tecnología de agentes en el dominio de los brokers de información y de aplicaciones comerciales en general.
- Aplicación de Java, CORBA y SNMP para la implementación de la plataforma de gestión del servicio utilizando también Tcl-Tk.

2.2 Funcionalidad

Desde el punto de vista funcional, el sistema ABROSE, es un servicio de brokerage accesible por Internet, que trata de cubrir las necesidades planteadas en los apartados iniciales, abordando los diferentes objetivos tecnológicos expuestos anteriormente.

El usuario accederá al sistema a través de un navegador de Web, pudiendo darse de alta o iniciar una sesión (con la consiguiente fase de acceso). Una vez dentro del servicio, dispondrá de una interfaz gráfica, por medio de la cual podrá realizar peticiones al broker o navegar por la base de conocimientos del mismo (ver Fig. 2).

Frente a una posible opción de implementación en el broker sería el encargado de consultar a los distintos proveedores y devolver las respuestas, que serían directamente presentadas en el navegador [4], en ABROSE lo que se hace es una búsqueda exhaustiva de proveedores afines a la petición y es el mismo usuario el que debe realizar el acceso.

La tecnología de agentes permitirá que el broker vaya aprendiendo de manera transparente las preferencias del usuario, aumentando la precisión a la hora de configurar la lista de proveedores afines, que hemos comentado.

La Fig. 2 ofrece el aspecto de una de las interfaces gráficas del sistema ABROSE (concretamente, se trata de la interfaz que permite al cliente realizar peticiones al broker).

El menú 'Mode' permitiría seleccionar, en función de que el sistema lo esté utilizando un usuario o un proveedor, si lo que se quiere hacer es una oferta o una petición. La descripción de dicha petición se escribiría

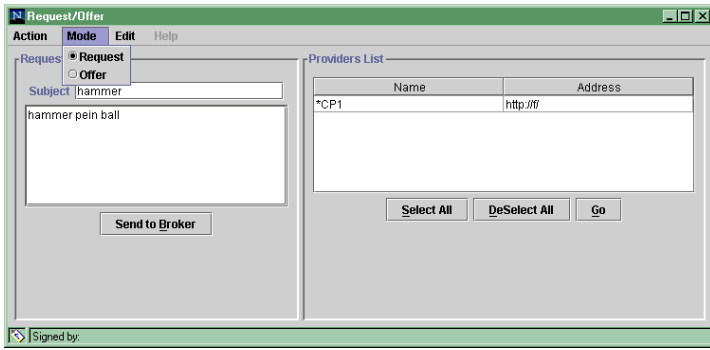


Figura 2. Interfaz gráfica de peticiones.

en lenguaje natural en los campos de texto de la izquierda de la interfaz. A la derecha aparecerían los proveedores que ofrece el broker como respuesta (o los usuarios interesados en la oferta si el cliente del sistema es un proveedor).

2.3 Arquitectura

La arquitectura del sistema está basada en dos dominios bien diferenciados: el *dominio del broker*, que tiene la doble misión de *brokerage* y gestión del sistema y el *dominio del usuario*, que asume toda la funcionalidad de visualización de información hacia el usuario final (navegación, peticiones, resultados, etc.).

La Fig. 3 muestra un esquema de la arquitectura con los diferentes bloques y el flujo de información existente entre ellos. Por cuestiones de simplicidad, se han abstraído detalles referentes a comunicaciones o gestión, que posteriormente se comentarán en detalle.

En lo que resta de apartado, se hará mención a los distintos módulos que constituyen el sistema, haciendo una división por dominios. Dicha descripción posibilitará una comprensión mucho más precisa del funcionamiento del sistema.

Dominio del Usuario

- *Asistente de Conexión (CA)*: es el bloque que ayuda al usuario a registrarse y conectarse al sistema.
- *Asistente de Gestión de Agentes (AMA)*: coordina las comunicaciones entre los diferentes módulos del dominio del usuario, así como la comunicación con el dominio del broker.
- *Asistente de navegación (NA)*: permite que el usuario (proveedor de contenidos o cliente) navegue por el espacio de conocimientos del broker y seleccione los dominios y criterios relevantes para sus peticiones.
- *Espía (Spy)*: se encarga de obtener información de las diferentes peticiones que haga el usuario, con el objetivo de mejorar el conocimiento del usuario que tiene el broker (perfil del usuario), lo cual irá facilitando progresivamente el proceso de adquisición de información precisa.
- *Front End (FE)*: es el punto de acceso a la información que posee el proveedor. El usuario

obtendrá una referencia para conectarse al FE como respuesta a sus peticiones.

Dominio del Broker

- *Gestor de Acceso del Usuario (UAM)*: la función principal de este bloque es la de mantener los perfiles de los usuarios y verificar el acceso a ABROSE. El perfil del usuario, tendrá por lo tanto, la información necesaria para permitir el acceso del usuario (login y contraseña), como mínimo. Se creará al iniciarse el sistema y deberá ser mantenido mientras se mantenga el servicio.
- *Gestor del Broker (BM)*: monitoriza la interacción entre los diferentes módulos del sistema. Se encarga además del arranque del sistema (su labor se explicará más profundamente en el apartado de gestión).
- *Terminal del Operador (OT)*: ayuda al operador del sistema a realizar la gestión y administración del servicio que provee ABROSE. Muestra el estado de cada bloque, facilitando el acceso a los parámetros de gestión que tenga cada uno.
- *Sistema Multi-Agente (MAS)*: en el dominio del broker, los agentes representan tanto a los clientes como a los proveedores. Cada agente va asociado a un módulo TA (agente de transacción), estando todos los TAs agrupados por conocimientos y controlados por el módulo MA (agente de mediación). El conocimiento del sistema estará basado en una red de conocimientos (BN), formada por el conocimiento de todos estos agentes internos. Cada vez que se hace una petición, basada en conocimiento específico del MA y/o del TA, se localizará a los correspondientes agentes del proveedor. El caso simétrico también es posible, cuando un proveedor hace una oferta y se localizan los correspondientes agentes de usuario. La arquitectura de los bloques que constituyen el MAS, se analiza con un poco más de detalle en el siguiente subapartado.

MAS (Multi-Agent System)

Se trata de una arquitectura en tres niveles, estando formado cada nivel por agentes que se comunican entre sí, marcados por el mismo patrón. Así, un agente de un determinado nivel, estará formado por agentes cooperativos en un nivel inferior (ver Fig. 4):

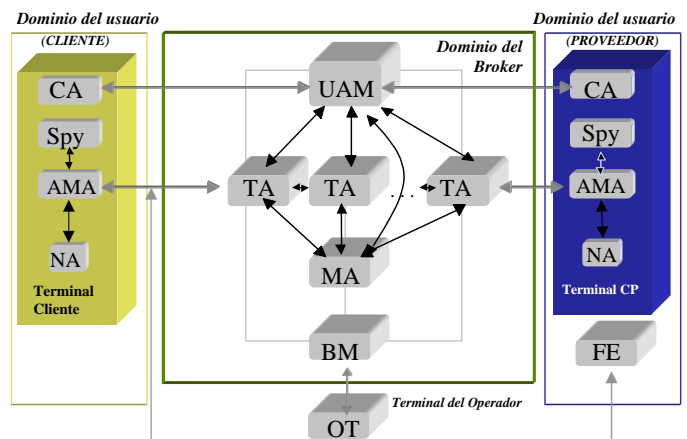


Figura 3. Esquema arquitectural del sistema ABROSE

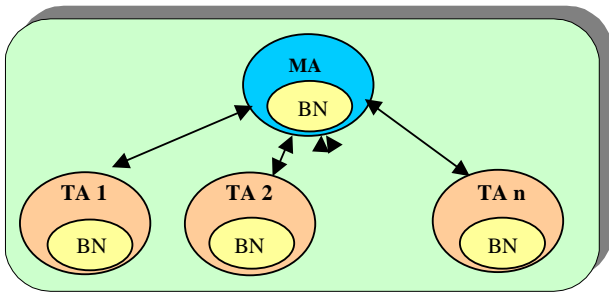


Figura 4. Esquema arquitectural del MAS.

- *Agente de mediación (MA)*, nivel superior: controlan el ciclo de vida de cada TA y encapsulan todo el conocimiento por ellos adquirido. El MA deberá controlar la creación de TAs asociados a nuevos usuarios, así como eliminar los TAs pertenecientes a clientes dados de baja. El MA puede añadirse o eliminarse del sistema, en función de la necesidad o no de un conocimiento global (BN global). Además, el MA va aprendiendo progresivamente de las transacciones que se hacen en el sistema.
- *Agente de transacción (TA)*, nivel medio: representan al usuario en el dominio del broker, almacenando conocimiento sobre él y tratando de conseguir información de otros TAs. Por eso, cada TA contendrá y mantendrá dos redes de conocimiento, una que contiene conocimientos sobre el propio usuario y otra con conocimiento del resto de TAs. El TA es el punto de entrada al dominio del broker, coordinando las comunicaciones entre dicho dominio y el usuario al que están asociados.
- *Agentes de conocimiento*, nivel inferior: tanto MAs como TAs, contienen información sobre sí mismos y sobre otros agentes situados en su mismo nivel. Esta información es la que organiza la sociedad de agentes. Los BAs son términos unidos a otros términos mediante enlaces con un determinado peso de unión. El BA de un TA o un MA constituye la BN (*Belief Network*) del agente, su base de conocimientos. Dicha BN está actualizándose continuamente para adaptarse a los conocimientos adquiridos durante las diferentes peticiones (además la BN del MA será un agregado de las diferentes BN de los TAs).

3. Comunicaciones en ABROSE

Este apartado explica la arquitectura e implementación de las comunicaciones en ABROSE, presentando dos nuevos módulos UCOMMS y BCOMMS que constituyen la base de dicha arquitectura, pues son los responsables de poner en contacto al cliente y al servidor (ver Fig. 5).

A lo largo del apartado se irá presentando progresivamente la arquitectura de comunicaciones en el dominio del usuario, en el del broker y entre ambos dominios.

3.1 Dominio del usuario

En términos generales, consistirá en un browser (Netscape) cuya máquina virtual ejecutará los diferentes módulos (CA, AMA, ...) que serán obtenidos del servidor de web mediante HTTP. Lo primero en ejecutarse, es un applet (lanzado desde la página web de acceso a ABROSE). Posteriormente se cargan el resto de los módulos en ventanas independientes, si es que incorporan una interfaz gráfica (como el CA o el AMA) o en *threads* independientes si no es así (UCOMMS).

Para comunicarse entre sí, todos estos módulos usan sus respectivas referencias de objetos Java, por lo que lo único necesario para habilitar las comunicaciones es un método que permita intercambiar dichas referencias. Una vez que se conoce la referencia del objeto destino, la comunicación entre dos módulos es tan sencilla como ejecutar el método deseado en el objeto en cuestión.

El módulo UCOMMS, responsable de comunicar el dominio del usuario con el dominio del broker, según se explicará en el apartado 3.3, se comunica con el resto de los bloques del dominio, haciendo uso también de las referencias de objetos Java.

3.2 Dominio del broker

Las comunicaciones se hacen en los mismos términos que en el dominio del usuario. Cada módulo se implementa con un thread independiente (BM, UAM, BCOMMS, MA ...), aunque en este caso no se usa ninguna interfaz gráfica.

Inicialmente se lanza el BM, que es el módulo responsable de inicializar el resto de los módulos y de repartir las referencias de unos y otros. Como se ha mencionado, las comunicaciones se harán utilizando dichas referencias para ejecutar los correspondientes métodos.

El módulo BCOMMS, hace el papel simétrico del UCOMMS en el dominio del broker. Es decir, será el encargado de unir el dominio del broker con el dominio del usuario. Las comunicaciones con el resto de los módulos del broker, se hacen igualmente utilizando la referencia Java.

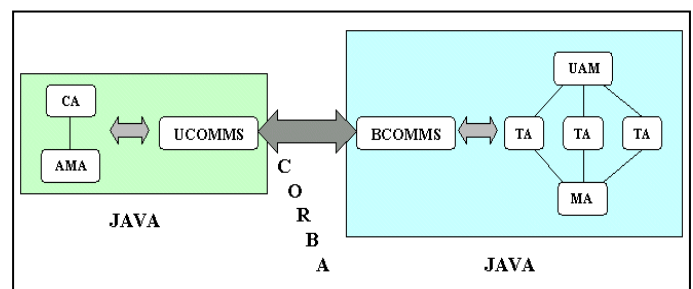


Figura 5. Esquema de comunicaciones en ABROSE.

3.3 Comunicaciones cliente-servidor (interdominio)

Las comunicaciones entre ambos dominios se han implementado utilizando CORBA.

Como se puede ver en la Fig. 5, se ha optado por un enfoque centralizado, con dos módulos (uno por dominio), responsables de enviar y recibir datos del otro dominio. De esta forma UCOMMS y BCOMMS son los dos únicos módulos que hablan CORBA entre sí.

Cuando el usuario necesite enviar alguna petición al broker, se invocará un método en el UCOMMS y este módulo se encargará de comunicar con el BCOMMS, que a su vez entregará la petición al módulo adecuado dentro del dominio del broker.

3.4 Comparativa

Otra posibilidad, consistiría en que cada módulo fuese responsable de sus comunicaciones con el resto de los módulos [5]. Cada módulo del dominio del broker constituiría un objeto CORBA y cada módulo del cliente accedería por su cuenta al módulo del broker correspondiente. Además los módulos del broker se comunicarían entre sí utilizando llamadas CORBA, en vez de las llamadas a través de referencias Java que es como se hace en nuestro caso.

Con el esquema utilizado en ABROSE, los únicos módulos que tienen que utilizar CORBA, como ya se ha dicho, son UCOMMS y BCOMMS, por lo que las comunicaciones se pueden gestionar de una manera mucho más sencilla. El resto de los módulos se desvincula por completo de la complejidad que implica el entorno de comunicaciones (conexiones, recepción y envío de datos, control de errores, etc.) y sólo precisarán una invocación Java en el UCOMMS o BCOMMS para acceder al otro dominio. Otra ventaja de este modelo, es que todos los módulos del broker se ejecutan sobre la misma máquina virtual, con lo que la invocación de un método sobre una referencia Java es prácticamente inmediata.

El otro esquema comentado, sin embargo, también tiene sus ventajas. El hecho de que prácticamente todos los módulos (salvo los del dominio del usuario entre sí) se comuniquen utilizando CORBA, exige definir previamente las correspondientes interfaces en IDL, quedando especificado ya desde el principio qué métodos y con qué atributos se pueden ejecutar desde el resto de los módulos. Esto puede suponer un mayor trabajo en la etapa de diseño, pero facilita tremendamente la integración del sistema, puesto que se tiene la certeza de que al menos la interconexión de módulos se hace de una manera homogénea.

Además, este modelo de comunicación de módulos, permite aprovecharse de las ventajas que ofrece CORBA en cuanto a distribución de objetos se refiere, posibilitando que cada uno de los módulos pueda estar ejecutándose en una máquina diferente.

Sin embargo, la utilización de CORBA se traduce en una disminución de la velocidad de ejecución con respecto al esquema de ABROSE. Además el esquema de ABROSE, como ya se ha dicho, sólo exige la ejecución de una máquina virtual en el broker (lo cual constituye uno de los motivos de la citada mayor velocidad de ejecución), mientras que de otra forma el consumo de recursos sería mayor.

Por último, hay que destacar que ABROSE no exige un alto grado de distribución, por lo que las ventajas del modelo centralizado tienen, en este caso, un mayor peso sobre las del esquema distribuido.

3.5 Multiusuario

Cuando un usuario se conecta al sistema, una de las primeras cosas que se hacen, es crear un nuevo módulo UCOMMS. Dicho módulo contactará con el BCOMMS del broker pasándole su referencia CORBA (y obteniendo a la vez la propia referencia del broker) y a partir de ese momento ambos dominios quedan en comunicación. De este intercambio, puede deducirse que únicamente habrá una instancia del BCOMMS ejecutándose a la vez en el broker, mientras que puede haber varias instancias de UCOMMS (una por cada usuario conectado). Esto incrementa considerablemente la complejidad del BCOMMS, haciendo que no sea únicamente un módulo puente, que se dedique a recibir llamadas del UCOMMS y a reenviarlas convenientemente a un módulo que las trate o que reciba invocaciones del broker para enviar respuestas al cliente. Desde el momento en que podemos tener varios usuarios conectados a él simultáneamente, se plantea el problema de mantener al BCOMMS permanentemente desbloqueado y listo para recibir peticiones de clientes o respuestas del broker.

La solución a este problema, pasa por crear un thread Java independiente, cada vez que el BCOMMS recibe una llamada del cliente. De esta manera, el proceso sería el siguiente:

- Cuando el UCOMMS recibe una petición del usuario, se la transmite al BCOMMS.
- En vez de hacer la petición al módulo del broker correspondiente él mismo (las peticiones CORBA pueden hacerse no bloqueantes, pero las peticiones Java, que son las que nos incumben ahora, bloquean al módulo llamante hasta que no termine la ejecución del método invocado), crea un thread que se encarga de hacer la verdadera petición.
- Así, es este thread el que queda bloqueado, liberando al BCOMMS que vuelve a estar libre para seguir recibiendo peticiones.

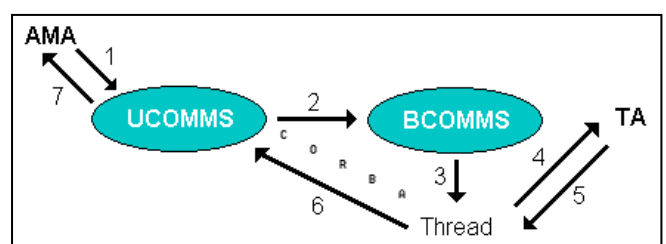


Figura 6. Detalle de comunicación entre bloques

Para devolver la respuesta al usuario, ya ni siquiera hace falta pasar por el BCOMMS. Es precisamente el mencionado thread creado por dicho módulo, el que recibe la respuesta (que normalmente será el atributo que devuelve el método Java invocado) y se la envía directamente al UCOMMS (con lo que el BCOMMS no precisa ser sobrecargado con una petición más). Podemos ver este proceso ilustrado en la Fig. 6.

3.6 Callbacks

Los callbacks, presentan la posibilidad de transmisión asíncrona de información desde el servidor hacia el cliente. Generalmente, la respuesta a una petición, suele venir como el atributo que devuelve un determinado método invocado (en cuyo caso, podríamos considerar que el cliente recibe la respuesta tan pronto como se acaba de ejecutar el método en el servidor). En ocasiones, sin embargo, es interesante que dicha respuesta venga de una manera asíncrona. Si la petición se prevé que puede tardar mucho tiempo en ser procesada, conviene no tener bloqueado al cliente en espera de que acabe la ejecución de un método y resulta más eficiente, desde el punto de vista del usuario, que una vez hecha la petición, quede desbloqueado y sea el servidor el que envíe la respuesta por sí solo cuando esté lista.

El problema en nuestro caso, viene generado, como ya se ha mencionado, por la introducción de Java como medio de comunicación entre bloques. Todas las llamadas Java son por definición bloqueantes, es decir no liberan al llamante hasta que el módulo que incorpora el método invocado, no termine la ejecución del mismo. En CORBA sin embargo, existen los llamados métodos *oneway*, que obviamente no pueden devolver nada (de otra forma obligarían a bloquear al llamante en espera de que acabe de procesarse el valor del atributo a devolver), no bloquean al llamante y son precisamente los que se utilizan para implementar callbacks.

En el caso de haber usado CORBA para comunicar los módulos del broker (siguiendo el esquema de comunicaciones distribuido que se ha contrapuesto al centralizado de ABROSE), no hacía falta ningún mecanismo del estilo de los threads auxiliares que se ha comentado para el BCOMMS, de cara a no dejarlo bloqueado.

Hecha esta introducción a los callbacks, podemos comentar un detalle que hay que solventar en el cliente (concretamente en el UCOMMS) y así podremos acabar detallando las comunicaciones UCOMMS-BCOMMS.

Cuando el usuario quiere hacer una petición, lo que en definitiva tiene lugar, es una invocación Java (bloqueante) típicamente del AMA o del CA hacia el UCOMMS. Para no dejar bloqueada la interfaz gráfica, se usa en el UCOMMS un artificio de threads semejante al que se utiliza en el BCOMMS. Sin embargo, hay ocasiones en las que sí es necesario bloquear la interfaz en espera del resultado de la petición. Por ejemplo en el

caso del login, conviene bloquear el terminal hasta saber si el acceso es o no válido. Hay también otras circunstancias como la modificación, creación o eliminación de un usuario, en las que conviene esperar para confirmar que la operación se realizó con éxito (típico método que devuelve un boolean). En definitiva, ahora tenemos el caso contrario: queremos que el cliente se quede bloqueado.

En principio para hacer esto, bastaría con que el UCOMMS no desviase la petición a un thread, sino que se encargase él mismo de procesarla haciendo una llamada CORBA bloqueante al BCOMMS. Cuando se acabe de procesar esa petición se devuelve un valor en la petición CORBA bloqueante y luego se devuelve un valor en la petición Java bloqueante. Sin embargo, desde el momento en que el BCOMMS no puede quedar bloqueado, no pueden usarse invocaciones CORBA bloqueantes, porque la idea es precisamente que tan pronto se hace la invocación CORBA, se genera un thread y se acaba el método invocado, con lo que se acabaría el bloqueo del UCOMMS y además el valor retornado no sería válido. Esto nos va a llevar a que todas las invocaciones CORBA deban ser *oneway*, es decir, no bloqueantes, lo cual exige otro tipo de solución al problema del bloqueo del cliente.

La solución adoptada es mantener al UCOMMS en espera tras hacer la petición CORBA, bloqueando así la llamada del AMA o del CA, hasta que llegue el callback del broker. En ese momento, se permite la terminación del método Java invocado y el terminal queda desbloqueado.

4. Gestión

4.1 Introducción

El esquema de bloques del sistema de gestión que incorpora ABROSE, tiene el aspecto de la Fig. 7:

Básicamente podemos dividir en dos subsistemas diferentes todo el esquema de bloques:

- El sistema ABROSE, descrito anteriormente.
- El *Terminal del Operador* (OT) y otras aplicaciones de gestión, que permitirán al operador interactuar con el sistema realizando múltiples acciones (iniciar el sistema, pararlo, grabar su estado,

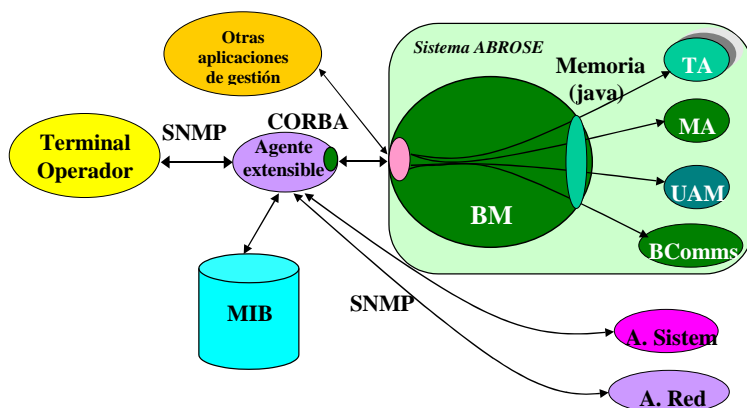


Figura 7. Esquema de bloques del sistema de gestión

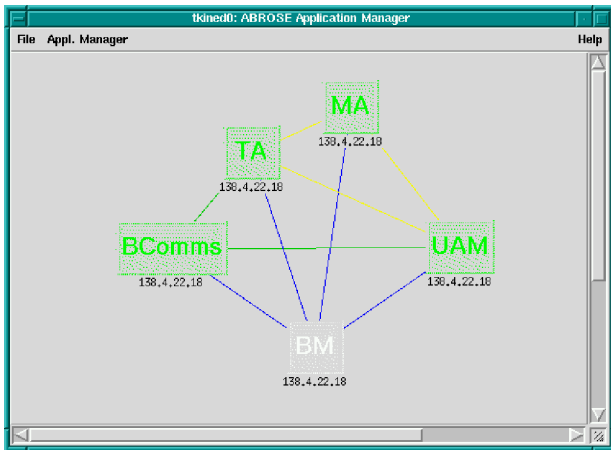


Figura 8. Interfaz gráfica de gestión, con el status de cada componente

realizar una monitorización gráfica de los diferentes módulos, acceder a las diferentes variables de gestión, acceder a ficheros de registro y obtener notificaciones de los distintos módulos. En definitiva, gestionar el sistema completo.

Los agentes del sistema y de red, permiten gestionar el entorno hardware en el que se ejecuta el sistema, mientras que el agente extensible permite al operador monitorizar el sistema entero utilizando únicamente una interfaz ocultando la presencia de numerosos agentes.

4.2 BM

Como puede apreciarse en la figura el BM es el módulo responsable de la gestión del sistema. Está encargado de iniciar, parar y almacenar el sistema, monitorizar el status de cada componente (ver Fig. 8) y actuar de puente entre el terminal del operador y el sistema, recibiendo peticiones sobre las variables a gestionar y enviando las notificaciones relevantes de cada módulo.

Para hacer todo esto, el BM utiliza diferentes threads que interaccionan entre sí:

- *Principal*: mantiene al resto de las threads y además se encarga de iniciar, almacenar y parar el sistema.
- *Gestor de notificaciones*: recibe las notificaciones de cada módulo. Si son referentes a *start*, *stop* o *save*, las convierte en *traps* SNMP y se las reenvía al OT.
- *Gestor de información*: se encarga de recibir la correspondiente petición de gestión (*get* o *set*) vía CORBA, de localizar el componente hacia el que va dirigida dicha petición (mirando en una tabla de configuración que mantiene el BM), de trasladar la petición al módulo adecuado utilizando referencias Java y de devolver el valor obtenido.
- *Ping*: se encarga de acceder periódicamente a una variable de gestión de estado ubicada en cada componente, enviando notificaciones periódicas al OT.
- *Manejador de gestión*: cada componente tiene un manejador de gestión, que incluye la información a gestionar del componente y además tiene la interfaz

mediante la cual el gestor de información accede a las variables a gestionar. El manejador del BM, incorpora además la tabla de configuración donde se almacenan las referencias al resto de los módulos (y que utiliza el gestor de información para reenviar peticiones).

4.3 Información gestionada

La información gestionada almacenada en un sistema, determina qué es exactamente lo que se va a gestionar. La existencia o no de una variable de gestión, implicará la posibilidad o no de gestionar una determinada característica del sistema.

La idea inicial en ABROSE, es poder gestionar los cinco puntos definidos por el OSI MN (*Open System Interconnection Managing System*): fallos, configuración, anotaciones, prestaciones y seguridad (FCAPS). Para ello, se ha definido la siguiente información en la MIB (*Management Information Base*) de ABROSE:

- Cada componente tiene una variable de status, que aporta información sobre el estado del módulo.
- Se han definido traps para iniciar, parar y almacenar (*start*, *stop*, *save*) el sistema.
- Tanto el BM como el BCOMMS, debido a su importante labor de interfaz al sistema de gestión y al broker respectivamente, tienen algunas variables adicionales:
 - El BCOMMS contabiliza el número de veces que los diferentes UCOMMS acceden al broker, almacenando también qué método se utilizó en cada acceso. También incorpora información del tiempo que se tardó en ejecutar cada método.
 - El BM tiene variables para configurar el tiempo entre muestreos del status de cada componente (*polling*), ubicación del fichero de registro de notificaciones, variables para llevar a cabo el *start*, *stop*, *save* y dos tablas: la ya mencionada de configuración, con todos los componentes y su estado y otra tabla con las notificaciones almacenadas durante el último arranque del sistema.

4.4 Funcionalidad de gestión

Se han implementado dos diferentes maneras de gestionar el sistema, de cara a poder utilizar cualquier plataforma de gestión, sin limitación de protocolos.

CallSystem

Es una aplicación que supone una funcionalidad mínima del sistema de gestión, pues utiliza el esquema presentado en la Fig. 9 en vez del de la Fig. 7. La idea es acceder directamente utilizando CORBA a la interfaz de gestión del BM, permitiendo únicamente iniciar, parar o almacenar el estado del sistema. Este esquema tan simple, resulta sin embargo de gran utilidad en entornos de prueba, aunque no puede utilizarse para gestionar el sistema en un entorno real.

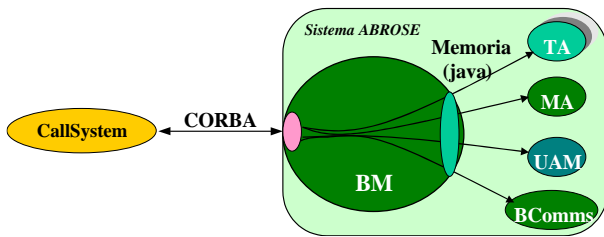


Figura 9. Aplicación de gestión CallSystem

OT

Es mucho más potente que la aplicación anterior, permitiendo iniciar el sistema, pararlo, almacenar su estado, monitorizar gráficamente los módulos, acceder a las diferentes variables de gestión, acceder a ficheros de registro y obtener notificaciones de los distintos subsistemas. Es la aplicación que permite gestionar por completo el sistema.

Para conseguir esto, el OT accede al agente extensible, que se estará ejecutando en un puerto UDP bien conocido. Este consultará la MIB y reenviará la petición hacia el agente correspondiente o hacia la interfaz CORBA del BM, gracias a un *traductor* de SNMP específicamente diseñado.

A diferencia del CallSystem, este esquema exige que tanto el agente extensible, como los diferentes agentes para manejar la red o el sistema, estén activos.

5. Conclusiones

Las principales conclusiones extraídas con respecto a comunicaciones, están relacionadas con la elección de un esquema centralizado de comunicaciones entre dominios y la utilización de Java dentro de los dominios.

El esquema centralizado, permite concentrar la complejidad inherente a todo sistema que exija cierta conectividad, únicamente en dos módulos, liberando al resto de la necesidad de tener en cuenta la arquitectura de comunicaciones. Como todo sistema centralizado, implica cierta pérdida de flexibilidad y un pequeño retardo adicional, al tener que pasar por un módulo intermedio antes de llevar a cabo la comunicación.

El hecho de utilizar Java como medio de comunicación de módulos dentro de cada dominio, posibilita una mayor rapidez que si se usase CORBA, pero obliga a suplir la versatilidad que ofrece CORBA a la hora de realizar callbacks, con un código un poco más elaborado. Por otro lado, el entorno de aplicación no pretende ser excesivamente distribuido ni heterogéneo, lo cual sí justificaría la utilización de CORBA en las comunicaciones internas de los dominios.

Desde el punto de vista de gestión, lo que se ha planteado es un modelo donde coexisten diferentes tecnologías: Java para gestionar internamente el sistema, CORBA como protocolo de comunicación del

sistema con el exterior y SNMP como protocolo de conexión entre el gestor y los diferentes agentes.

Para facilitar la labor de desarrollo, se utiliza una aplicación que, minimizando las tareas de gestión realizables, permite arrancar el sistema completo, así como pararlo y salvar su estado.

El modelo completo de gestión incorpora un plataforma sobre Tcl-Tk, que permite utilizar una vistosa interfaz gráfica para monitorizar el status de cada componente, visualizar las diferentes variables de gestión, iniciar o parar el sistema, etc.

Tanto la arquitectura de comunicaciones como la plataforma de gestión presentadas, ya han sido convenientemente verificadas y validadas en el prototipo implementado para la versión 1 de ABROSE.

Agradecimientos

Este trabajo ha sido parcialmente financiado por la Comisión Europea a través de los proyectos ACTS ABROSE (AC316) y ABS (AC206).

Referencias

- [1] ACM – “*Recommender Systems*”. Special Issue of Communications of the Association of Computing Machinery – Vol40 N°3 – March 1997
- [2] “*An adaptive Web page recommendation service*”. Balabanovic Marko. Proceedings of autonomous agents, 1997
- [3] *ABS Broker Business Model*, D23. P. Alzon, P. Tothesan, M. Hubert, E. Athanassiou, A. Hoang Van. 1996 (<http://b5www.berkom.de/ABS/D23.htm>)
- [4] ABS International Deliverable, D42. 1998
- [5] ABS International Deliverable, D32. 1997
- [6] *An Approach to Electronic Brokerage in TINA Environments*. Juan I.Asensio, Victor A. Villagrà, José I.Moreno, Julio Berrocal. Fifth International Conference On Intelligence in Services and Networks. IS&N’98. May 25-28, 1998. Antwerp, Belgium. Published by Springer in LECTURED NOTES IN COMPUTER SCIENCE 1430.
- [7] *Application of TINA-C Computing and Service Architecture Concepts to the development of an Advanced Information Brokerage Service in the context of EC*. J.I.Asensio, J.I.Moreno, V.Villagrà, J.Redondo. A. Nolle, I. Tothesan, G.Karestos. Telecommunications Information Networking Architecture. TINA’97 Conference. Santiago de Chile, 18-21 November 1997
- [8] *ABROSE: Agent Based Brokerage Services in Electronic Commerce (ACTS: AC316)* (<http://b5www.berkom.de/abrose/>)
- [9] *ABS: Architecture for Information Brokerage Service (ACTS: AC206)* (<http://b5www.berkom.de/ABS/>)

Sistema de Gestión de Servicios de Telecomunicación en un Entorno Cambiante

KOEHN, Rogério – GUERRERO, Carmen
Departamento de Electrónica y Sistemas
Campus de Elviña s/n
Universidad de A Coruña
15071 - La Coruña
Tel: 981-167000, Fax: 981-167160
[rkoehn ,clopez]@udc.es

BORDON, Luis – FERNANDEZ, Gaudencio
Norsistemas
Av. Marina 16
08005 – Barcelona
Tel: 93-2256940, Fax: 93-2256941
[lbordona ,gfernand]@norsistemas.es

Abstract

This paper describes a system architecture for telecommunication service management using a network infrastructure based on radio links, a solution normally used by minor telecommunication providers. This system deals with service management in a changing environment. It is composed by a service provisioning module that manages all necessary resources for service delivery on a Network database, a trouble-ticketing module that associates customer requests with relevant services and network resources, and a service activation module that interacts with exchange management and access network systems. The most important feature of this architecture is the flexibility in adding new types of services and in changing the processes that manage these services.

1. INTRODUCCIÓN

El marco de liberalización del mercado de las telecomunicaciones en Europa eleva el grado de competencia entre operadores, obligándoles a ofrecer nuevos servicios con un tiempo de respuesta muy rápido. Una vez terminado el monopolio, los nuevos operadores construyen sus redes basándose en los servicios que ofrecerán. Ligado a esta estrategia, los sistemas de gestión de servicios de estos operadores, deben funcionalmente ofrecer una alta flexibilidad que permita la introducción de nuevos servicios con personalización regional o individual y por tanto de nuevas tipologías de red y deben ser capaz de interactuar con otros sistemas de nivel de gestión inferiores, como pueden ser los sistemas de configuración de las centrales de conmutación, o los sistemas de gestión de alarmas de la red de transmisión o red de acceso [1].

La gestión de nivel de servicio es el objetivo más importante de aquellas empresas de redes orientadas al mercado. En este nivel de gestión se incluyen actividades como la configuración y provisión de servicios, la gestión de incidencias y la supervisión de la calidad de servicio.

A pesar de los esfuerzos existentes por parte de los organismos de estandarización, casi todo los trabajos existentes actualmente están centrados en la arquitectura TMN, sobre todo en los niveles de gestión de elemento de red y de gestión de red [2]. En lo que se refiere a nivel de gestión de servicio, las recomendaciones aún no están suficientemente desarrolladas[3].

Como se describe en la arquitectura TMN, el nivel de gestión de servicios se puede ver como una relación cliente-servidor del nivel de gestión de red. El nivel de

red ofrece al nivel de servicio un circuito portador, o de cliente, que soportará el servicio extremo a extremo. Por su vez este circuito portador de nivel de red estará compuesto por recursos físicos y lógicos de nivel de gestión de elemento de red. La figura 1 ilustra esta relación.

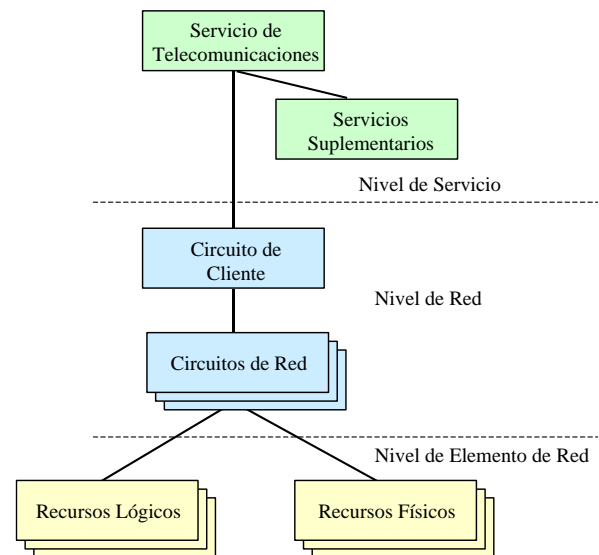


Figura 1 – Entidades involucradas en un servicio

La arquitectura del sistema de gestión ha de soportar el entorno multiservicio de las redes actuales. La flexibilidad de la red obliga a los operadores a introducir tecnologías de red diversas, como redes de acceso vía radio punto multipunto, bucle de abonado vía fibra óptica, redes de transmisión de tecnología WDM, o circuitos contratados a terceros operadores. Esta diversidad de tecnología condicionará el diseño de la arquitectura de un nivel de gestión de servicio.

Los procesos que se puede considerar en la gestión de servicios son básicamente la provisión de servicios y la gestión de incidencias asociadas a servicios. El control de calidad de los servicios puede considerarse como un subapartado de la gestión de incidencias, una vez definido los umbrales de calidad, donde se genera una incidencia de servicios si no se cumple con estos umbrales. En este artículo se presenta una arquitectura de un sistema de gestión de servicios de telecomunicación que ofrece soluciones a la problemática de la diversidad de servicios, tecnologías y procesos de gestión hacia el usuario [4].

La presentación de este trabajo se estructura del siguiente modo: el apartado 2 describe los diferentes procesos involucrados en la gestión de servicios y su correspondiente diagramas de flujo y estado. El apartado 3 presenta un modelo de red en el que se describen los recursos de la red de conmutación y acceso necesarios para la provisión de los servicios a gestionar. El apartado 4 presenta la arquitectura propuesta para implementar los procesos asociados a la gestión de los servicios. Finalmente se señalan las conclusiones obtenidas del trabajo presentado.

2. PROCESOS EN LA GESTIÓN DE SERVICIOS

Previo a la descripción de los procesos de provisión y de gestión de incidencia, es importante conocer las características básicas de los servicios de telecomunicación sobre los que centramos este trabajo.

2.1 Tipos de servicios

Los nuevos operadores de telecomunicaciones tratan de diversificar su oferta de servicios de telecomunicaciones ofreciendo desde los clásicos servicios de voz hasta servicios de red inteligente y de datos.

El sistema que se describe en esta publicación se centra en servicios de voz, no obstante puede ser extensible a todo tipo de servicio de telecomunicaciones.

Los principales servicios de voz que se ofertan son:

POTS (Plain Old Telephone System) – servicio de voz analógico de 4 kHz, y que se permite la transmisión de datos dentro de la banda audible de hasta 28800 bps. Este servicio presenta dos variantes: POTS clásico donde sólo se dispone de una única línea y POTS empresa, donde el servicio esta compuesto por un conjunto de líneas configuradas en un grupo de salto.

BRA (Basic Rate Access) - servicio RDSI de acceso básico 2B+D y que permite la transmisión de datos hasta 144 kbps. Este servicio presenta las siguientes variantes: BRA clásico con un unico acceso básico, BRA empresa con salto de llamada con más de un acceso básico pero agrupado en un único numero telefónico y BRA empresa con la posibilidad de

marcación directa a extensiones, donde a un grupo de acceso básico se les asigna un rango de numeración telefónica.

PRA (Primary Rate Access) – servicio RDSI de acceso primario 30B+D y que permite la transmisión de datos hasta 2048 kbps. Este servicio presenta las siguientes variantes: PRA con salto de llamada o PRA con marcación directa a extensiones.

2.2 Proceso de Provisión de Servicio

Desde el punto de vista de red, el proceso de provisión de un servicio a un cliente significa esencialmente el estudio, en primer lugar, de la factibilidad de la instalación de dicho servicio, y la posterior construcción y configuración del acceso. Desde el nivel de red se debe determinar si el cliente es accesible por la red, y en caso afirmativo proporcionar y activar los equipos, circuitos y demás elementos de red necesarios para la puesta en marcha del servicio.

Este proceso se implementa a partir de dos sistemas: un *sistema de información* que mantiene un inventario actualizado de los elementos de red tanto en lo que respecta a su identificación y características como a su estado de operación, y un *flujo de trabajo* que determina los estados a través de los cuales debe pasar un servicio desde el momento en que se solicita hasta su activación a través de los distintos departamentos de red.

En cada etapa del flujo de trabajo se accede a la información disponible acerca de los elementos de red, y en función de ella se procede a la asignación de los mismos al servicio en estudio y construcción según las características demandadas.

Se pueden diferenciar de este modo tres niveles en el sistema dispuestos en jerarquía de mayor a menor en:

1.- Nivel de servicio. En él se gestionan las características tanto técnicas como relativas al servicio necesarias para la provisión de red (tipo de servicio, parámetros del servicio, servicios suplementarios, dirección del cliente, etc). El servicio tiene asociado un estado que determina las diferentes fases por las que pasa durante el proceso de provisión.

2.- Nivel de red. Un servicio debe pasar por estudio, planificación y configuración a través de los distintos departamentos que gestionan la red. A cada una de estas áreas le corresponde dar la viabilidad y asignar los recursos propios de cada segmento. El control de la provisión dentro de las áreas de red se lleva a cabo mediante unas variables de estado que informan de la actividad desarrollada en ellas.

3.- Nivel de elemento de red.- Contempla los recursos físicos y lógicos que componen la red. Registrados en un sistema de información accesible por los agentes del flujo de trabajo, son asignados a los servicios a lo largo del mismo, variando de este modo su estado de operación, desde su planificación hasta su activación final.

El proceso de provisión de red puede subdividirse en dos subprocesos: el estudio de viabilidad, y la construcción y configuración del acceso. El primero comienza cuando el departamento comercial del operador solicita al departamento de red un estudio a fin de determinar la viabilidad de prestar servicio a un cliente determinado, y finaliza con la respuesta de red tras la consulta de los recursos disponibles y el acceso a la sede del cliente. Durante esta fase el nivel de red se limita a planificar los elementos que serían necesarios para el establecimiento del servicios, sin una implementación real. Si la respuesta es afirmativa el área de red queda a la espera de que el departamento comercial lance la orden definitiva de construcción del acceso. Cuando esto ocurre se procede a la instalación física de los equipos necesarios en la sede del cliente así como a la configuración de los elementos de la red de acceso hasta la central local correspondiente, tras lo cual se realizan las oportunas pruebas sobre el acceso. Finalmente se procede a la activación del servicio, concluyendo así el proceso de provisión.

A continuación se describe el flujo de trabajo seguido por un servicio para su provisión:

1) Estudio de viabilidad

El departamento comercial realiza una petición de estudio de un proyecto indicando los servicios que lo componen, sus características y los datos sobre la localización del cliente necesarios para determinar su acceso. El servicio entra de esta manera en estado "En estudio".

1.1) Acceso

Se determina si el acceso es vía radio o fibra óptica. En el primer caso se comprueba que la sede del cliente se encuentre iluminada por alguna de las estaciones base de la red de acceso, y se le asocia una de ellas. Además se planifica el equipo de recepción que debe situarse en la sede del cliente.

1.2) Transmisión

Se determina el circuito de transmisión que debe transportar el servicio, bien directamente desde la sede del cliente, o bien desde una estación base hasta la central, planificando la capacidad solicitada por el servicio al mismo tiempo que se controla la cantidad y

ocupación de los circuitos disponibles que afluyen al punto de conexión con la red de transmisión (central local o nodo de datos).

1.3) Conmutación

Se planifican los elementos de entrada a la central local: módulos, interfaces, time slots y puertos. También se procede, en el caso de servicios de voz, a la reserva de los rangos de numeración que se hayan solicitado para el servicio.

Si la reserva de los recursos mencionados se ha llevado a cabo sin problemas, el servicio pasa a ser "Viable". En este punto todos los elementos de red involucrados se encuentran "Planificados", es decir, reservados a la espera de que se de la orden de construcción del acceso.

2) Construcción y configuración

Cuando el departamento comercial, informado previamente de la viabilidad del servicio, da la orden de construcción del mismo, el área de red lleva a cabo la instalación y configuración efectiva del acceso. El servicio entra en estado "En configuración".

2.1) Acceso

Se realiza la instalación de los equipos de acceso situados en la sede del cliente, se configuran en el sistema de gestión de la red de acceso sus tarjetas y puertos asignados, para la provisión del servicio requerido

2.2) Conmutación

Se configuran en el sistema de gestión de las centrales de conmutación locales, los elementos necesarios de la central, para la provisión del servicio requerido.

A continuación se realizan pruebas para verificar el funcionamiento correcto del acceso. Una vez concluidas el servicio pasa a estar "Configurado" y los recursos de red asociados se ponen en estado "Operativo". Cuando finalmente se pone en funcionamiento el servicio, tanto éste como los elementos de red asociados pasan a estado "Activado".

En la figura 2 se ilustra los recursos que se activan desde la central local hasta la casa del cliente, considerando un servicio POTS o BRA de tecnología de acceso vía radio punto-multipunto.

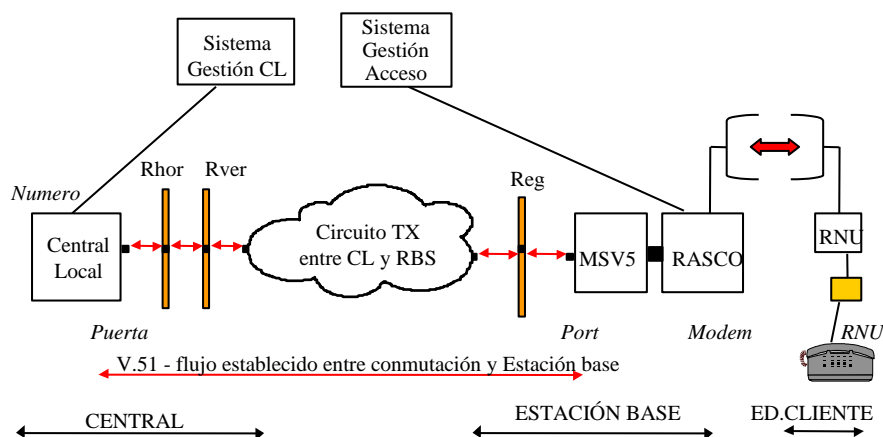


Figura 2 - Estructura de red para servicio de voz

2.3 Proceso de Gestión de Incidencia

De manera general se puede decir que el objetivo de un sistema de Gestión de Incidencias es resolver todas las reclamaciones de usuarios relativas a servicios de telecomunicaciones ofrecidos por el operador en cuestión. Estas reclamaciones serán notificadas por el cliente y estarán producidas por una pérdida en la calidad del servicio ofrecido o por una pérdida total del mismo.

Se pueden dar dos situaciones: que el cliente notifique una disfunción en el servicio, antes o sin que la operadora se percate de la situación –lo que comúnmente se denominará *Reclamación de Cliente*– y que la operadora, a través de sus sistemas de gestión de red, notifique una fallo en el funcionamiento de un equipo o una pérdida de calidad en un servicio ofertado –lo que se denominará *Alarma de Red*–. El sistema de Gestión de Incidencias, por tanto, deberá soportar estos dos tipos de incidencias.

En ambos tipos de incidencias se generará un *trouble ticket* –parte de incidencia– que pasará por diferentes departamentos de la operadora y a través de diferentes estados hasta su cierre final, es decir, hasta que el servicio degradado vuelva a funcionar correctamente. Normalmente los departamentos integrantes del flujo de información seguido por el parte de incidencia deberán ser los siguientes: un departamento de *Atención al Cliente*, que se encargue de recoger todas las incidencias notificadas por los usuarios y que al mismo tiempo interrogará al mismo sobre los síntomas de la disfunción del servicio, y un departamento de *Mantenimiento de Red*, que se encargará de, a partir de la información recogida por Atención al Cliente, resolver el problema, notificando posteriormente al anterior departamento la resolución del parte.

De manera análoga, el departamento de Mantenimiento de Red podrá notificar al de Atención al Cliente una *Alarma* en un equipo de la red que podrá afectar a servicios de usuarios finales, pudiéndose adelantar éste avisando al cliente de una pérdida del servicio antes de que el cliente mismo inicie una reclamación.

Los diferentes estados por los que pasará un parte de incidencia podrán ser:

- *Nuevo*: El parte ha sido generado por Atención al Cliente, o por Mantenimiento de Red en caso de Alarma.
- *Aceptado*: Ha sido aceptada la resolución del parte por parte de Mantenimiento de Red.
- *Rechazado*: Mantenimiento de Red considera irresoluble al parte por diferentes motivos.
- *Retrasado*: La resolución del parte ha sido aplazada por Mantenimiento de Red.
- *Solucionado*: Mantenimiento de Red ha solucionado el problema asociado al parte.
- *No Competencia*: Mantenimiento de Red, después de un periodo de investigación considera que no es de su competencia la resolución del parte.

El diagrama de estados por los que pasará un parte de incidencia correspondiente a una Reclamación de Cliente esta ilustrado en la figura 3.

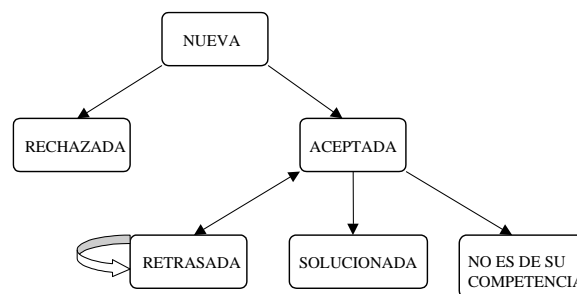


Figura 3 – Estados de las reclamaciones de servicio

El diagrama de estados por los que pasará un parte correspondiente a una Alarma de Red se muestra en la figura 4.

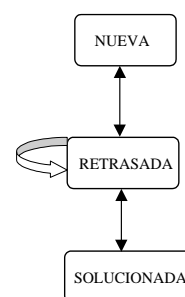


Figura 4 – Estados de las alarmas de servicio

3. MODELO DE RED

Los recursos de conmutación implicados en la provisión de servicios de voz POTS y BRA de acceso directo que deben estar soportado en el modelo de datos (ver figura 2) son los siguientes:

Las centrales locales (CL) deben considerarse, de manera lógica, como un recurso de red, ya que los recursos de nivel jerárquico inferior (elemento de red) son los que limitan la capacidad de la red. Debemos tener en cuenta que existen servicios con limitaciones por central local (p.e. el servicio de marcación directa a extensión presenta una limitación de 500 clientes por central local).

Módulos de interfaces V5.1 en las centrales locales. Estos módulos se modelan como conjuntos de interfaces V5.1 en la central local. Una central local presenta una serie de módulos que a su vez presentan una serie de interfaces. Se debe tener en cuenta que hay servicios que presentan limitaciones por módulo (p.e. los servicios de desvío de llamadas presentan un límite de 1536 números por módulo).

Interfaces V5.1 entre la central local y *el nodo de acceso (MSV5)*. Este es un recurso compartible por más de un servicio (por una interfaz de este tipo pueden soportarse hasta 30 servicios POTS). Este recurso se identifica por un identificador de interfaz que esta compuesto por 3 bytes. El primero indica el módulo de la central local en el que se conecta el interfaz y los dos siguientes son el número de la primera puerta remota asociada a ese interfaz.

- **Puertas remotas.** Son recursos no compartibles debido a que se asigna una puerta de este tipo a un servicio POTS o se asignarán dos de estas puertas a un servicio BRA. Una interfaz V5.1 tiene 32 puertas remotas. En cada módulo hay un rango disponible de 4096 puertas remotas y la numeración empieza por 5120. Una puerta remota tiene tres atributos (Time Slot, Dirección L3, Dirección EFA).

• **Servicios suplementarios** que presentan unas limitaciones en las centrales locales. Estos servicios deben ser tratados como recursos que se debe tener un control de los servicios ya reservados o asignados para poder saber en todo momento si pueden ser provisionados o no.

Números telefónicos. Este será un recurso lógico. Para un servicio POTS, sólo reservaremos un número y para un servicio BRA reservaremos dos. Ejemplo: En el caso de que el cliente demande una centralita PABX con salto de llamada y con 20 líneas, debemos reservar 20 números.

Los recursos de conmutación utilizados por los servicios PRA son los siguientes:

Central Local (CL). Contiene los diferentes módulos PRA y los servicios suplementarios que soporta.

Módulos PRA. Son módulos (diferentes de los v5.1) que contienen los puertos que dan servicio a los PRA.

Puertos. Cada puerto permite una conexión de un servicio PRA. Un servicio PRA está formado por un grupo de PRAs donde cada uno de ellos ocupa un puerto.

Rango de Numeración: Para este tipo de servicios se debe asignar una numeración al grupo de PRA.

Una vez hemos decidido los recursos de conmutación involucrados en el proceso de servicios de acceso directo, debemos definir las relaciones que existen entre ellos. En el esquema que se presenta en la figura 5 se presenta la relación entre recursos.

Los recursos utilizados en la red de acceso de tecnología punto multipunto para los servicios POTS, BRA y PRA son los siguientes:

Estación Base (RBS): es el emplazamiento donde están los equipos MSV5 y RASCO que permiten de conectar los clientes de los edificios a la central local. En una central puede haber más de un equipo MSV5 y más de un RASCO. En una estación base habrá tantos equipos RSCO como sectores (antenas) tenga la estación RBS.

MSV5: equipo que está en la RBS y que se conecta por un lado con el nodo de conmutación de voz y de datos y por el otro con la parte a radio-frecuencia (RASCO) de la RBS; el MSV5 desenvuelve las tareas de conmutación de los flujos en entrada hasta el nivel 64k. De la salida de un MSV5 pueden colgar más de un RASCO.

RASCO: equipo que está en la RBS y que se conecta por un lado con el MSV5 y por el otro con la antena, y que contiene los módems. Las señales que salgan de cada módem vienen agrupados en una única señal a radio-frecuencia que llega hasta la antena; por cada

sector hay una antena. Un RASCO se asocia a una antena o a un sector.

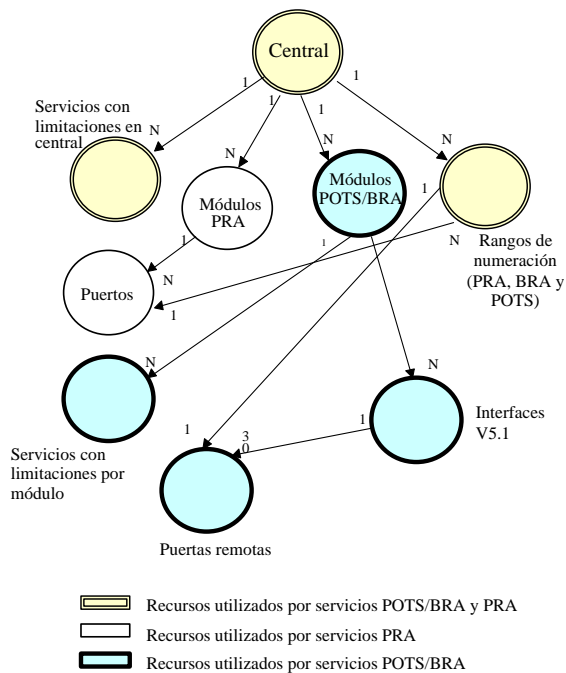


Figura 5 – Relación entre recursos de conmutación

Módems: equipos que están en el RASCO; a cada sector están asociados algunos módems; cada módem tiene asociada una frecuencia portadora y está conectado con una RNU vía radio. Por cada módem se configuran los siguientes parámetros: número máximo de canales 64k que pueden ser modulados, tipo de modulación y BER objetivo de la conexión radio;

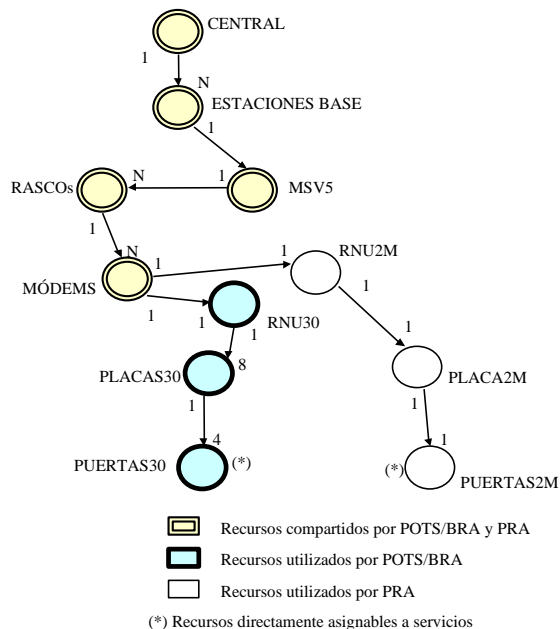


Figura 5 – Relación entre recursos de acceso

RNU: equipo que está en el edificio de los clientes; por un lado está conectado con la antena a través de un modem parecido al modem de la RNU y por el otro, vía cables, con las casas de los clientes. Existen tres tipologías de RNU: RNU30, para servicios POTS, BRA y nx64k, RNU2M para servicios PRA y 2 Mb/s

transparente, y RNUN2M para servicios nx2 Mb/s con n hasta a 8. Una RNU está asociada a un módem del RASCO;

Tarjetas de RNU: tarjetas con las cuales es posible configurar la RNU en función de los diferentes servicios pedidos por los clientes;

Puertas de la tarjeta: son los conectores físicos de cada tarjeta.

En la figura 6 podemos ver una relación entre los recursos de acceso.

4. ARQUITECTURA DEL SISTEMA

La arquitectura del sistema para soportar los subprocesos de provisión de servicios y gestión de incidencias de servicios del proceso de gestión de servicios, contempla los siguientes módulos:

- Módulo de provisión de servicios (BDR),
- Módulo de gestión de incidencias de servicios (MAS),
- Módulo de activación de servicios (MAD) que tiene interfaces con el sistema de gestión de centrales locales (SG-CL) y con el sistema de gestión de la red de acceso (SG-RA).

En la figura 7 se ilustra la arquitectura del sistema.

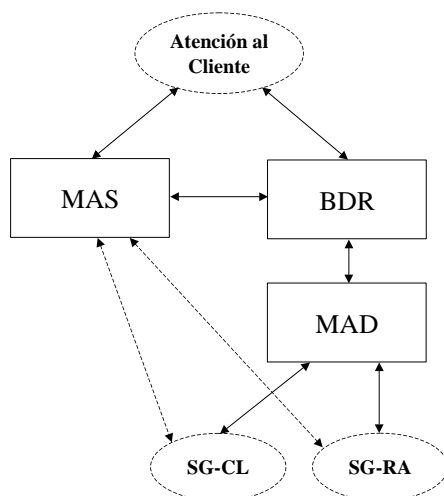


Figura 7 – Arquitectura del Sistema

4.1 Modulo de Provisión de Servicios (BDR)

Implementa la función de provisión de un servicio en base a los recursos disponibles en la red según un proceso definido en estados. Se compone básicamente de tres elementos:

- Una **base de datos (BDR)** que guarda información sobre los recursos presentes en la red y el estado en el que se encuentran, desde aquellos que están planificados pero no implementados realmente hasta los que han sido activados y se hallan prestando servicio.

- Un **workflow** que establece las fases por las que debe pasar la provisión del servicio hasta su puesta en funcionamiento, en el que se definen tanto las etapas que lo componen como los usuarios que deben operar en cada una de ellas, así como la información de han de gestionar en cada caso.

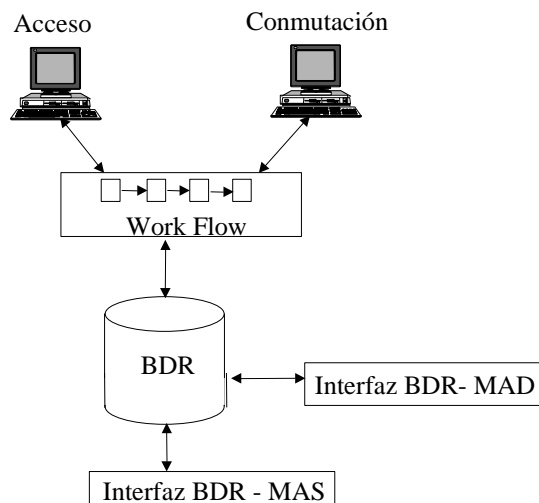


Figura 8 – Componentes de la BDR

- Una **interfaz de usuario**, característica de cada uno de los perfiles establecidos en el workflow, que presenta la información y los controles necesarios para la operación de provisión de red.

Diferentes tipos de servicio implican la utilización de distintos recursos, y variaciones en el sistema de estudio previo y construcción de red correspondientes. Este módulo, por tanto, ha de presentar la flexibilidad necesaria para ajustarse a las características propias de todos ellos, tanto en la modelización de la información, como en la definición del proceso de provisión que hace el workflow, procurando maximizar al mismo tiempo la integración de estos elementos en estructuras que optimicen la gestión del sistema.

4.2 Modulo de Gestión de Incidencias de Servicio (MAS)

El modulo de gestión de incidencias (MAS) se compone de las siguientes aplicaciones: una aplicación que contiene toda la información del cliente afectado y de los servicios contratados por éste que permita crear los oportunos partes de Reclamación de Cliente (gestor de partes) con su base de datos (BD-MAS) para la gestión de los diferentes estados de los anteriores partes, posteriormente notificando el cambio de estado de los mismos, mediante un interfaz común, a la aplicación anterior, y que permitirá, por otra parte, crear partes de Alarma de Red (gestor de alarmas). Esta segunda aplicación tiene un interfaz con BDR donde esta el inventario de todos los equipos de red y los servicios que se están ofreciendo, para así poder obtener de la relación entre equipos afectados

por una alarma de red y los servicios involucrados en este fallo.

La interrelación entre los diferentes departamentos y sistemas de información involucrados es la que se muestra en la figura 9.

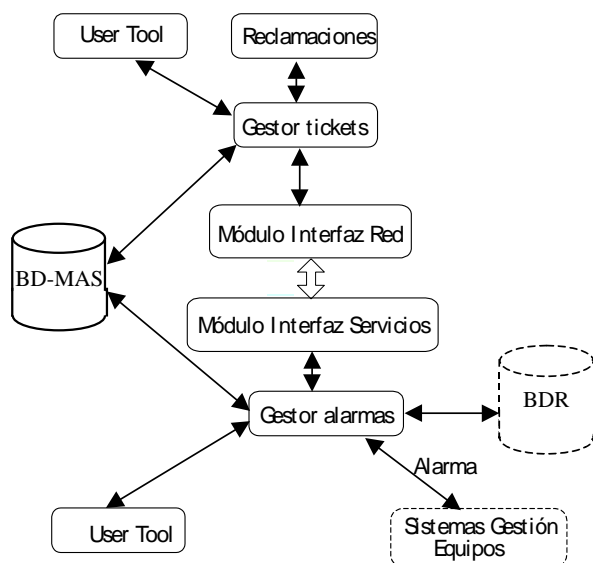


Figura 9 – Módulos del MAS

4.3 Modulo de Activación de Servicios (MAD)

MDA se compone de cuatro componentes:

- El principal proceso es MDA-CL, que automáticamente obtiene las Órdenes de Servicio de BDR y envía Peticiones de Servicio al sistema de gestión de las centrales locales;
- El MDA GUI que permite controlar el proceso principal;
- El MDA-RA que es lanzado por el operador de la red de acceso para cada actividad de configuración, obtiene la Orden de Servicio de la base de datos BDR y envía la Petición de Servicio al sistema de gestión de la red de acceso;
- El MH-GUI (Manual Handler GUI) que permite a un operador manejar manualmente las Peticiones de Servicio.

La relación entre estos componentes se muestran en la figura 10.

Como funcionalidad añadida, MDA-CL y MDA-RA pueden simular la ejecución, ejecutándose en un modo especial que no interactúa con el sistema de gestión de las centrales locales ni con el sistema de gestión de la red de acceso. Esta capacidad es utilizada para ayudar a los operadores en las tareas que requieren de su intervención.

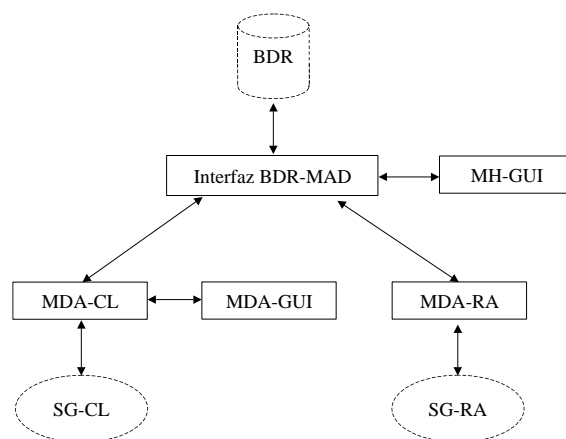


Figura 10 – Módulos del MDA

Para la configuración de las centrales locales, el motor de MDA-CL interactúa por una parte con la BDR y por otra con el sistema de gestión de las centrales locales. En el primer caso, utiliza ODBC como protocolo de comunicación, y una tabla intermedia en la que la BDR deposita las peticiones y el motor de la aplicación deposita los resultados. En la comunicación con el sistema de gestión de las centrales locales se utiliza el protocolo TELNET sobre una conexión TCP/IP. La interfaz gráfica de usuario (GUI) utiliza una conexión TCP para comunicarse con el motor de la aplicación.

Es importante señalar que el motor funciona de forma autónoma, realizando una consulta periódica al interfaz BDR. Este procedimiento se ilustra en la figura 11.

La configuración del sistema de gestión de la red de acceso se lleva a cabo mediante un módulo separado (MDA-RA) que se comunica con la interfaz del sistema de gestión de la red de acceso proporcionada utilizando una librería de enlace dinámico (DLL) proporcionada por el fabricante. Esta librería utiliza, a su vez, CORBA para completar la configuración. La comunicación con la BDR se lleva a cabo de la misma manera que en el caso de la comunicación MDA-CL: se utiliza el protocolo ODBC para consultar una tabla intermedia en la que se depositan las peticiones y los resultados. Este procedimiento se ilustra en la figura 12.

A diferencia de la configuración del sistema de gestión de las centrales locales, la configuración del sistema de gestión de la red de acceso no se ejecuta de forma autónoma, si no que viene iniciada desde la BDR por petición explícita (lanzando un ejecutable), y procesa una petición de servicio en cada ejecución.

5. CONCLUSIONES

Un seguimiento del estado del arte de los estándares de gestión en el nivel de servicio nos permite destacar la enorme labor que aún queda por desarrollar para la normalización de la gestión de servicio. El entorno altamente competitivo en el que se mueve el sector obliga a los operadores emergentes a diseñar

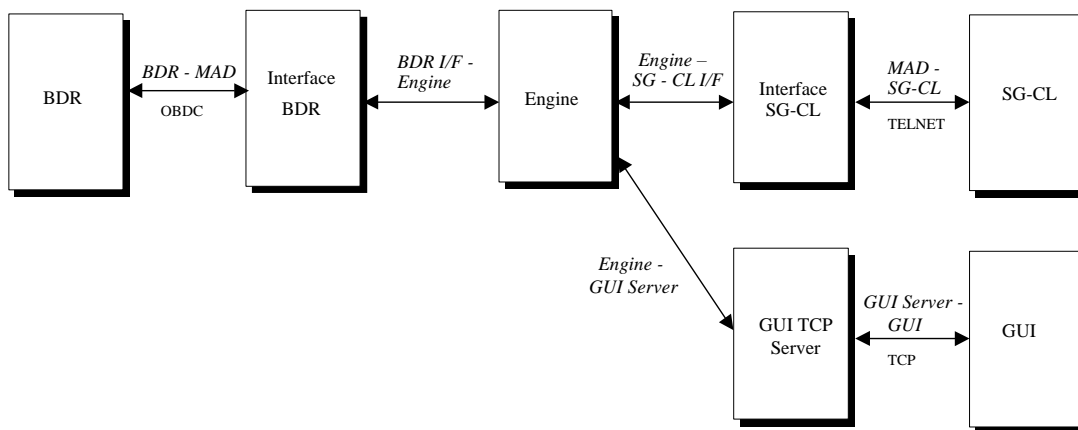


Figura 11 - Comunicación entre los diferentes módulos de MDA-CL

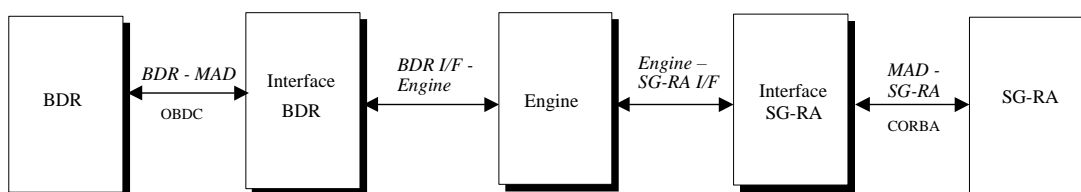


Figura 12 - Comunicación entre los diferentes módulos de MDA-RA

arquitecturas adelantándose a las labores de estandarización y al mismo tiempo, proponiendo soluciones flexibles y que se puedan ir incorporando los cambios que el mercado demanda para la creación de nuevos servicios y nuevas redes.

En este sentido, un sistema de gestión de servicios debe ser modular y puede estar compuesto por un módulo de planificación de la provisión donde se soporta la gestión del proceso de provisión y los recursos implicados, el módulo de activación de servicios que es el responsable de dialogar con los respectivos sistema de gestión de nivel de red o elemento de red para la configuración y activación de los servicios, y el módulo de aseguramiento de servicio que soporta el proceso de reclamaciones de servicios afectados por incidencias de la red.

A pesar de que el sistema descrito se centra en la gestión de servicios de voz y con una tecnología de red de acceso radio punto a multipunto, su arquitectura se puede extender a todo tipo de servicios y tipos de redes, residiendo la dificultad de la totalidad de su implementación en el modulo de activación de servicios, ya que depende de los mecanismos de interfaz que suministran los sistemas de gestión de los servicios y redes que se quieran utilizar.

6. REFERENCIAS

[1] Bagnasco E., Geymonat M. "The Impact of Telecommunications in Europe and the Requeriment for Network Management". Chapter 8. Telecommunications Network Management.

Tecnologies and Implementations. Edited by Salah Aidarous and Thomas Plevyak. IEEE Series on Network Management. 1998

- [2] Koehn R., del Ser L., Sánchez D., Guerrero C., Carneiro V., "Gestión Avanzada de una Red Corporativa de Telecomunicaciones", 1º Jornadas de Ingeniería Telemática (JITEL97), Bilbao – España, Setiembre 1997.
- [3] Glitho R. H., Hayes S. "Telecommunications Management Network: Vision vs Reality". IEEE Communications Magazine. March 1995.
- [4] Del Ser L., Koehn R., Sanchez D., Guerrero C., Viña A., "Gestión TMN, Vía para Cumplir Objetivos Corporativos" - Revista Computing nº 124 - Febrero 1998.

Capítulo 3A: Modelado de Tráfico

TREESOL: Algoritmo para la Optimización de la Capa Física de una Red de Telecomunicación Implementada de Forma Propietaria.

J.R. Salas Blanco, K.D.-Hackbarth (U. Cantabria)

Modelos y métodos para el diseño y enrutamiento de tráfico de redes de conmutación de circuitos.

A. Cuadra Sánchez, K.D.-Hackbarth (U. Cantabria)

Modelado Markoviano de Tráfico Agregado ATM.

A. Jurado, E. Casilari, A. Reyes, A. Díaz-Estrella y F. Sandoval (U. Málaga)

Modelado Wavelet de tráfico de vídeo VBR

E. Casilari, A. Reyes, A. Díaz-Estrella y F. Sandoval (U. Málaga)

Un estudio por simulación del efecto de la correlación en el retardo medio del tráfico asíncrono en una red FDDI.

J.C. López Ardao, C. López García y A. Suárez González (U. Vigo)

Técnicas eficientes de filtrado y análisis de tráfico para la monitorización continua de redes de comunicaciones.

J.J. Ruiz, E. Magaña, J. Aracil, J. Villadangos (U. Navarra)

Disciplina de Planificación Basada en Rondas con Disminución de la Distorsión del Patrón de Tráfico.

J.M^a Quinteiro, A. Sarmiento, A. Santana, F. Cabrera (U. Las Palmas)

TREESOL: Algoritmo para la Optimización de la Capa Física de una Red de Telecomunicación Realizada de Forma Propietaria

JOSE RAMON SALAS, KLAUS D. HACKBARTH
Grupo de Ingeniería Telemática / Departamento de Ingeniería de Comunicación
Universidad de Cantabria
Avda. Los Castros s/n, 39005 Santander (CANTABRIA)
Telf: 942-201392 Ext.14, Fax: 942-201488
E-mail: jramon@tlmat.unican.es, klaus@tlmat.unican.es.

Abstract

This paper shows an algorithm for the optimization of a network topology under cost minimization criteria and biconnectivity conditions for all or some of its couples of nodes. The cost of each link is expressed by a concave increasing cost function with a large fixed charge, which is usual in case that the operator implements his own infrastructure (proper realisation). The algorithm allows treating large networks (hundreds of nodes) with a regular personal computer in a reasonable amount of time. The algorithm was applied for the determination of the transmission infrastructure for telecommunication networks.

1. Introducción

Las redes de telecomunicación se dividen normalmente en dos capas bien diferenciadas: una capa lógica y una capa física. En redes de conmutación de circuitos, la capa lógica se compone de una serie de conmutadores y una estructura de grupos de circuitos de 2 Mbps (E1), cuyo diseño y dimensionamiento se exponen en [1]. La importancia de la capa física reside en la integración de la demanda de todas las capas lógicas que implementa un operador (RDSI, Frame Relay, ATM, etc.) y de los circuitos alquilados a terceros, por ejemplo redes corporativas. Debido a los altos costes de la infraestructura de medios de transmisión, la capa física supone cerca del 70% del coste total de una red de telecomunicación, sobre todo cuando la red tiene una cobertura extensa (WAN).

El punto más importante en el diseño de la capa física es determinar su topología, esto es, fijar el número de enlaces y su lugar geográfico. Para ello, hay que tener en cuenta que el coste de un enlace no crece linealmente con el flujo que le atraviesa, sino que se describe aproximadamente con una función cóncava. Esto último, junto a los altos costes fijos de la instalación de una infraestructura, supone que este problema de optimización no es lineal y es combinatorialmente duro [2].

Los algoritmos tradicionales están basados en el método denominado “drop-out”. Dicho método consiste en la reducción de una red inicial mediante la eliminación sistemática de enlaces hasta que no produzca ninguna reducción en el coste total de la

solución, o no se pueda eliminar ningún enlace más por razones de conectividad.

Este tipo de método necesita el encaminamiento de todas las demandas cada vez que se elimina un enlace, por lo que se emplea mucho tiempo de proceso, especialmente en ejemplos grandes. De hecho, este tipo de método sólo se puede utilizar en ejemplos de pequeño – mediano tamaño (menos de 50 nodos) [2]. Otra desventaja de este tipo de algoritmos es que sus resultados son poco óptimos en los casos en que los costes fijos sean altos. Ejemplos de estos algoritmos se pueden encontrar en [3], [4], [5] y [6].

El nuevo algoritmo TREESOL tiene el objetivo de optimizar la capa física de una red de telecomunicación bajo la condición de que la solución tenga un coste mínimo. Otro de los objetivos de este algoritmo es que debe utilizar el menor tiempo de cálculo posible, incluso en ejemplos de redes con un número grande de nodos (cientos de nodos). Además, hay que tener en cuenta que, por lo general y por motivos de seguridad, las redes de telecomunicación necesitan dos caminos diferentes para encaminar cada demanda entre dos nodos; por lo que la solución ofrecida por TREESOL debe ser la de una topología, al menos, biconectada, esto es, entre cada pareja de nodos deben existir dos caminos totalmente diferentes.

2. Función de coste

La función real de coste de los enlaces de una red de telecomunicación es una función discontinua

en escalera, debido a las capacidades escaladas de los sistemas de transmisión empleados. Cuando se realiza una optimización de la topología no se suele utilizar este tipo de función. En su lugar se emplea una aproximación de esta consistente en una función cóncava monótona creciente con costes fijos, la cual es ampliamente utilizada en el diseño de topologías de redes de telecomunicación [4], [6]. Dicha función viene dada por la siguiente expresión:

$$f(X) = a_m * X_m^\alpha + b_m$$

donde :

a_m = coste por unidad de flujo dependiente de la longitud del enlace m .

b_m = coste fijo dependiente de la longitud del enlace m .

X_m = flujo que atraviesa el enlace m .

α = parámetro de concavidad [0,1].

El valor de los parámetros de esta función refleja el esquema de coste bajo el cual se optimiza la capa física de la red. Un esquema de costes con líneas alquiladas, por ejemplo una red corporativa, es aquel en el que el coste por unidad de flujo es alto (parámetro a) y los costes fijos son bajos (parámetro b). En cambio, bajo el esquema de costes en el cual se implementa de forma propietaria la red de telecomunicación (realización propia) ocurre lo contrario, el parámetro a es pequeño y el parámetro b muy alto. En la Fig.1 se puede observar las diferencias de ambos esquemas de costes para una longitud de enlace concreta.

Como se puede observar en la Fig. 1, a partir de un cierto flujo, el cual es muy bajo debido principalmente al esquema de tarificación de los circuitos alquilados; siempre es más económico implementar el enlace de forma propietaria. Este hecho pone de manifiesto lo elevado de las tarifas de las compañías europeas de telecomunicación. Precisamente optimizar una red de telecomunicación bajo este esquema de coste con realización propia es el objetivo principal del algoritmo TREESOL. De esta forma, este nuevo algoritmo también puede ayudar a un operador a evaluar su política de tarificación del alquiler de sus capacidades dentro de un entorno competitivo.

3. Algoritmo TREESOL

Como se ha comentado en la sección introductoria, el algoritmo TREESOL tiene el objetivo de optimizar la capa física de una red de

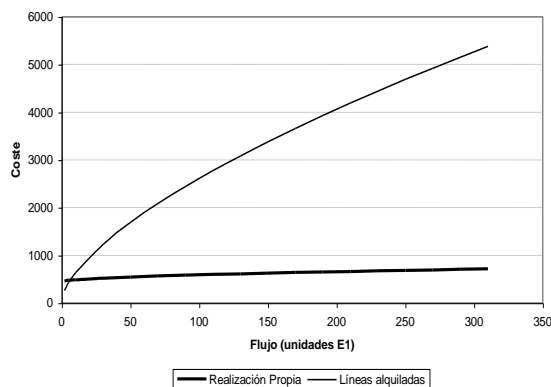


Fig.1: Esquemas de coste con líneas alquiladas y realización propia.

telecomunicación, de forma que la solución tenga un coste mínimo y que cumpla los requerimientos de biconectividad.

Además, se pretende que el algoritmo utilice el menor tiempo de proceso posible, incluso en ejemplos con un gran número de nodos.

Para el desarrollo del algoritmo se ha supuesto que la topología óptima de una red de telecomunicación sin requerimientos de biconectividad, es un árbol. Esta suposición es cierta en aquellos casos donde se tenga un coste fijo elevado, esto es, bajo un esquema de costes con realización propia.

En cambio, si el coste fijo es pequeño, caso del esquema de coste con líneas alquiladas, la suposición no se cumple para todos los casos, y los costes de las soluciones obtenidas con TREESOL pueden ser superiores a los obtenidos con los algoritmos de tipo "drop-out".

Por tanto, la base principal del algoritmo es calcular en primer lugar un árbol con coste óptimo, el cual es una estructura que no cumple los requerimientos de biconectividad, al cual se le añaden enlaces hasta cumplir con dichos requerimientos. La selección de estos enlaces que se van a añadir al árbol se realiza a través de unos determinados criterios:

- Longitud del enlace, asumiendo que la función de coste tiene un coste fijo alto.
- Suma de la demanda que tiene como origen o destino alguno de los dos nodos que forman el enlace.
- Contribución al aumento de biconectividad de la red.

Con estos criterios, el algoritmo se divide en cuatro pasos diferentes:

- (1) Cálculo de un árbol inicial
- (2) Cálculo de un árbol con coste reducido
- (3) Mallado del árbol
- (4) Mejora

A continuación se comenta cada uno de los pasos de los que se compone el algoritmo TEESOL junto con un pequeño ejemplo de 10 nodos que representan las principales capitales de provincia españolas.

La Fig. 2 muestra un posible grafo inicial de esta red de 10 nodos a la que se le va a optimizar su topología. Se puede ver que es una red altamente mallada, a la cual se le eliminarán la mayoría de sus enlaces.

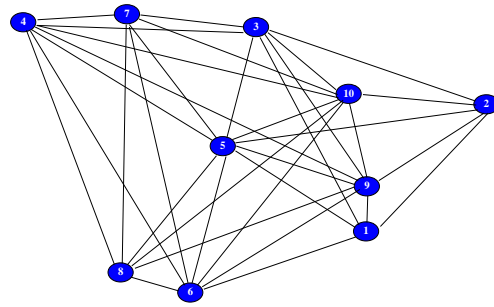


Fig.2: Estructura inicial de la red a optimizar.

En dicha figura, se puede observar como el árbol obtenido se asemeja a una topología en estrella. Esto es debido a que el nodo central (nodo 5) tiene una demanda muy alta, por lo que los enlaces que contengan este nodo serán los que tengan el coste más bajo.

3.1 Cálculo de un árbol inicial

Existen muchos algoritmos que calculan un árbol mínimo a partir de una estructura inicial, pero en este paso, la dificultad reside en asignar un coste correcto a cada enlace de la estructura inicial. Para ello se asigna a cada enlace un coste artificial que relaciona la longitud del enlace y una estimación del flujo que lo atraviesa. El coste asignado a cada enlace viene dado por la siguiente expresión:

$$C_m = \frac{l_m}{X_m^{\alpha_1}}$$

donde:

- C_m = coste artificial por unidad de flujo.
- l_m = longitud del enlace m .
- X_m = flujo que atraviesa el enlace m .

La utilización de este coste artificial viene determinada por la posibilidad que tiene el usuario de variar la influencia del flujo estimado a través del parámetro $\alpha_1 = [-1, 2]$. De esta forma, se le puede dar más importancia a la longitud del enlace, ó al flujo que le atraviesa.

Una vez asignados los costes a cada enlace, se calcula el árbol mediante un algoritmo de coste mínimo conocido, como es el algoritmo de PRIM [7]. La Fig.3 muestra el árbol obtenido de este paso para un valor de $\alpha_1 = 1$.

3.2 Cálculo de un árbol con coste mínimo

Puede ocurrir, de hecho ocurre en la mayoría de los casos, que el árbol calculado en el paso anterior no tenga un coste mínimo. Por este motivo se incluye un procedimiento que mejora el coste de un árbol dado hasta que se consiga un árbol con coste mínimo, o al menos una buena aproximación a este.

Para ello se sustituyen los enlaces con los costes más altos del árbol inicial por otros que produzcan una reducción de coste. Por motivos de eficiencia computacional, sólo se consideran como candidatos aquellos enlaces que formen, junto con dos enlaces del árbol inicial, un anillo básico de tres enlaces [6].

Una vez se realiza una sustitución, se encamina toda la demanda para calcular el coste del nuevo árbol. Si este coste es menor que el de la estructura anterior, el procedimiento continua realizando otra sustitución. En caso contrario, se deshace la sustitución y se prueba otro candidato. El procedimiento termina cuando no se pueda realizar ninguna sustitución que produzca una reducción de coste.

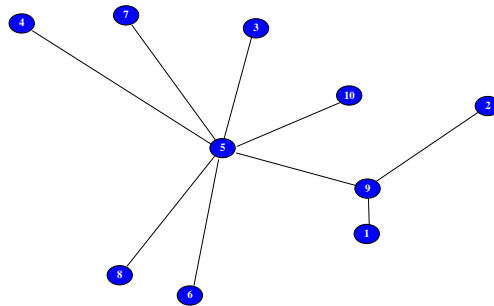


Fig. 3: Arbol inicial

La Fig. 4 muestra el árbol con coste mínimo obtenido de este paso. Se puede observar en dicho árbol que sus enlaces son más cortos que en el árbol de la Fig. 3. Esto es debido a que con una función de coste con un coste fijo alto, es la longitud del enlace la que determina su coste real, por lo que con este paso se tiende a calcular el árbol mínimo en distancia.

En este ejemplo concreto, el cual tiene unos costes fijos elevados, se podría haber llegado a esta misma solución si se hubiese utilizado $\alpha_l = 0$, con lo que sería el algoritmo de PRIM el que calcularía el árbol mínimo en distancia. De esta forma se puede evitar la ejecución de este segundo paso, especialmente en ejemplos pequeños con demanda baja, obteniendo la solución más rápidamente. En ejemplos medianos y grandes esto no siempre es posible, debido principalmente a tener unos flujos muy altos y una influencia reducida de los costes fijos.

3.3 Mallado del árbol

Cuando finaliza el paso (2), se obtiene una estructura en árbol. En el caso de que no existan requerimientos de biconectividad, el algoritmo TREESOL terminaría en ese punto. Pero en la mayoría de los casos esto no es así y se debe proporcionar un segundo camino a aquellas demandas que lo requieran. Este es el objetivo de este paso, añadir enlaces al árbol mínimo hasta que se cumplan los requerimientos de biconectividad.

Debido a los altos costes fijos, se pretende que este procedimiento añada el menor número de enlaces posible, y que además, esos enlaces sean lo mas cortos posibles. El hecho de añadir un enlace a un árbol provoca que se forme un anillo que proporciona biconectividad a todos los nodos que contiene.

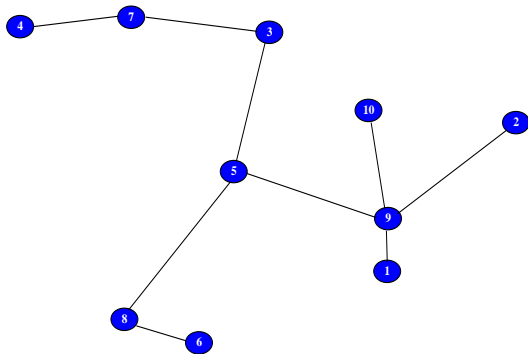


Fig. 4: Arbol con coste mínimo

Para añadir enlaces, se asigna un peso a cada enlace que no pertenezca al árbol mínimo que relaciona la longitud del enlace y el tamaño del anillo que forma al añadirle al árbol. Este peso viene determinado por la siguiente expresión:

$$p_m = \frac{l_m}{ib_m^{\alpha_2}}$$

donde:

l_m = longitud del enlace m .

ib_m = incremento de biconectividad. Este valor refleja el tamaño del anillo formado al añadir el enlace m .

Como en el procedimiento de 3.1, este peso puede ser variado por el usuario a través del parámetro $\alpha_2 = [0,2]$, de tal forma que se le puede dar más importancia a la longitud del enlace ó al tamaño del anillo. Bajo el esquema de coste con costes fijos altos, se pretende añadir el menor número de enlaces posibles, por lo que los anillos que se formen deben ser lo más grandes posibles. Pero al mismo tiempo también se desea que esos enlaces sean cortos, por lo que una solución de compromiso es utilizar un valor $\alpha_2 = 1$.

Aquel enlace cuyo peso sea mínimo se añade al árbol. Este proceso continua hasta que se cumplan todos los requerimientos de biconectividad. La Fig. 5 muestra la estructura resultante después de este paso. En dicha figura se observa que se han añadido cuatro enlaces al árbol de la Fig. 4, lo que ha proporcionado la formación de cuatro anillos independientes, los cuales proporcionan un segundo camino a cada pareja de nodos.

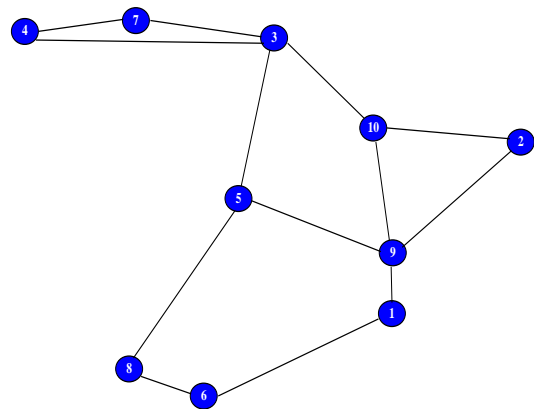


Fig. 5: Estructura resultante después del mallado del árbol

3.4 Mejora

Con el paso (3) se obtiene una solución factible que cumple con todos los requerimientos y se puede implementar. Pero puede darse el caso de que se pueda reducir el coste de la solución obtenida en el paso anterior, eliminando para ello algunos enlaces sobrantes, sin que esto signifique el incumplimiento de los requerimientos de biconectividad. Este es precisamente el objetivo de este procedimiento.

Una condición necesaria, pero no suficiente, que debe cumplir un enlace que se intenta eliminar, es que en los nodos que le forman deben incidir al menos tres enlaces. La condición suficiente para eliminar un enlace es que no debe romper la biconectividad. Esta condición se cumple si los nodos que forman el enlace pertenecen a un mismo anillo después de eliminar dicho enlace. Los enlaces $e(5,9)$ y $e(9,10)$ de la Fig. 5 cumplen las dos condiciones, por lo que estos enlaces se pueden eliminar sin que ello afecte a la biconectividad.

Este procedimiento elimina un enlace que cumple las condiciones anteriormente mencionadas, para posteriormente calcular el coste de la topología resultante. Si este coste es menor que el de la topología anterior se intenta eliminar otro enlace. De esta forma este procedimiento continúa hasta que no se puedan eliminar más enlaces. Como puede observarse, se vuelve a utilizar el principio “drop-out” comentado en la introducción. Este método es costoso en cuanto a tiempo de ejecución, pero en esta ocasión se aplica a una estructura muy reducida, por lo que el tiempo empleado en este paso es aceptable.

En la Fig. 6 se muestra la solución final a este ejemplo de 10 nodos. En dicha figura se puede observar que la topología resultante tiene pocos enlaces, exactamente 11, los cuales forman dos anillos. Hay que tener en cuenta que en este ejemplo, el número mínimo de enlaces que se pueden tener para cumplir los requerimientos de biconectividad son 10, los cuales formarían un solo anillo, aunque esto no quiera decir que sea la topología con coste mínimo.

4. Resultados

Los resultados mostrados a continuación dan una idea del comportamiento temporal del algoritmo, así como del coste de la solución obtenida. Estos resultados se han obtenido ejecutando el algoritmo en un PC con procesador Pentium II a 233 MHz y 64 MB de memoria RAM.

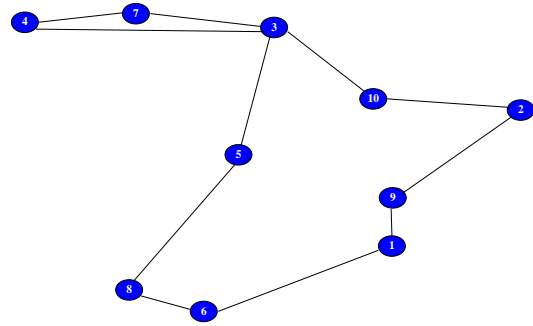


Fig. 6: Estructura resultante después de ejecutar el algoritmo TREESOL

En la Tabla I y en la Fig. 7 se muestra el tiempo de proceso empleado por el algoritmo TREESOL dependiendo del número de nodos de la red a optimizar. Para ello se han utilizado varios ejemplos de diferente tamaño desde 100 nodos hasta 1000, en los que el número de demandas y enlaces iniciales crece de forma aproximadamente lineal con el número de nodos, lo cual es habitual en las redes reales. A partir de esos resultados se obtiene que el tiempo empleado por TREESOL depende del número de nodos de la forma $t = a \cdot n^3$, siendo n el número de nodos y a una constante. La curva obtenida con esta fórmula se presenta junto a los datos de la Tabla I en la Fig. 7.

Tabla I: Tiempo de proceso en función del número de nodos.

Nº nodos	100	250	500	750	1000
Nº enlaces iniciales	450	1025	2300	3700	5900
Nº demandas	1050	2800	5050	7800	10600
Tiempo	13''	2' 54''	22' 10''	75' 54''	184' 14''

En cuanto al coste de la solución obtenida, se comparan los resultados obtenidos para diferentes ejemplos con el algoritmo TREESOL, con los obtenidos con los algoritmos que utilizan el método “drop-out” de las referencias [3] y [5], denominados LOPSOL y CIISOL respectivamente. En la Tabla II se muestra el coste por unidad E1 de demanda (en miles de Euros) para tres ejemplos diferentes y para los tres algoritmos; mientras que en la Tabla III se muestra el tiempo empleado por cada algoritmo para obtener la solución de cada ejemplo.

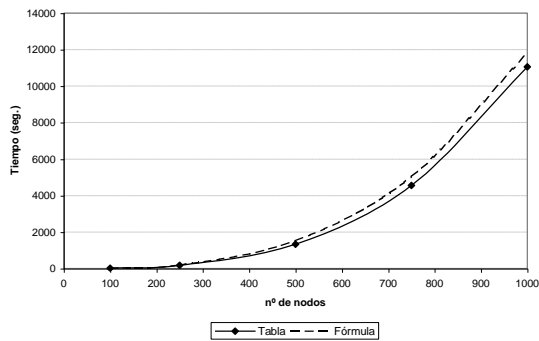


Fig. 7: Tiempo empleado por TREESOL en función del número de nodos.

Como se puede observar en ambas tablas, el algoritmo TREESOL es el que ofrece la mejor solución (la más barata) en los tres casos además de obtener la solución en un tiempo muy razonable.

Finalmente, cabe mencionar que la biconectividad se fija individualmente para cada nodo, lo que permite calcular topologías mixtas en las que una parte (la red de acceso) puede requerir solamente uni-conectividad en forma de un árbol, mientras la otra parte (la red dorsal) requiere biconectividad. A modo de ejemplo, la Fig. 8 muestra la optimización con TREESOL de una red española con 8 nodos dorsales. Cada uno de ellos forma un cluster con los nodos de acceso que tiene asociado, los cuales, en esta ocasión, necesitan biconectividad, por lo que TREESOL les conecta entre sí en forma de anillos de acceso a la red dorsal.

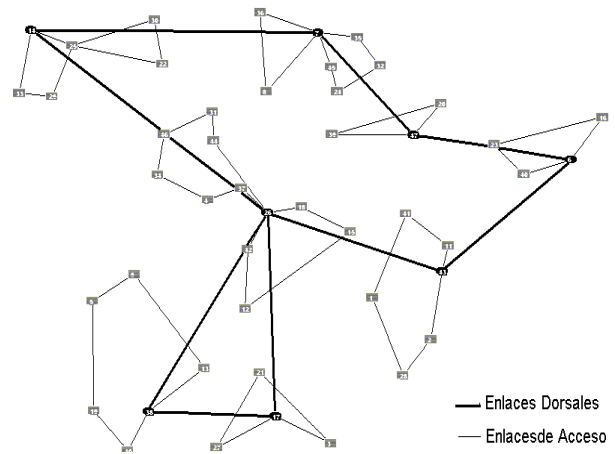


Fig. 8: Estructura resultante para un ejemplo de 47 nodos.

El algoritmo TREESOL se ha incluido en la herramienta informática para la planificación de redes corporativas denominada CONPAT (Corporate Networks Planification Tool), desarrollada por el Grupo de Ingeniería Telemática de la Universidad de Cantabria en colaboración con el operador alemán ARCOR Manesmann [8]. La incorporación de TREESOL a dicha herramienta completa la parte de la misma dedicada a la optimización topológica, la cual ya incluía los algoritmos basados en métodos “drop-out” de las referencias [3] y [5].

5. Conclusiones

Como se ha podido ver a lo largo de este documento, el algoritmo TREESOL realiza una optimización de la topología de una red de telecomunicación bajo un esquema de coste con costes fijos altos (realización propia). A partir de un umbral de demanda, esta forma de implementar la capa física de la red resulta más económica que alquilar los medios de transmisión a otro operador.

Los conceptos en los que está basado el algoritmo TREESOL, permite la obtención de una topología con pocos arcos, los cuales forman anillos en el caso de que, por motivos de seguridad, se necesiten dos caminos diferentes entre cada pareja de nodos. Este hecho, junto con los altos costes fijos, permiten pensar que la solución dada por TREESOL sea la más óptima.

Referencias

- [1] A. Cuadra, K. D. Hackbarth. “Modelos y Métodos para el Diseño y Enrutamiento de

Tabla II: Coste por unidad E1 de demanda obtenida con los tres algoritmos.

Nº nodos	25	47	101
LOPSOL	3.2	7.5	6
CIISOL	2.7 (-18%)	6.9(-10%)	5.7 (-9%)
TREESOL	2.7 (-18%)	6.4 (-17%)	4.8 (-24.5%)

Tabla III: Tiempo empleado por cada uno de los tres algoritmos.

Nº nodos	25	47	101
LOPSOL	0	4''	18''
CIISOL	57''	23' 28''	5h 26' 31''
TREESOL	1''	5''	2'

Tráfico en Redes de conmutación de circuitos".
JITEL'99, Madrid 1999.

- [2] M.Minoux. "*Network Synthesis and Optimun Network Design Problems: Models, Solution Methods and Applications*". Network, Vol. 19, 1989.
- [3] K.D. Hackbarth. "*Ein Verfahren zur Optimierung von Nachrichtennetzen mit Meehrwegeführung*". Zeirschrift fuer Operation Research, Vol. 27, 1983.
- [4] B. Yaged. "*Minimum Cost Routing for Static Network Models*". Networks, Vol. 1, 1971.
- [5] K. D. Hackbarth. "*A Heuristic Procedure for Calculating Telecommunication Networks*". Mathematic Programming study, nº 26, 1986.
- [6] M. Minoux. "*Reserche de la configuración Optimale d'un Reseau de Télécommunications avec Fonctions de Coût Concaves*". Annales de Télécommunication, Vol. 29, nº 1-2, 1974.
- [7] T. H. Cormen, C. E. Leiserson, R. L. Rivest "*Introduction to Algorithms*". MIT Press & McGraw-Hill, 1990.
- [8] K. D. Hackbarth, M. Mar Ortiz. "*X-CONPAT, una Herramienta para la Planificación de Redes Corporativas*". VI Jornadas I+D en Telecomunicaciones, Noviembre 1996.

MODELOS Y MÉTODOS PARA EL DISEÑO Y ENRUTAMIENTO DE TRÁFICO EN REDES DE CONMUTACIÓN DE CIRCUITOS

Antonio Cuadra
 Klaus D.-Hackbarth
 Universidad de Cantabria
 Departamento de ingeniería de comunicaciones. Grupo de Ingeniería Telemática
 Avda Castros S/N 39005-Santander
 E-mail: toni@tlmat.unican.es , klaus@tlmat.unican.es

Abstract

This paper shows a model and a corresponding tool for the configuration of a telecommunication trunk network based on a predetermined set of nodes, its total outgoing traffic and a set of design parameters. The model and the tool covers all steps for this task as traffic matrix generation, design of the network hierarchy, traffic routing and trunk group dimensioning. The tool incorporates additional functions to analyse both graphical and numerical values at the end of each step by allowing a correct selection of the design parameter values. The document outlines the possible applications of the tool and indicates its success in a regulation institution.

1. Introducción

Las redes de telecomunicación se dividen en tres planos, el plano de transporte de información, de control y de gestión. Cada plano se subdivide en capas correspondientes y el plano de transporte de información de esta subdivisión resulta una capa lógica y una capa física. En este trabajo se presenta un modelo y sus métodos correspondientes para el diseño y dimensionamiento de la capa lógica de una red de conmutación por circuitos sea en forma de una red pública (RTB/RDSI) o en forma de una red corporativa.

En primer lugar se expone el modelo de la red y un método correspondiente para su solución. Después se enseña una herramienta informática denominada TAROCA© (Traffic routing and cost analysis) y los aspectos de su implementación. Se informa sobre algunas aplicaciones de TAROCA© para el diseño del dimensionamiento y estudio de costes de redes concretas y al final se exponen las conclusiones y futuras líneas de trabajo

2. Modelos de red

La conmutación de circuitos digitales proporciona circuitos a 64 Kbps que a su vez se interconectan por medio de señales digitales sincronizadas a 2Mbps (denominadas grupos E1), cada una de las cuales alberga 32 circuitos. Estos 32 circuitos se dividen según la información que contengan en tres partes que coinciden con los tres planos: 30 circuitos para el transporte de la información de usuario, uno para transporte de señalización y otro para información de gestión. Para su uso óptimo en muchos casos la capa lógica se compone de una estructura jerárquica de hasta tres niveles. El nivel bajo se constituye por

nodos de acceso cuya misión es concentrar tráfico de lugares con baja carga y dirigirlo a un nodo superior (por razones de seguridad en algunas redes este acceso se realiza a dos nodos superiores). La red dorsal se puede subdividir a su vez en dos niveles y proporcionar un encaminamiento de tráfico entre uno y un máximo de tres caminos. Se observa que la constitución de esta jerarquía determina por completo el esquema de encaminamiento de tráfico (véase fig. 2.1).

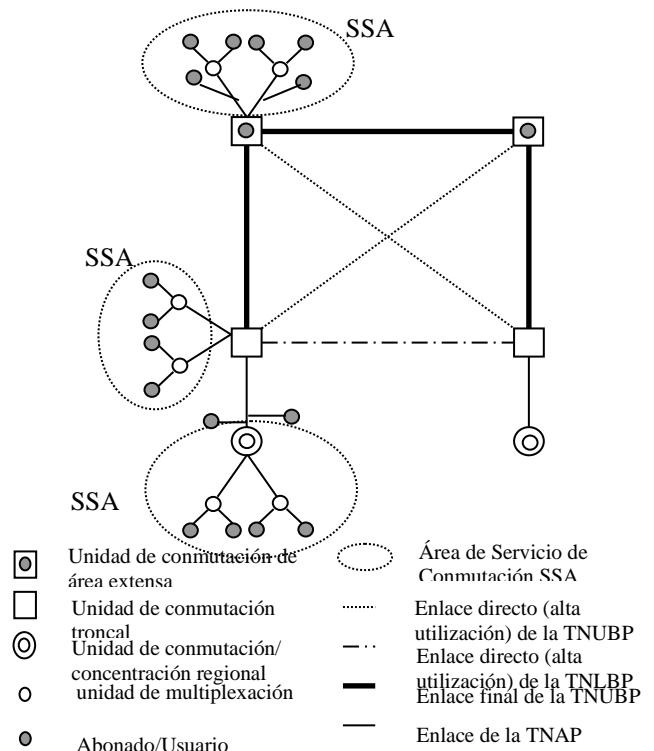


Figura 2.1 Esquema del encaminamiento de tráfico. De esta figura se extrae que se deberán dar dos tipos de división en la red, una para la parte de acceso

troncal (TNAP, trunk network access part) y otra para la parte troncal dorsal (TNBP, trunk network backbone part). Esta última se subdividirá en dos partes más: una dorsal inferior y otra dorsal superior (trunk network lower backbone part TNLBP y trunk network upper backbone part TNUBP).

Al final cada haz de grupos E1 entre dos nodos constituye un enlace de la capa lógica que se realiza actualmente en la capa física sobre equipos de la JDS (Jerarquía Digital Síncrona o SDH) y una infraestructura de cables ópticos o de radioenlaces. El enrutamiento de los grupos E1, resultado de la capa lógica, y su ensamblado a señales superiores viene dado por unos equipos en modo de transconectores de bajo y alto nivel (DX4/1,DX4/4) y multiplexadores de alto nivel para la parte dorsal y equipos del tipo de Multiplexores de extracción y inserción (ADMUX) para la parte de acceso. Las diferentes subcapas se muestran en la fig. 2.2.

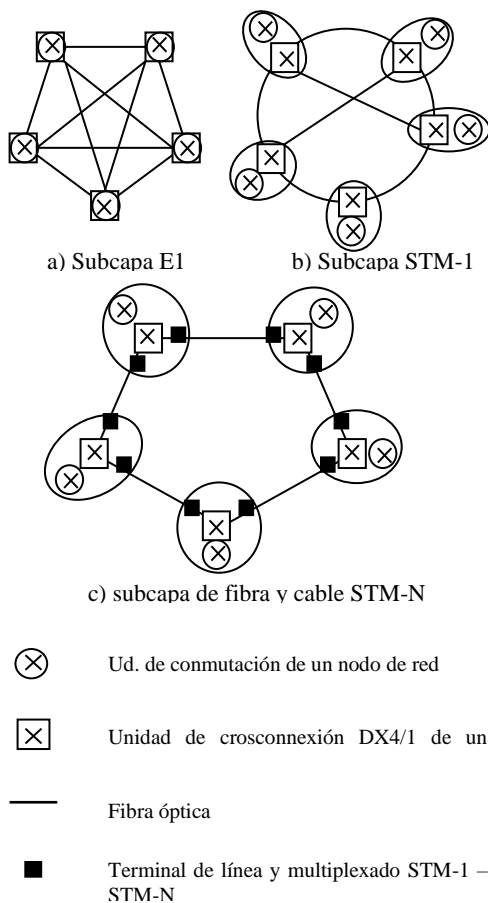


Figura 2.2 Subcapas a distintos niveles

Para el cálculo de la capa lógica se requieren como datos principales el conjunto de nodos a considerar con sus lugares topológicos en forma de sus coordenadas y una cifra por nodo que permite estimar el tamaño de tráfico de origen. Este tráfico se divide en dos partes, una que es el tráfico interno, y otra que es el saliente t_i^o que se distribuye entre el resto de nodos $v_j \neq v_i$. El valor de tráfico resultante t_{ij}^f de la demanda de tráfico del nodo v_i al v_j depende normalmente de la distancia geográfica entre v_i , v_j y

del peso del nodo destino. En *TAROCA*® se aplica la siguiente función genérica:

$$t_{ij}^f = c_i (t_j^o)^\beta * \max(l_{ij}, 1)^\alpha$$

donde α determina la influencia de la longitud y β determina la influencia del peso en forma del tráfico saliente del nodo destino. La constante c_i se calcula individualmente para cada nodo; se debe tener en cuenta que la suma de todas las relaciones de tráfico hacia afuera debe coincidir con el tráfico saliente del propio nodo.

En la mayoría de los casos esta matriz de tráfico no se puede realizar directamente por una capa lógica con enlaces entra cada pareja de nodos con una $t_{ij} > 0$ pero constituye el dato de entrada para fijar la subdivisión jerárquica. Esta subdivisión permite que el tráfico en distintas direcciones se concentre en enlaces comunes bajo el correspondiente esquema de encaminamiento de tráfico.

El encaminamiento de tráfico tradicional, que será el que consideraremos en este proyecto, lleva hacia una jerarquía de capa de red lógica donde el tráfico se enruta hacia enlaces comunes de alta utilización bajo una elevada pérdida de tráfico (típicamente del 10%) y un desbordamiento de este tráfico hacia otros enlaces de alta utilización o hacia la ruta final en último caso; los enlaces de la ruta final no permiten más desbordamiento y el dimensionamiento se habrá de implementar con el valor del GoS (1%), véase Girad [2].

Entonces el encaminamiento de tráfico y dimensionamiento de los grupos E1 se dividirá en cuatro partes:

- Primer encaminamiento de tráfico considerando la primera división de la red que determinará las capacidades de los enlaces en la TNAP y la carga del tráfico en la red dorsal malla completa.
- Segundo encaminamiento de tráfico para constituir y determinar las capacidades de los enlaces directos en la TNLBP.
- Tercer encaminamiento para constituir y determinar las capacidades de los enlaces directos de la TNUBP.
- Cuarto encaminamiento para las capacidades del enlace final en la TNUBP.

Algunos operadores introducen sobre todo en la TNUBP un encaminamiento dinámico no jerárquico (DNR, dynamic non-hierarchical routing). Este DNR da más flexibilidad en el encaminamiento dentro de los caminos finales y permite entonces la distribución de una sobrecarga local en forma equilibrada véase Gibb森 [1]. El DNR es un esquema que tiene sobre todo influencia en la operación de la red pero no cambia significativamente los valores en el dimensionamiento frente al cálculo actual. Las grandes redes de transmisión modernas tanto

públicas como privadas utilizan como arquitectura de red física sistemas y equipos JDS que la mayoría de los fabricantes proporcionan. La parte de transporte de esta red aplica la misma división y proporciona equipos, estructuras y topologías para ambas regiones de acceso y dorsal. De los costes de conmutación por un lado y de los costes de la capa física por otro se puede derivar una función que aproxima los costes de un E1 en cualquiera de los enlaces de la red dorsal y los costes de un nodo, aunque los costes reales sólo se pueden calcular exactamente a base de los equipos que requiere una configuración concreta. En este trabajo se utiliza la siguiente función de coste para la realización de un E1 en un enlace de la capa lógica:

$$c_{ij}^a = c_{i1}^a + c_{2l_{ij}}^a \text{ (Acceso)}$$

$$c_{ij}^d = c_{i1}^d + c_{2l_{ij}}^d \text{ (Dorsal)}$$

con:

- c_{ij}^a c_{ij}^d costes por grupo E1 entre el nodo v_i y el nodo v_j
- c_{i1}^a c_{i1}^d costes unitarios por grupo E1 debido a los costes en los nodos tanto de conmutación como de transconexión (equipos)
- c_{i2}^a c_{i2}^d costes unitarios por grupo E1 y por Km debido a los costes variables (fibra e infraestructura)
- l_{ij} distancia entre los nodos v_i y v_j .

A estos costes se deben añadir los costes fijos e independiente del flujo de tráfico y sus grupos E1 correspondientemente para el concentrador en un nodo de acceso c_f^a y un conmutador en un nodo dorsal c_f^d . Los costes totales de una configuración se obtienen por:

$$c^T = c^a + c^d$$

$$c^a = \sum_{(i,j)} c_{ij}^a + N^a \times c_f^a$$

$$c^d = \sum_{(i,j)} c_{ij}^d + N^d \times c_f^d$$

N^a , N^d número de nodos de acceso o dorsales respectivamente.

3. Herramienta TAROCA©

Para el diseño y dimensionamiento de una red de conmutación de circuitos basada en el modelo expuesto en la sección anterior se ha diseñado una herramienta informática denominada TAROCA© (Traffic Routing and Cost Analysis). TAROCA© sigue una secuencia conforme con este modelo:

- 1) Creación del ejemplo a estudiar mediante la selección del correspondiente fichero de datos.
- 2) Distribución del tráfico mediante el procedimiento TRADIS con el resultado de una matriz de tráfico.
- 3) División de la red troncal en parte de acceso (TNAP) y parte dorsal (TNBP) mediante el procedimiento de clasificación FCLASIG.
- 4) Primer encaminamiento de tráfico (FTRAROUT) para determinar la parte

correspondiente a la estructura en estrella de la red de acceso y la estructura mallada de la red dorsal, así como determinar el número de grupos E1 sobre los enlaces para la red de acceso.

5) División de la red dorsal en una parte inferior (TNLBP) y una parte superior (TNUBP) basándose en el tráfico calculado en el primer encaminamiento de tráfico: SCLASIG.

6) Segundo encaminamiento de tráfico STRAROUT para encontrar cargas de tráfico óptimas y para determinar el correspondiente número de grupos E1 sobre los enlaces directos entre nodos de la parte dorsal inferior (TNLBP), además de calcular el tráfico desbordado a la red dorsal superior (TNUBP).

7) Tercer encaminamiento de tráfico TTRAROUT para encontrar cargas de tráfico óptimas y para determinar el correspondiente número de grupos E1 sobre los enlaces directos entre nodos dorsales inferiores y superiores a excepción de los del trayecto final (que serían los de los nodos de la TNLBP y sus asignados correspondientes de la TNUBP), así como el tráfico desbordado a los enlaces finales.

8) Cuarto encaminamiento de tráfico FoTRAROUT para determinar la carga de tráfico y el correspondiente número de grupos E1 sobre los enlaces del trayecto final que serán los enlaces entre los nodos de la parte dorsal inferior y sus asignados de la parte superior más los enlaces de la parte dorsal superior entre sí.

9) Cálculo de costes para la parte de acceso y la parte dorsal.

La clasificación se realiza por el peso de los nodos dando a los nodos con más tráfico de entrada y salida el nivel más alto en la jerarquía. Como esa forma de clasificación puede resultar en algunos ejemplos una acumulación de nodos superiores, en una parte del algoritmo se considera un parámetro de distancia mínima que deben cumplir los nodos superiores entre sí. La asignación de los nodos inferiores a un nodo superior se realiza simplemente por el criterio de distancia mínima.

Se observa que la herramienta permite cualquier tipo de estructuración jerárquica sea en tres niveles o solamente en dos o incluso uno. Esta posibilidad da lugar a un conjunto muy amplio de diversas aplicaciones que se expone en la quinta sección.

4. Aspectos de implementación

La herramienta se compone de un conjunto de programas ejecutables y de una serie de ficheros de datos de entrada y de salida. En la versión actual de TAROCA© estos programas son independientes entre sí y se comunican por medio de ficheros de datos y a través de un proyecto común TAROCA© que se encarga de adaptarlos: Los datos requeridos en cada ejecutable se almacenan en estructuras de datos internas para minimizar el requerimiento de

memoria y de tiempo de ejecución, véase Hackbarth[3]. Este concepto permite una gran flexibilidad para el desarrollo del programa, sobre todo con vistas a futuras ampliaciones e igualmente permite que su implementación y comprobación se realice de manera sencilla debido a la independencia de los módulos ejecutables. Cada uno de los cuales han sido comprobados separadamente y cualquier corrección se limita al programa donde ocurra el error además las extensiones y los cambios en la especificación se reduce a modificar en la mayoría de los casos un único ejecutable.

Los diferentes ejecutables obtienen sus datos a partir de la estructura interna que le suministra el programa de adaptación correspondiente. Es tarea del programa de adaptación entablar la comunicación entre los dos ficheros de datos de entrada y las estructuras internas, y generar los cuatro ficheros de salida.

Los ficheros de salida se generan y/o se actualizan al finalizar cada uno de los programas ejecutables que son llamados desde el programa principal *TAROCA*® en el siguiente orden: *TRADIS*, *CLASIG* (primera aplicación), *FTRAROUT*, *CLASIG* (segunda aplicación) y *STFoTRAROUT*. Además las entradas de datos de cada subprograma emplean parte de los resultados de los pasos anteriores y por lo tanto *TAROCA*® utiliza los ficheros de salida como ficheros de datos intermedios que se van actualizando sucesivamente. De este modo un fichero intermedio va a tener tres estados posibles: creación, actualización y estado final con la información completa del fichero de salida correspondiente.

Cada ejecutable se compone de dos partes, el núcleo que realiza los algoritmos correspondientes y una concha que realiza la comunicación de datos entre las estructuras internas (en memoria) y los ficheros externos. Este principio de encapsulación permite un diseño independiente de ambas representaciones de datos (interna y externa) y un grado máximo de reutilización de los núcleos en otros entornos. La relación entre los ejecutables y los ficheros de datos se muestra en la figura 4.1.

Cuando un usuario hace una llamada al programa principal *TAROCA*® éste pregunta el nombre del escenario (o si es un escenario nuevo permite crearlo) para poder acceder a los dos ficheros de que está compuesto para poder visualizarlos o modificarlos. Además antes de ejecutarse cada subprograma permite modificar los parámetros relativos a él. Y en ciertos casos permite volver a ejecutar un procedimiento con distintos parámetros (como por ejemplo en *CLASIG*, con un número de nodos dorsales diferente).

El entorno *TAROCA*® proporciona acceso a los ficheros y ventanas de análisis para comprobar los resultados intermedios después de cada paso. Esta

facilidad permite al usuario controlar la influencia de algunos parámetros como puede ser el umbral para la distribución de tráfico en *TRADIS*. De este modo el usuario puede ajustar estos valores antes de continuar con el proceso de configuración de la red. La parte de análisis que *TAROCA*® proporciona tanto en forma alfanumérica como en forma gráfica se incorpora dentro de cada funcionalidad y opera con la estructura de datos interna para permitir un cálculo rápido y actualizaciones sencillas.

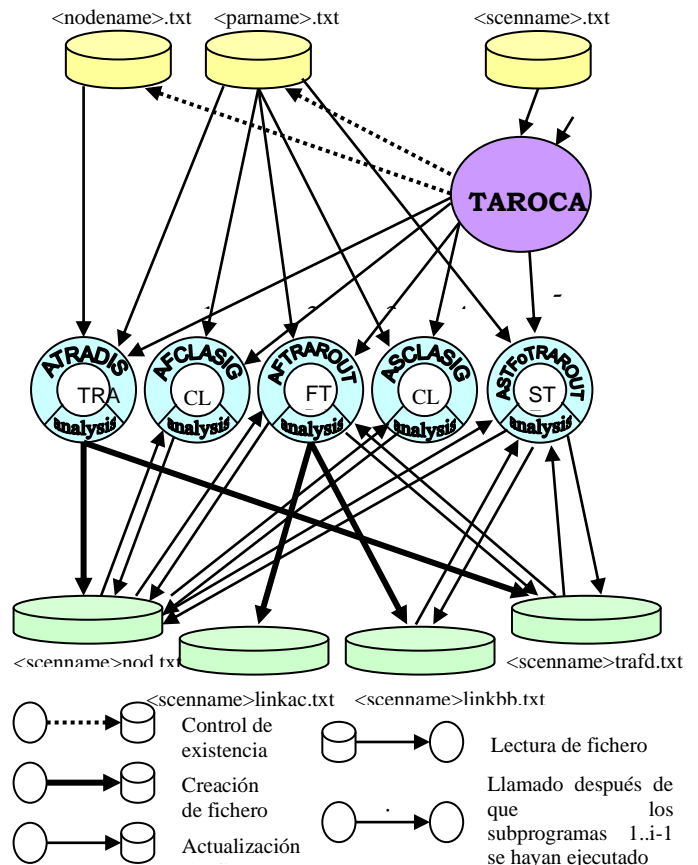


Figura 4.1 Estructura de *TAROCA*®.

5. Aplicaciones

La herramienta tiene varias aplicaciones que se pueden agrupar en tres áreas principales:

- Diseño y dimensionamiento de redes sean públicas o corporativas
- Estudios estratégicos sobre la realización y el encaminamiento en redes
- Estudios de costes y tarifas tanto por un operador como para una institución ajena como por ejemplo una comisión nacional de regulación del mercado

En el campo de diseño y dimensionamiento destaca la facilidad de estructurar la capa lógica en forma óptima y compararla con las siguientes estructuras extremas:

- a) todos los nodos de acceso salvo un nodo dorsal

- b) todos los nodos dorsales y de nivel superior
- c) todos los nodos dorsales con un único nodo superior

En el campo de estudios estratégicos destaca la facilidad de *TAROCA*® para permitir determinar un umbral para la constitución de enlaces de primera y segunda selección. Valores altos en estos umbrales producen una baja densidad en los enlaces hasta el extremo que no hay ningún enlace entre nodos dorsales de bajo nivel y todo el tráfico se desborda al nivel superior, resultando enlaces con alto número de grupos. El último permite formar grupos superiores al nivel E3 o incluso STM-1 evitando la transconexión en nodos intermedios de la ruta de la capa física.

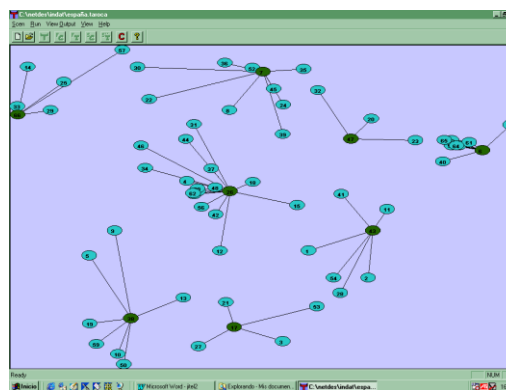


Figura 5.1 Ejemplo de España

Par estudiar en cada caso la influencia de diferentes capas lógicas a la realización de la capa física se requiere una herramienta correspondiente para la última tarea. Un ejemplo es la herramienta *CONPAT*® desarrollada en el GIT/UNICAN, véase Hackbarth-Ortiz[4], Salas-Hackbarth[5] y mediante un programa de adaptación se pueden transmitir los resultados de *TAROCA*® directamente a *CONPAT*®.

El campo de estudios de costes requiere la consideración de una red nacional completa de todos sus municipios que resulta dependiendo de la aplicación de varios miles de nodos. El diseño de *TAROCA*® y su implementación permite el tratamiento de estos ejemplos, hasta actualmente 6000 nodos.

Como aplicación concreta se menciona el estudio de la RTB alemana a base de 5000 nodos realizado por el instituto de estudios de regulación y mercado de telecomunicación WIK, Bad Honnef y encargado por la Comisión de Regulación del Mercado de Telecomunicación Aleman, Bonn, véase Schmidt[6].

Actualmente el GIT/Unican esta realizando una base de datos sobre los municipios españoles con sus posiciones geográficas y su número de habitantes que le permitirá diversos estudios en todas las áreas de aplicación tanto al nivel académico como al nivel de consultoría profesional. Como ejemplo se enseña la jerarquía en los nodos dorsales de los municipios más importantes de España, para un tráfico de 40mE/habitante y hora cargada. En la figura 5.1 aparece dicho ejemplo para un total de 66 nodos, 58 de los cuales dorsales inferiores y el resto dorsales superiores con una distancia mínima entre ellos de 200Km.

6. Conclusiones y futuras ampliaciones

Se ha expuesto un modelo y una herramienta informática para el diseño de la capa lógica de una red de telecomunicación y para estudios estratégicos o de costes y tarificación

Las futuras líneas de trabajo se basan en los comentarios recibidos durante la fase de "Request for Comments" al documento de la Comisión de Regulación Alemana [6]. Los puntos más importantes son:

- Mejora en los procedimientos de CLASIG para considerar limitaciones en la asignación de nodos inferiores a un nodo superior.
- Ampliación del modelo en distinguir entre enlaces directos en el TNLBP entre nodos con un mismo nodo superior asignado y enlaces directos en el TNLBP entre nodos asignados a diferentes nodos superiores.
- Umbrales máximos de ocupación de los grupos E1 para cada tipo de enlace.
- Introducción de nodos especiales de interconexión de la capa lógica con nodos de otro operador para estudios de interconexión, su influencia en la estructura de la capa lógica, de la matriz de tráfico y de los costes incluyendo estudios de coste y tarifas de interconexión.

Referencias

- [1] Gibbsen et al. " *Dynamic Alternative Routing, in Routing in Communication Networks* ", ed M. Steenstrup. Prentice Hall 1995.
- [2] Girard A. " *Routing and Dimensioning in Circuit Switched Networks* ". Addison Wesley 1990.
- [3] Hackbarth K.. "Implementation of Algorithm for Telecommunication Network Planning, in Graphtheoretic Concepts in Computer Science", ed U. Pape. Trauner Verlag, Austria 1984.
- [4] Hackbarth-Ortiz. " *XCONPAT, una herramienta para la planificación de redes corporativas* ".

Acta de las VI Jornadas de I+D en Telecomunicación, Madrid 1996.

- [5] Jose Ramón Salas-Hackbarth. "*Treesol: Algoritmo para la optimización de la capa física para una red de telecomunicación realizada de forma propietaria*". Jitel'99, Acta de ponencia, Madrid 1999.
- [6] Schmidt et al. "*Un modelo de coste analítico para la red nacional de telecomunicación (en alemán)*". Informe de discusión público del instituto de estudios de servicios de telecomunicación WIK encargado por la Comisión de Regulación Alemana, Bad Honnef 1999.

Modelado Markoviano de Tráfico Agregado ATM

A. Jurado, E. Casilari, A. Reyes, A. Díaz-Estrella y F. Sandoval
Dpto. Tecnología Electrónica, E.T.S.I. Telecomunicación,
Universidad de Málaga, Campus de Teatinos, 29071 Málaga.

Tfno.: 34-952132755; FAX 34-952131447; E-mail: ajurado@mail.fujitsu.es, casilari@dte.uma.es

Abstract

In this paper we describe and compare different markovian strategies when they are designed to approximate aggregate ATM traffic. In particular we focus our study on the two-state MMPP (Markov Modulated Poisson Process) model because of its simplicity and ability to match both cell and burst scales. The queueing behaviour of MMPP model is studied depending on the characterisation policy and the chosen statistical descriptor of ATM traffic. The study is performed using the aggregation of two different classes of individual sources modelled via On-Off processes. Finally the comparison is extended to an actual ATM trace, proving the capability of MMPP models to fit real aggregate ATM traffic.

1. Introducción

Se puede afirmar que la evolución del mundo de los servicios de telecomunicación desde hace ya casi una década ha venido determinada por dos hechos fundamentales:

- En primer lugar, por una mayor globalización de los servicios, que hasta ahora ha podido ser posible especialmente por el desarrollo de una mayor conectividad, pero sobre todo por la fulgurante expansión de un protocolo distribuido como internet. Esta globalización ha motivado no sólo la aparición de nuevos servicios sino también ha suscitado el interés de los usuarios por acceder a servicios remotos que hasta la fecha sólo podía disfrutar a nivel local.

- En segundo lugar, se ha constatado la necesidad de una progresiva integración de los servicios que rompe de plano la separación tradicional de las redes físicas, exclusivamente pensadas para un tipo de servicio (básicamente redes de distribución de TV, redes telefónicas y redes de datos).

Un protocolo tan flexible y universal como ATM constituye una piedra angular en la construcción de una red universal determinada por estos fenómenos de globalización e integración de servicios.

Sin embargo, la flexibilidad inherente a ATM, así como la necesidad de su integración con las arquitecturas de protocolos hoy en día existentes (fundamentalmente las basadas en IP) exige la definición a varios niveles (vigilancia, control de admisión de llamadas,...) de diversos controles y mecanismos de gestión de tráfico, muchos de los cuales, hoy por hoy, se encuentran en estado embrionario o, cuando menos, por optimizar de una manera eficiente. Por su parte, la correcta definición de estos controles así como el dimensionamiento [1] de componentes de la transmisión (buffers,

enlaces,...) exige un amplio conocimiento de los flujos de tráfico que van a ser transmitidos. Este conocimiento implica caracterizar y modelar con precisión dichos flujos. Puesto que la principal ventaja de ATM radica en que habilita la multiplexación sobre el mismo enlace de servicios de muy diversas naturaleza y exigencias de calidad de servicio, puede resultar de especial interés modelar el comportamiento del tráfico resultante de agregar fuentes individuales heterogéneas. De este modo, con un único modelo se aproximaría el flujo global transportado sobre el enlace ATM.

Entre los modelos de tráfico agregado más conocidos se encuentran los modelos markovianos y, en concreto, los procesos markovianos con generación poissoniana (MMPP o *Markov Modulated Poisson Process*). En este trabajo se comparan diversas estrategias de modelado MMPP cuando se aplica al tráfico generado por agregación de fuentes de distinta clase. En concreto el modelado se establece sobre la multiplexación de fuentes individuales caracterizadas mediante procesos On-Off. La comparación se realiza observando en cola (en términos de pérdidas sufridas por el tráfico) la calidad de la aproximación que lleva a cabo cada estrategia. En una última prueba, se incluye la aproximación que efectúan dichos modelos MMPP cuando se aplican sobre tráfico ATM real.

2. Modelado markoviano

Un proceso estocástico X_n se dice markoviano (por defecto, de orden uno) o que forma una cadena de Markov si el valor de la variable aleatoria para el estado $n+1$ depende exclusivamente del valor que posea en el estado n y no de estados anteriores. En el caso de un proceso markoviano con espacio de estados discreto, el número de valores que puede adoptar X_n queda acotado a un conjunto finito y por tanto el proceso queda completamente definido por

dicho conjunto y por una matriz de transición bidimensional que determina la probabilidad de transitar de un estado a otro o de quedarse en el actual.

En el caso de un proceso modulado por Markov, X_n determina en el tiempo (o modula) los parámetros de generación estadística de otro proceso estocástico Y_n . Pensando directamente en el modelado de tráfico, esta señal Y_n ha de poseer cierto significado físico (normalmente: cantidad de información en bits, células o paquetes recibidas durante cierto intervalo o bien tiempo entre llegada de dichas unidades).

En otras palabras, en un proceso modulado por Markov el proceso evoluciona de una manera markoviana por una cadena de estados. Para cada estado la generación de tráfico se efectúa de una manera distinta, habitualmente siguiendo una función de distribución cuyos parámetros dependen del propio estado. Así, para un proceso MMPP el proceso markoviano subyacente determina la tasa media de generación (λ) de un proceso de Poisson.

Entre las principales ventajas de los modelos markovianos podemos citar:

- Se trata de modelos de propósito general, capaces de ser aplicados a distintos tipos de tráfico.
- Si el número de estados es bajo, el modelo se puede definir con pocos parámetros. Esto implica una mayor parsimonia y transportabilidad del modelo.
- La definición de la matriz de conmutación del modelo a partir del tráfico a imitar suele ser simple.
- Son modelos sencillos de implementar en la medida en que las dependencias son simples (a corto plazo: de un estado con el anterior).
- Poseen un tratamiento analítico en colas bien conocido y estudiado.
- Se encuentra implementado en generadores hardware ATM comerciales, como STG de NTR o Adtech AX/400.

Por el contrario, como principales desventajas de este tipo de modelos podemos mencionar:

- El incremento de la complejidad del modelo (o lo que es lo mismo del número de estados) exige un crecimiento geométrico de la parametrización del mismo a causa de la necesidad de definir una matriz de conmutación bidimensional.
- El modelo markoviano es una técnica "clásica" de modelado incapaz de representar eficientemente nuevos aspectos detectados en el tráfico multimedia

[2], especialmente la variabilidad o dependencia a largo plazo (LRD, *Long Range Dependence*, autosemejanza, efecto Joseph o José). No obstante, se ha de decir que la LRD tiene una repercusión relevante sólo para escenarios de transmisión con *buffers* grandes y ocupaciones altas, condiciones poco realistas para la transmisión de muchos tipos de servicios. Para ocupaciones bajas y *buffers* pequeños un modelo markoviano puede aproximar razonablemente bien la realidad [3]. Por otra parte, a costa de perder buena parte de su tratabilidad, si en lugar de utilizar tiempos de permanencia en los estados de corte exponencial (o geométrico, tal y como resulta de la aplicación de una matriz de conmutación) se emplean distribuciones subexponenciales, como la de Pareto, el modelo (denominado ya semimarkoviano) podría generar tráfico de características LRD.

3. Estrategias de modelado MMPP

Como ya se ha apuntado, uno de los modelos modulados por Markov más empleados es el MMPP. Dicho modelo combina la simplicidad de la conmutación markoviana con la de la generación poissoniana. Por la propia adecuación de Poisson para caracterizar el agregado de procesos estadísticamente independientes, de acuerdo con el teorema de Palm [4], los procesos MMPP han sido muy utilizados para representar el tráfico de fuentes multiplexadas.

Añadidamente, el tráfico que se deriva de agregar fuentes MMPP puede ser modelado, con total equivalencia, por otra fuente MMPP [4].

El modelo MMPP más sencillo y utilizado es el que se ilustra en la Fig. 1. Como se puede observar, el modelo consta de dos estados (1 y 2) en los cuales la generación sigue procesos de Poisson de tasas medias λ_1 y λ_2 , respectivamente. Por su parte, la permanencia en cada estado se rige por distribuciones exponenciales de valores r_1^{-1} y r_2^{-1} .

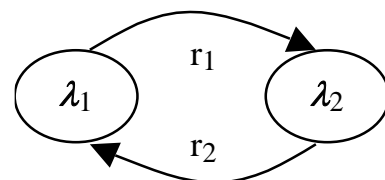


Figura 1. Modelo MMPP de dos estados

Sin embargo, y tal como se ilustra en la Fig. 2, el diseño de un modelo exige, previamente a la elección (en este caso el modelo MMPP) y ajuste de cierto proceso estadístico, lo que se suele denominar caracterización. La caracterización supone analizar la realidad a imitar (el tráfico) con el objeto de describirla mediante un conjunto reducido de variables (descriptor estadístico) que resumen y abstraen sus aspectos más importantes. Tras la definición de este descriptor, que, en ocasiones

puede actuar directamente como un modelo en sí, la tarea de modelado se centra en elegir un proceso matemático que aproxime dicho descriptor, ya con independencia de la realidad física detallada que resume. El objetivo de esta fase, tras decidir qué modelo se emplea, es una parametrización adecuada del mismo que asegure un ajuste óptimo del descriptor.



Figura 2. Proceso de diseño de un modelo

De este modo, la definición del modelo MMPP puede ser múltiple en función de la caracterización seguida. En este sentido los estadísticos comúnmente utilizados para caracterizar un tráfico ATM son los siguientes [5]:

* Media (a largo plazo):

$$\mu = \lim_{t \rightarrow \infty} \frac{E[N(t)]}{t}$$

donde $N(t)$ representa el número de células generadas por el tráfico durante cierto intervalo t .

* Índice de dispersión de cuentas:

$$I(t) = \frac{E\left\{\left[N(t) - E(N(t))\right]^2\right\}}{E[N(t)]}$$

* Índice de asimetría:

$$S(t) = \frac{E\left\{\left[N(t) - E(N(t))\right]^3\right\}}{E[N(t)]}$$

* Índice de dispersión de intervalo (también denominado coeficiente de variación):

$$IDI = \frac{E\left\{\left[X_i - E(X_i)\right]^2\right\}}{E[X_i]^2}$$

donde X_i representa el tiempo de generación de la célula i .

* Autocorrelación del tiempo entre llegadas:

$$R_x(\tau) = \frac{E\left\{\left[X_i - E(X_i)\right] \cdot \left[X_{i+\tau} - E(X_{i+\tau})\right]\right\}}{E\left\{\left[X_i - E(X_i)\right]^2\right\}}$$

A partir de estos estadísticos se suele definir la parametrización del modelo MMPP, tal es el caso de las propuestas de Heffes [6], Gusella[7] y Akimaru [8]. En este último caso se propone el uso de un estadístico difícil de estimar sobre una muestra de tráfico (la derivada $I'(t)$ del índice de dispersión), aunque se ofrece un método que permite calcular el MMPP equivalente a partir de la definición como procesos On-Off de las fuentes individuales. Por el contrario, la caracterización utilizada por Kang [9]

analiza directamente el comportamiento del tráfico a imitar cuando es encolado en cierto servidor de tasa de servicio fija con cola infinita simulada. De esta manera, aparte de la ocupación y de la correlación a corto plazo del tiempo entre llegadas, el modelo recoge el que podríamos llamar comportamiento de ráfaga, estudiando la probabilidad de que la cola del servidor alcance valores altos. Con este objetivo se aproxima la caída de la función de distribución complementaria de la ocupación de la cola $W(x)$ mediante una exponencial, del modo:

$$W(x) = Prob(Q > x) \approx a_1 e^{-s_1 x} \text{ cuando } x \rightarrow \infty$$

donde la variable Q representa la evolución en el tiempo de la ocupación o longitud de la cola, la cual se asocia directamente con el tiempo de espera y, por tanto, con el retardo que sufre el servicio que está siendo encolado.

Dado que el comportamiento de la cola MMPP/D/1 es perfectamente conocido [10] y deducible a partir de los parámetros del modelo, Kang propone el ajuste de tasa de caída a_1 y constante s_1 asintóticas a partir de los parámetros extraídos de la comentada aproximación. El ajuste de estos valores, así como de la correlación a largo plazo, garantizan que la sencilla forma de $W(x)$ de un modelo MMPP (Fig. 3) se adecua en la medida de lo posible al doble comportamiento a nivel de ráfaga y célula que presenta un tráfico ATM.

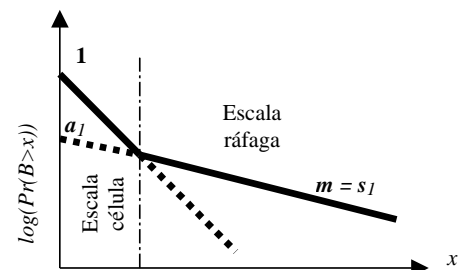


Figura 3. Distribución de la ocupación de la cola. Modelo MMPP/D/1

Finalmente, para concluir con este apartado, en la Tabla 1 se han resumido las caracterizaciones empleadas por cada ajuste, incluyéndose además las utilizadas por dos modelos On-Off de tráfico agregado propuestos por Andrade [11] y Sykas [12] que también se consideran en el apartado de pruebas. Como se puede observar los modelos MMPP, que se definen mediante 4 parámetros, exigen 4 estadísticos que ajustar. Conocida la definición de dichos estadísticos a partir de los parámetros MMPP, el cálculo analítico de estos parámetros se define a partir de un sistema no lineal de cuatro ecuaciones en donde las incógnitas son los propios parámetros.

4. Simulación y resultados

Para cotejar los distintos esquemas de ajuste MMPP se propone comparar su comportamiento en cola cuando son diseñados para ajustar cierto tráfico agregado. En ausencia de una aplicación concreta del modelo de tráfico, la comparación de la

Modelo de Superposición	Estadísticos ajustados
Andrade (On-Off) [5]	$\mu, I_{t \rightarrow 1/\mu}, I_{t \rightarrow \infty}$
Sykas (On-Off) [12]	Tres primeros momentos del tiempo entre llegadas (X_i)
Akimaru [8]	$\mu, I_{t \rightarrow \infty}, S_{t \rightarrow \infty}, I'_{t \rightarrow 0}$
Heffes [6]	$\mu, I_{t \rightarrow 0}, I_{t \rightarrow \infty}, S_{t \rightarrow t1}$
Gusella [7]	$\mu, I_{t \rightarrow t1}, I_{t \rightarrow \infty}, IDI$
Kang [9]	$\rho, R_X(1), \text{Tasa de caída asintótica } (s_1), \text{ Constante asintótica } (a_1)$

Tabla 1. Caracterización propuesta por los diversos autores

aproximación en cola constituye el mejor criterio a la hora de evaluar la calidad del ajuste, ya que en cierta medida examina el impacto que el tráfico tiene sobre la red al ser transmitido, así como la capacidad de la misma red para proporcionar la calidad que el tráfico exige.

En concreto, los modelos se validaron considerando tanto tráfico de agregación sintético, modelado por la superposición de fuentes ATM, como de una muestra real, tomada en un enlace ATM, correspondiente a la agregación del tráfico proveniente de varias redes LAN Fast-Ethernet.

4.1. Pruebas con fuentes On-Off

En esta primera comparación se consideró el tráfico sintético generado por la multiplexación de diversas clases de fuentes individuales, modeladas mediante fuentes On-Off, tal y como se muestra en la Fig. 4.

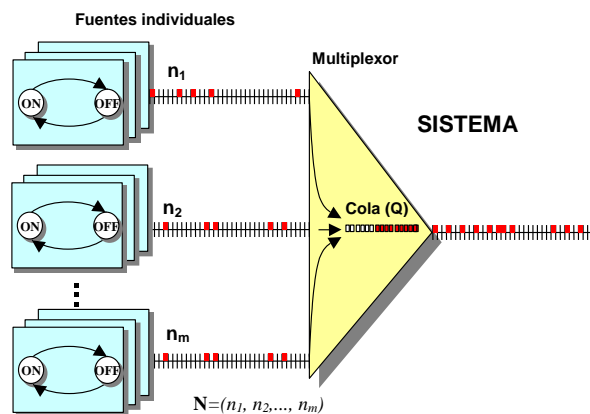


Figura 4. Sistema de simulación y análisis

Las clases podrían representar *grosso modo* los distintos clases de servicios que compartirán los enlaces ATM, de manera que el tráfico global quedaría descrito en cada momento mediante un vector N que indicase el número de fuentes activas de cada clase.

Para las pruebas se consideraron dos clases de fuentes (clase 1 y clase 2), cada una de las cuales fue modelada mediante un modelo On-Off. El modelo On-Off es especialmente adecuado para representar el tráfico intermitente o rafagueante proveniente de muchos tipos de fuentes de tráfico (sobre todo datos y voz). Un modelo On-Off es un proceso de generación determinista modulado también por una

cadena markoviana de dos estados, con la particularidad de que durante la permanencia en uno de ellos (Off) la generación es nula. Así, el modelo queda perfectamente definido mediante tres parámetros: la velocidad de pico (o tasa de generación constante durante el periodo On) y los tiempos medios de permanencia en cada estado.

No es difícil establecer cierto paralelismo y darse cuenta de que los modelos On-Off caracterizan en gran medida el tráfico con los descriptores estadísticos propuestos por la ITU y el ATM Fórum (velocidad de pico, velocidad media y sostenible y tamaño de ráfaga máximo), los cuales ya se emplean no sólo en generadores de tráfico sintético real, como el equipo HW E1609A de Hewlett Packard, sino también en políticas de control de admisión y vigilancia.

Las características de los dos tipos de fuentes aparecen en la Tabla 2. Estas fueron seleccionadas para cubrir cierto rango de servicios. Así, las fuentes de la clase 1 presentan un comportamiento mucho menos intermitente o rafagueante que el de la clase 2, ya que la relación entre las velocidades de pico y media son, respectivamente, 2 y 5. De esta manera, y jugando con el número de fuentes activas de cada tipo (n_1 y n_2) y, por tanto, con el peso relativo de cada tipo de tráfico, se podrá evaluar la capacidad de los modelos MMPP para adaptarse a tráficos más o menos intermitentes.

	Clase 1	Clase 2
V_p	0.013333 células/slot	0.066666 células/slot
$t_{on} (a^{-1})$	884 slots	177 slots
$t_{off} (b^{-1})$	884 slots	708 slots

Tabla 2. Características de las fuentes individuales

En la simulación se asume que la velocidad de transferencia del enlace ATM es de 150 Mbits/s, y que la longitud de la célula es de 53 bytes (424 bits). Por lo tanto, el tiempo de servicio de cada célula (o *slot*) es de 2.83×10^{-6} s.

El análisis en cola se llevó a cabo estimando, mediante simulación, las pérdidas sufridas por el tráfico. Para ello se consideraron distintos tamaños

de la cola del multiplexor (de 0 a 100 células) así como diversos grados de actividad de ambos tipos de fuentes. Los resultados obtenidos son los representados en las Fig. 5, 6, 7 y 8.

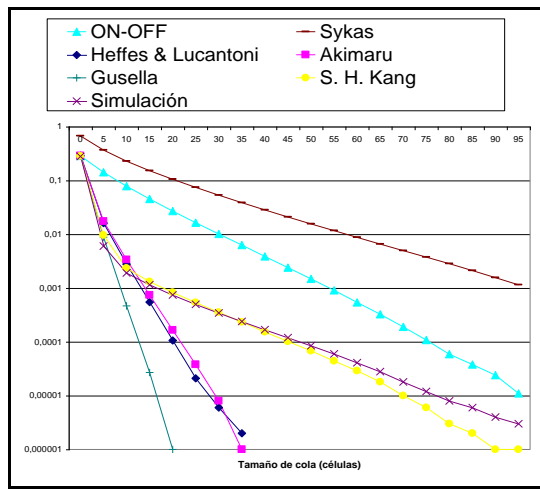


Figura 5. Probabilidad de pérdidas para $n_1=42$, $n_2=32$ (ocupación del 70.66%)

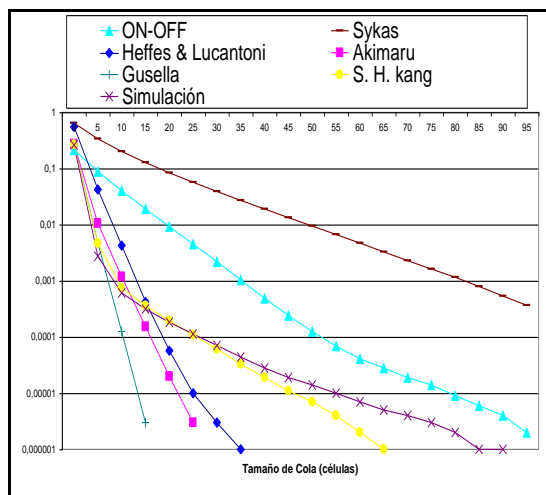


Figura 6. Probabilidad de pérdidas par $n_1=36$, $n_2=31$ (ocupación del 65.33%)

Para todas las gráficas se puede observar claramente la presencia de dos zonas dominadas, respectivamente, por el comportamiento a nivel de célula (colas pequeñas) y a nivel de ráfaga (colas mayores). Se constata además que la inflexión entre ambas zonas ocurre para colas similares al tamaño medio de la ráfaga On de las fuentes individuales (que en ambos tipos coincide en un valor próximo a 12 células).

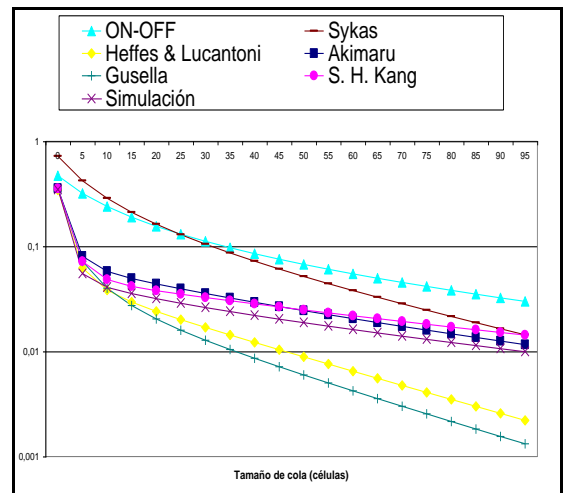


Figura 7. Probabilidad de pérdidas para $n_1=52$, $n_2=44$ (ocupación del 93.33%)

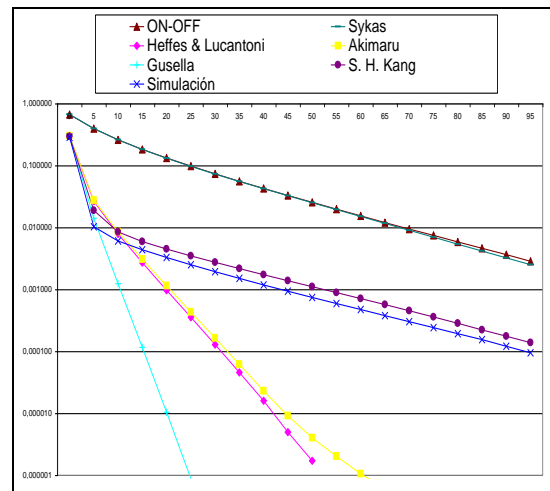


Figura 8. Probabilidad de pérdidas para $n_1=10$, $n_2=50$ (ocupación del 73.33 %)

Las gráficas prueban, para todos los casos, la dificultad de modelar el tráfico agregado mediante procesos pensados para fuentes individuales como los On-Off, puesto que tanto la solución de Sykas como la de Andrade ofrecen aproximaciones de las pérdidas demasiado pesimistas. No obstante, podrían ser utilizados como aproximaciones simples y gruesas del caso peor.

Igualmente se prueba la validez del ajuste de Kang bajo cualquier combinación de fuentes y con independencia del grado de ocupación del enlace.

Los otros modelos MMPP, no obstante, sólo permiten el ajuste del nivel de célula. Para el nivel de ráfaga el desajuste es mayor conforme la ocupación del canal disminuye (menos fuentes de cada tipo), así como en el caso de que la intermitencia sea alta (mayor predominio de las fuentes de clase 2).

4.2. Pruebas con tráfico ATM real

La mayor parte de los trabajos que proponen el uso de MMPP para modelar tráfico agregado utilizan como flujo de pruebas, al igual que el apartado anterior, señales sintéticas generadas por la superposición de fuentes simples, habitualmente caracterizadas como procesos On-Off. Hay que reseñar, sin embargo, que son escasos los trabajos, como el de Ardivisnon [13], en donde se emplean muestras ATM procedentes de enlaces reales. En este apartado se propone comparar el ajuste de los modelos MMPP de Gusella, Kang y Heffes cuando son diseñados para imitar la agregación de tráfico real. A tales efectos se capturó y utilizó tráfico agregado procedente, como ya se ha mencionado, de la multiplexación de tráfico de redes locales Fast Ethernet a 100 Mbps sobre un enlace ATM a 155 Mbps. El sistema de muestreo arrojaba un registro por cada célula transmitida, anotando el tiempo de transmisión (con una precisión de 100 nseg.), y los campos de VPI (*Virtual Path Identifier*) y VCI (*Virtual Channel Identifier*). Dado que la monitorización se produjo sobre el mismo enlace y teniendo en cuenta el hecho de que el tráfico apenas cargaba el enlace, para realizar las pruebas en colas se propone estudiar las pérdidas que sufriría el tráfico agregado si la tasa de servicio fuese 25 veces menor. Los resultados de este análisis son los reflejados en la Fig. 9, donde se consideran colas de hasta 2000 células. Aunque la resolución de la figura quizás no permite observarlo con nitidez, la transición en este caso entre el comportamiento de ráfaga y célula se produce con tamaños de cola de 31-32 células, correspondiente con el tamaño de paquete máximo impuesto por la red Ethernet (1500 bytes).

La Fig. 9 vuelve a probar la capacidad del modelo de Kang para ajustar el tráfico agregado, en tanto que las otras aproximaciones MMPP tan sólo son capaces de ajustar el nivel de célula.

Para acabar con el apartado de pruebas en la Tabla 3 se presenta la comparación de todos los métodos de acuerdo con las criterios de prueba ya comentados.

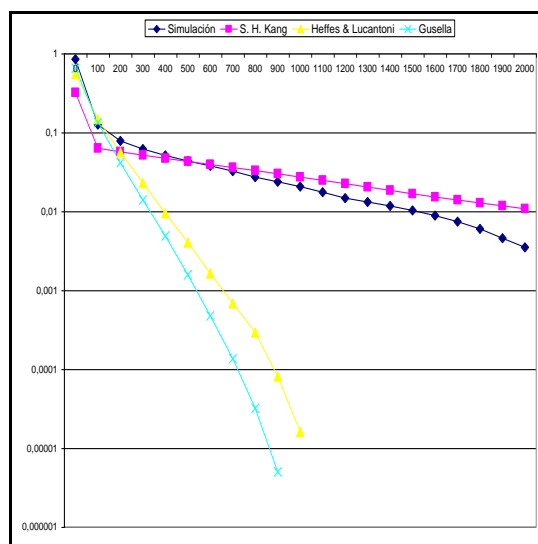


Figura 9. Probabilidad de pérdidas para una tasa de servicio 1/25 de la real y diversos tamaños de cola (muestras ATM reales)

5. Conclusiones

En este trabajo se ha demostrado la posibilidad de modelar tráfico agregado mediante procesos modulados por Markov sencillos. Para ello, tomando como base el modelo MMPP de dos estados, se compara la eficiencia de diversos diseños del modelo en función de la caracterización que eligen a la hora de imitar el tráfico.

La comparación se efectúa considerando tanto tráfico sintético, fruto de la multiplexación On-Off, así como muestras de tráfico ATM reales. Las pruebas, que comparan hasta seis modelos diferentes, demuestran la efectividad de la propuesta de Kang [9] a la hora de capturar el doble comportamiento a nivel de célula y ráfaga, típico de un tráfico ATM. Bajo ciertas circunstancias (bajo rafagueo, alta ocupación) esquemas como el de Heffes o Gusella aproximan mejor la realidad, especialmente para el nivel de célula.

En cualquier caso, hay que recordar que la aproximación de Kang se establece analizando directamente el comportamiento objetivo (en cola) bajo ciertas condiciones de tasa de servicio o, lo que es lo mismo, de ocupación del canal, mientras que

Modelo de Superposición	Comportamiento al nivel de Célula	Comportamiento al nivel de Ráfaga	Comportamiento dependiendo de la Carga	Comportamiento dependiendo del Rafagueo
Sykas	Mal	Mal	Independiente	Independiente
Andrade	Regular	Regular	Independiente	Independiente
Akimaru	Bien	Regular	Mejora el ajuste	Empeora
Heffes	Bien	Regular	Mejora el ajuste	Empeora
Gusella	Bien	Mal	Independiente	Empeora
Kang	Bien	Bien	Independiente	Independiente

Tabla 3. Comparación de las prestaciones de los diversos modelos markovianos

los modelos restantes ajustan un descriptor estadístico propio del tráfico a imitar e independiente del servidor. Esto implica que el modelo de Kang se debe efectuar *ad hoc* para cada ocupación, sin proponer un modelo global válido para cualquier circunstancia.

También se debe comentar que las pruebas se han efectuado considerando tamaños de colas pequeños (de hasta dos mil células, en consonancia con los tamaños de *buffer* de muchos conmutadores comerciales). Si se considerasen tamaños mayores y en el marco de trabajo de tráfico real, podrían comenzar a tener relevancia los efectos de la posible autosemejanza o dependencias a largo plazo, algo que en ningún modo recoge un modelo simple como el MMPP y que un modelo markoviano sólo puede aproximar, con limitaciones, mediante complejas estructuras multinivel [14].

Agradecimientos

Este trabajo ha sido financiado en parte por la Comisión Interministerial de Ciencia y Tecnología (CICYT), Proyecto N° TIC96-0743. Igualmente, queremos agradecer a Telefónica I+D la cesión de las muestras de tráfico ATM.

Referencias

- [1] Baiocchi, A., Bléfari, N., y Roveri, A., "Buffer Dimensioning Criteria for an ATM Multiplexer Loaded with Homogeneous On-Off Sources", en *Queueing, Performance and Control in ATM*, Eds. W. Cohen y C.D. Pack, Elsevier Science Publisher B. V., 1991.
- [2] Leland, W.E., Taqqu, M.S., Willinger, W., y Wilson, D.V., "On The Self-Similar Nature Of Ethernet Traffic" (Extended Version). *IEEE/ACM Transactions On Networking Communications*, Vol. 2, N° 1, Febrero, 1994, pp. 1-15.
- [3] Ryu, B.K., y Elwalid, A., "The Importance of the Long-Range Dependence of VBR Video Traffic in ATM Traffic Engineering: Myths and Realities," *Proceedings of the ACM SIGCOMM'96*, Stanford, Agosto, 1996, pp. 3-14.
- [4] Frost, V. S. y Melamed, B., "Traffic Modeling for Telecommunications Networks", *IEEE Communications Magazine*, Marzo, 1994, pp. 70-81.
- [5] Andrade, J., "Statistical parameters to describe cell traffic generated by broadband services", *Comunicaciones de Telefónica I+D*, Vol. 4, N° 2, Julio-Diciembre, 1993, pp. 81-88.
- [6] Heffes, H., y Lucantoni, D.M., "A Markov Modulated Characterization of Packetized Voice and Data Traffic and Related Statistical Multiplexer Performance". *IEEE Journal of Selected Areas in Communications*, Vol. 4, N° 6, 1986, pp. 856-868.
- [7] Gusella, R., "Characterizing the Variability of Arrival Processes with Indices of Dispersion", Informe Interno TR-90-51, International Computer Science Institute, Berkeley, USA, Septiembre, 1990.
- [8] Akimaru, H., Okuda, T., y Nagai, K., "A Simplified Performance Evaluation for Bursty Multiclass Traffic in ATM Systems", *IEEE Transactions on Communications*, Vol. 42, N° 5, Mayo, 1994, pp. 2078-2083.
- [9] Kang, S.H., Changhwan, O.H., y Sung, D.K., "A Traffic Measurement-Based Modelling of Superposed ATM Cell Streams", *IEICE Transactions on Communications*, Vol. E80-B, N° 3, Marzo, 1997, pp. 434-441.
- [10] Baiocchi, A., y Bléfari-Melazzi, N., "Steady-State Analysis of the MMPP/G/1/K Queue", *IEEE Transactions on Communications*, Vol. 41, N° 4, Abril, 1993, pp. 531-4.
- [11] Andrade, J., Burakowski, W., Villén-Altamirano, M., "Characterization of Cell Traffic Generated by an ATM Source", en *Teletraffic and Datatraffic in a Period of Change*, Eds. A. Jensen y V.B. Eversen, pp. 545-550, 1991.
- [12] Sykas, E.D., Vlakos, K.M., y Nikolaos, G. A., "Performance Evaluation of Statistical Multiplexing Schemes in A.T.M. Networks", *Computer Communications*, Vol. 14, N° 5, Junio, 1991, pp. 273-286.
- [13] Arvidsson, Ä., y Lind, C., "Using Markovian Models to Replicate Real ATM Traffics", en *ATM Networks: Performance, Modelling and Evaluation*, Vol. 2, Ed. D. Kouvatsos, Chapman & Hall, 1996, pp. 39-54.
- [14] Karlsson, P., y Arvidsson, Ä., "Modelling of traffic with high variability over long time scales with MMPPs", Department of Telecommunications and Mathematics, Universidad de Karlskrona, Suecia, 1995, Accesible via HTTP en: <http://www.itm.hk-r.se/~akear/bursty.html>

Modelado Wavelet de tráfico de vídeo VBR

E. Casilari, A. Reyes, A. Díaz-Estrella y F. Sandoval
Dpto. Tecnología Electrónica, E.T.S.I. Telecomunicación,
Universidad de Málaga, Campus de Teatinos, 29071 Málaga.
Tfno.: 34-952132755; FAX 34-952131447; E-mail: casilari@dte.uma.es

Abstract

In this paper we propose the utilisation of wavelets to approximate the self-similar nature of VBR video traffic. The characterisation in the Wavelet domain, instead of the time domain, allows to preserve the parsimony of fractal models avoiding their generation complexity and the need of estimating the Hurst parameter. The proposed model includes an improved approximation of the Wavelet coefficients which utilises a modified version of the exponential distribution. The model performance is compared in a queue with those of several actual MPEG traces.

1. Introducción

Hasta hace pocos años el limitado ancho de banda que ofrecían las redes existentes imponía fuertes restricciones a la transmisión de vídeo digital. Sin embargo, el crecimiento de la interconectividad de LANs, la expansión de tecnologías de fibra óptica y la posibilidad de reutilizar los soportes clásicos con mayores anchos de banda (mediante técnicas como VDSL, ADSL,...), así como el auge y aparición de nuevos servicios multimedia y la consolidación de estándares de compresión como MPEG, hacen pronosticar que los flujos que contienen imágenes en movimiento (vídeo) serán una fuente de tráfico telemático de primer orden en las futuras redes digitales. La importancia de este tipo de servicios para el diseño y gestión de las redes radica en dos motivos: de un lado, los elevados recursos que suelen requerir (entendiendo por recurso habitualmente el ancho de banda exigido) y, en segundo lugar, los altos niveles de calidad de servicio (básicamente pérdidas y retrasos) que solicitan a la red.

Si bien hasta la fecha gran parte de las comunicaciones de vídeo aprovechaban esquemas de compresión que originaban flujos de tasa binaria constante o CBR (*Constant Bit Rate*), los escenarios de transmisión asíncronos, como el propuesto por ATM, aconsejan adoptar soluciones de tasa binaria variable o VBR (*Variable Bit Rate*). Los flujos generados por esta clase de compresión no sólo proporcionan una calidad constante, frente a la degradación variable que padecen los servicios CBR, sino que además se pueden beneficiar de la multiplexación estadística que proveen las redes asíncronas, optimizando de esa manera el ancho de banda compartido. Como desventaja, sin embargo se tiene que la gestión del flujo VBR (incluyendo tareas de control de admisión, vigilancia, conformado, controles reactivos, renegociación de recursos, ...) se complica mucho más allá de los simples controles que bastan para manejar un flujo CBR. Por esta razón se justifica la necesidad de conocer y modelar con detalle la naturaleza del tráfico que generan los servicios de vídeo. Así, desde finales de la década de los 80 ha sido abundantísima la bibliografía dedicada al modelado y caracterización del tráfico de vídeo VBR. En esta línea, y a partir principalmente de

un trabajo de Beran [1], se comenzó a estudiar la presencia de un fenómeno estadístico en el tráfico de vídeo: las dependencias a largo plazo o LRD (*Long Range Dependence*). Las LRD han venido a revolucionar, al menos parcialmente, el ámbito del modelado de tráfico telemático, ya que también se han detectado en otros tipos de tráfico [2]: servicios Web, tráfico de enlaces LAN y MAN,..., aunque en el caso del vídeo la LRD se justifica por algo tan intrínseco a este tipo de servicios como es la evolución por distintos tipos de escenas [3] [4]. Las dependencias a largo plazo, que se reflejan en coeficientes de autocorrelación con un decaimiento lento o subexponencial, se asocian a la existencia de una variabilidad del tráfico en una amplia gama de escalas temporales que los modelos tradicionalmente empleados (habitualmente basados en Poisson y en distribuciones exponenciales) no son capaces de recoger y que pueden tener consecuencias graves para muchos esquemas de control, donde se supone una escala "natural" y acotada en donde se manifiesta el carácter rafagueante del tráfico. Por su parte, esta variabilidad de la señal se liga a cierto grado de autosemjanza o fractalidad que ha permitido aplicar en este campo las conclusiones y la analítica desarrollada por Mandelbrot [12] y otros autores que ya habían estudiado procesos fractales en otras ramas de la ciencia.

Hasta la fecha existen dos técnicas genéricas en las que se podrían agrupar los modelos planteados para aproximar la naturaleza LRD de una señal de tráfico:

* Modelos multinivel [3], en donde se imita directamente la variabilidad existente a una serie de niveles o escalas de tiempo. Cada nivel suele modelarse mediante un proceso "clásico" (habitualmente cadenas de Markov) que modula la generación a niveles inferiores. Su variabilidad en ese caso se limita a las escalas contempladas (modelos "quasi-fractales") presentando el inconveniente de que ofrecen estructuras complicadas, de escasa tratabilidad, cuya definición requiere un número amplio de parámetros lo que perjudica la llamada parsimonia (simplicidad paramétrica) del modelo. En el caso del tráfico de vídeo se deben citar los llamados modelos escénicos, donde se define un nivel superior (la escena) que regula la media del tráfico generado a largo plazo, imitando

directamente la existencia de escenas o periodos de diversa actividad dentro del tráfico de vídeo.

* Modelos autosemejantes: estos modelos ya responden a estructuras estrictamente fractales, es decir, de variabilidad a todas las escalas de tiempo. Se corresponden con modelos generalistas bien conocidos, como los ruidos gaussianos fraccionarios o FGN (*Fractional Gaussian Noise*) o los procesos ARIMA fraccionarios o FARIMA. Mientras que los primeros generan señales fractales puras (con la misma forma de distribución estadística con independencia de la escala temporal desde la que son contemplados), los segundos son capaces de añadir el ajuste de las dependencias a corto plazo o SRD (*Short Range Dependence*), incorporando una típica estructura autorregresiva de media móvil o ARMA (*Autorregresive Moving Average*). La principal ventaja de estos modelos, frente a la aproximación desarrollada por los del grupo anterior, es que son capaces de representar la LRD mediante un único valor, el parámetro H o de Hurst, que denota el grado de autosemejanza de la muestra a modelar. De esta manera estos modelos presentan un alto grado de parsimonia. Por el contrario, su principal desventaja radica precisamente en la dificultad de una estimación ajustada e insesgada de H , el cual es un valor acotado entre 0.5 y 1. Esta dificultad se explica por la propia naturaleza de los estimadores propuestos, los cuales se basan en la medición de comportamientos asintóticos (R/S, periodograma, método de la varianza) o en la minimización de funciones que suponen previamente una forma al espectro (estimador de Whittle) y que introducen siempre cierto sesgo en la medida, ya de por sí limitada a un estrecho intervalo. Esta problemática se agrava cuando estos diversos estimadores se aplican sobre series de tráfico real finitas y con posibles problemas de no estacionariedad. A esta dificultad hay que añadir la complejidad computacional de la propia estimación (sobre todo en el caso de Whittle) así como la complejidad de generación que la ejecución rigurosa de estos métodos requeriría (FGN y FARIMA no son sino filtros ARMA de infinitos coeficientes) aunque se debe apuntar que existen soluciones aproximadas (que, por ejemplo, acotan el número de coeficientes a considerar) que pueden solventar parcialmente este problema a efectos prácticos.

De esta manera, la aproximación a la naturaleza LRD exige decidir en un compromiso a favor de la parsimonia o de la complejidad de medida y generación.

Sin embargo, una característica común a ambos tipos de procesos es que indefectiblemente establecen el modelado y la generación en el dominio del tiempo, siendo muy escasos los trabajos (como el presentado en [5]) que proponen el análisis en el dominio de la frecuencia. Así, en este artículo se propone la utilización del dominio Wavelet como una solución alternativa a los métodos comentados para caracterizar las LRD. Dicha solución, que propone modelar los coeficientes resultantes de la transformada Wavelet, mantiene la parsimonia típica de los procesos fractales añadiendo una estimación mucho más unívoca. En concreto, se propone una mejora al algoritmo propuesto

en [6]. La mejora, que no añade complejidad al modelo, redundará en un mejor ajuste de la correlación así como una aproximación mayor a la realidad cuando el modelo es evaluado en una cola.

El siguiente trabajo se estructura en las siguientes secciones: en la sección 2 se hace un breve repaso de las Wavelets, comentando sus principales ventajas y definiendo la base que se empleará a la hora de modelar el tráfico de vídeo. En la sección 3 se ilustra, con el empleo de series reales, la dificultad de estimar el parámetro H y la capacidad de los procesos incorrelados para caracterizar las series en el dominio Wavelet. En la sección 4 se comparan los diversos modelos cuando son analizados en cola. Para finalizar, en la sección 5 se recogen las principales conclusiones del trabajo.

2. Modelado Wavelet

El modelado Wavelet permite representar un proceso discreto en el tiempo $x(t)$ en función de una base ortonormal y un conjunto de coeficientes (coeficientes Wavelet) del modo:

$$x(t) = \sum_{j=1}^K \sum_{m=0}^{2^{K-j}-1} d_j^m \cdot \phi_j^m(t) + \mu_x \quad (1)$$

donde $t \in [0, 2^K - 1]$ y μ_x es el valor medio de $x(t)$.

Los valores d_j^m constituyen los coeficientes Wavelets calculados a partir de la relación:

$$d_j^m = \sum_{t=0}^{2^K-1} x(t) \cdot \phi_j^m(t) \quad (2)$$

donde ϕ_j^m es la mencionada base ortonormal definida a partir de:

$$\phi_j^m(t) = 2^{-j/2} \cdot \phi(2^{-j} \cdot t - m) \quad (3)$$

donde $j \in [1, K]$ y m son dos índices naturales que representan, respectivamente, el retraso o dilación y la traslación en el tiempo.

Como Wavelet madre $\phi(t)$ se utiliza la Wavelet madre Haar, definida mediante la siguiente función por ramas:

$$\phi(t) = \begin{cases} 1, & \text{si } 0 \leq t < 1/2 \\ -1, & \text{si } 1/2 \leq t < 1 \\ 0, & \text{en otro caso} \end{cases} \quad (4)$$

Fijada esta base, la transformada permite caracterizar unívocamente la señal $x(t)$ a partir de los coeficientes d_j^m , los cuales, a partir de la ecuación (2), se pueden computar de una manera simple mediante la relación:

$$d_j^m = 2^{-j/2} \cdot \left(\sum_{t=m \cdot 2^j}^{(m+0.5) \cdot 2^j - 1} x(t) - \sum_{t=(m+1) \cdot 2^j}^{(m+1.5) \cdot 2^j - 1} x(t) \right) \quad (5)$$

La ventaja principal de esta caracterización en el dominio Wavelet es que cada serie de coeficientes, para cada j , representa el comportamiento de la señal a distintas escalas, dependiendo del propio índice j . Así, dada la estructura "autosemejante" de la propia transformada Wavelet, en forma de un árbol binario que se va abriendo infinitamente, tal y como se ilustra en la figura 1, se puede intuir que los coeficientes Wavelet son capaces de representar de una manera

compacta la autosemejanza que en el dominio del tiempo presenta la señal. De esta manera, el modelado Wavelet se efectúa directamente sobre estos coeficientes con el objeto de que los modelos resultantes sean capaces de generar coeficientes sintéticos que imiten los obtenidos de la realidad. Sobre estas series sintéticas se procederá a efectuar la transformada Wavelet inversa antes formulada, de modo que se conseguirán series sintéticas ya en el dominio del tiempo.

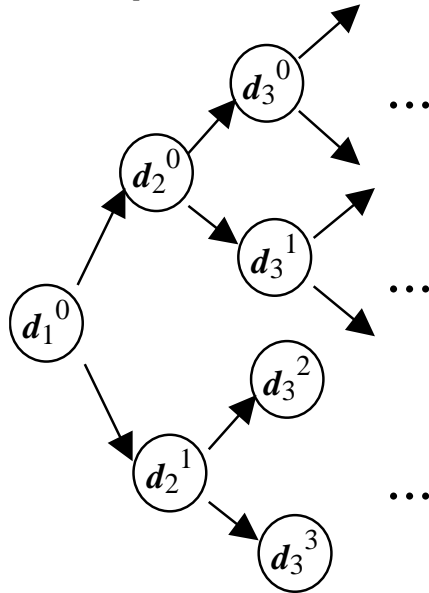


Figura 1. Estructura de los coeficientes Wavelet

La principal ventaja de establecer el modelo en este dominio es que se puede probar que mediante un simple modelo incorrelado de los coeficientes d_j^m para cada nivel de dilación j , la naturaleza LRD queda convenientemente caracterizada. De esta manera, con un leve aumento de la complejidad paramétrica (incomparablemente pequeña frente a cualquier modelo de los que se ha denominado "quasi-fractales"), el modelo aproxima las LRD mediante estimadores simples y sin sesgo (los utilizados en ajustar estadísticos de primer orden).

3. Modelado de secuencias de vídeo VBR MPEG reales

En este apartado se propone aplicar el modelo presentado anteriormente al tráfico generado por secuencias de vídeo con compresión VBR. En concreto, las series consideradas fueron tres largas secuencias, con duraciones superiores a media hora, codificadas bajo el conocido estándar MPEG-1 en lazo abierto (VBR). Dos de ellas, "Star Wars" y "Jurassic Park", se corresponden respectivamente con media hora y la secuencia total de las películas homónimas, "La Guerra de las Galaxias" y "Parque Jurásico". La tercera serie, denominada Wurzburg, está formada por la secuenciación de 17 muestras de media hora, codificadas todas mediante el mismo esquema e incluyendo una gran variedad de contenidos (dibujos animados, retransmisiones deportivas, largometrajes y debates). Las características básicas de las secuencias, a nivel de la jerarquía MPEG de GOP (*Group of Pictures*), son las tabuladas en la tabla 1.

Serie	BW medio	μ (bits)	σ (bits)
"Star Wars" (SW)	334 Kbps	$1.67 \cdot 10^5$	$5.99 \cdot 10^4$
"Jurassic Park" (JP)	1.12 Mbps	$0.26 \cdot 10^6$	$0.14 \cdot 10^6$
"Wurzburg" (W)	473 Kbps	$2.27 \cdot 10^5$	$1.27 \cdot 10^5$

Tabla 1. Características de las fuentes MPEG.
(μ : tamaño medio del GOP, σ : desviación típica)

Se ha de decir que tanto la secuencia "Star Wars" como las incluidas en la muestra "Wurzburg", las cuales presentan el típico formato de GOP de IBBPBBPBBPBB, son empleadas con regularidad por la literatura de tráfico de vídeo.

Para probar la dificultad de estimar con precisión el parámetro H en series de tráfico reales la tabla 2 incluye los resultados que arrojan los distintos métodos cuando son aplicados a las mismas. En el caso de los estimadores paramétricos (Whittle local y agregado) los valores se acompañan con sus respectivos intervalos de confianza para un 95%. Para un estudio detallado de los métodos véase, por ejemplo [7]

Método	H (SW)	H (JP)	H (W)
R/S	0.8570	0.8588	0.9019
Varianza	0.8223	0.8779	0.9050
Periodograma	0.8657	0.8363	0.9187
Whittle local	0.8503 ± 0.0152	0.89018 ± 0.0080	0.9250 ± 0.0064
Whittle agregado	0.8949 ± 0.0453	0.9086 ± 0.0434	0.9570 ± 0.0307

Tabla 2. Estimación del parámetro H con distintos métodos en las muestras empleadas

La tabla muestra la enorme heterogeneidad de los resultados, con una incertidumbre aproximada de más del 10% (0.05) con respecto al margen total de variación del parámetro (0.5).

Esta falta de coherencia de los resultados no sólo es aplicable a la comparación entre los distintos estimadores sino que se extiende a cada uno de los estimadores en función de cómo son programados, es decir, del número de puntos que se consideran para la minimización (Whittle) o para el cálculo de la asíntota (métodos no paramétricos). Por ejemplo, la figura 2 muestra la evolución de las estimaciones del método del periodograma en función del porcentaje de frecuencias bajas consideradas.

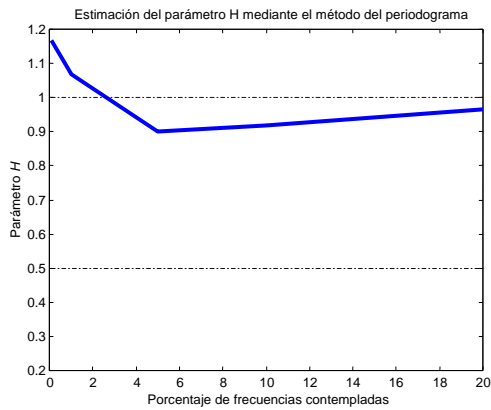


Figura 2. Estimación del parámetro H con el método del periodograma

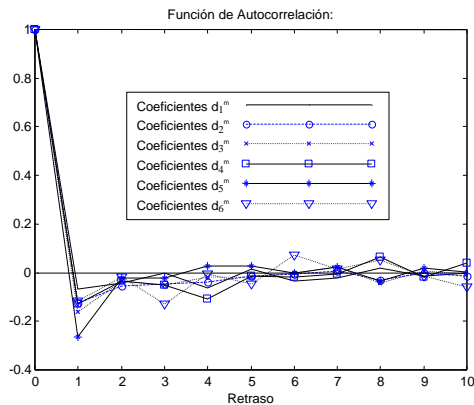


Figura 3. Coeficientes de autocorrelación de las series de coeficientes Wavelet. "Jurassic Park"

Para solventar este problema se propone emplear la transformada Wavelet. A modo de ejemplo, la figura 3 muestra la estructura de correlación de los coeficientes Wavelet para diversos niveles de dilación.

En todos los casos se prueba la práctica incorrelación de las series, lo cual implica que pueden ser modelados mediante ruidos blancos en donde el único esfuerzo de caracterización se centra en la elección de una función de densidad estadística adecuada. Así, en [6] se propone la convencional distribución gaussiana o normal. Dado que los coeficientes presentan media nula, la función queda definida del modo:

$$f(x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{x^2}{2\sigma^2}\right) \quad (6)$$

donde el único parámetro a estimar es la desviación típica de la serie.

Frente a esta aproximación y a la vista de la distribución de los coeficientes Wavelets de las señales reales consideradas, proponemos el empleo de una versión simétrica de la distribución exponencial, de la manera:

$$f(x) = \begin{cases} \frac{1}{2\mu} \exp\left(-\frac{x}{\mu}\right), & \text{si } 0 \leq x \\ \frac{1}{2\mu} \exp\left(\frac{x}{\mu}\right), & \text{si } 0 \geq x \end{cases} \quad (7)$$

La nueva aproximación mantiene la parsimonia de la función gaussiana en tanto se define también mediante un solo parámetro μ , el cual se puede medir sobre la serie a imitar como la media de los valores absolutos de la serie de coeficientes.

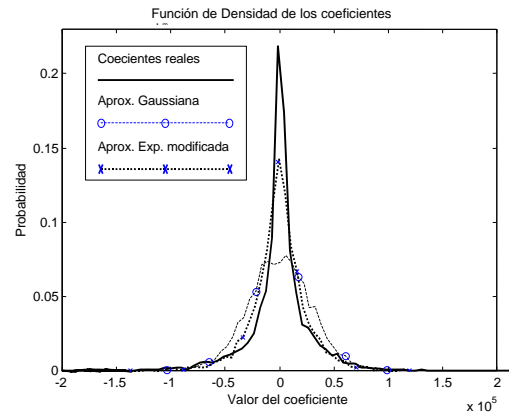


Figura 4. Aproximaciones a los coeficientes Wavelet (d_1^m). Señal "Jurassic Park"

La mejora introducida por la distribución propuesta se ilustra en la figura 4, en donde se representan los ajustes que realizan ambas funciones para la serie de coeficientes de dilación 1 calculados en la señal "Jurassic Park". Con las otras series y para otros niveles de dilación las mejoras son similares a las representadas en dicha figuras.

Una vez sintetizados los coeficientes, basta con aplicar la transformada inversa Wavelet, formulada en la ecuación (1), para obtener series a partir del modelo. Mediante este método y utilizando por separado las distribuciones comentadas se obtuvieron series de más de 100000 puntos para todas las muestras de vídeo real consideradas. El ajuste que estas series efectúan de la función de densidad estadística y los coeficientes de autocorrelación es el representado, respectivamente, en las figuras 5 y 6.

Como se podía esperar, el agregado de señales estadísticamente independientes y con la misma densidad (los coeficientes) que al fin y al cabo efectúa la transformada inversa origina que la señal sintética posea en ambos casos una distribución gaussiana, por la aplicación del teorema central del límite.

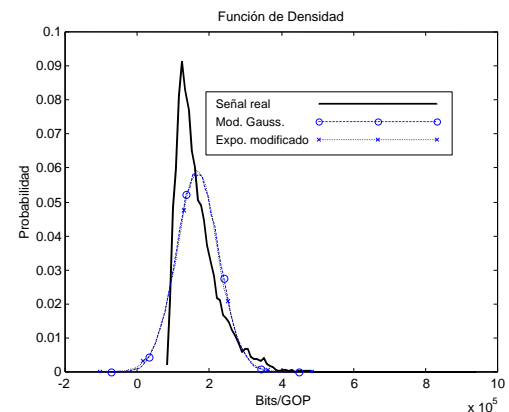


Figura 5. Ajuste de la densidad estadística con series generadas por modelos Wavelet. Señal "Star Wars"

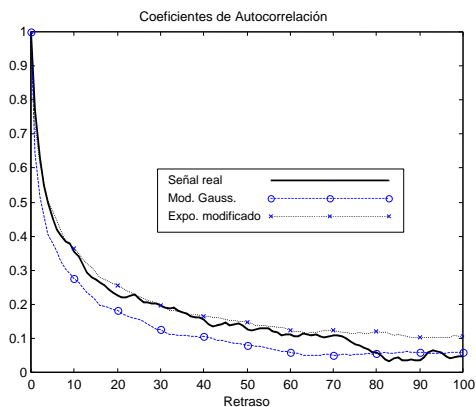


Figura 6. Ajuste de la autocorrelación con series generadas por modelos basados en Wavelet. Señal "Star Wars"

Sin embargo, la figura 6 prueba que una mejora en el ajuste de la densidad de los coeficientes redonda en una mejor aproximación de la autocorrelación de la señal a imitar.

En cualquier caso, la caída de corte hiperbólico que presentan las señales generadas mediante los modelos basados en Wavelet demuestra que mediante la simple generación de señales incorreladas (los coeficientes) se pueden sintetizar procesos de naturaleza LRD o autosemejante.

4. Simulación y resultados en cola

Más allá de la simple comparación de estadísticos de primer o segundo orden, como criterio de evaluación del modelo presentado se opta por evaluar el comportamiento en una cola simulada, estimando el tamaño medio de la misma así como las pérdidas que padece bajo cierto grado de ocupación cuando se utiliza, alternativamente, como entrada a la misma, muestras de tráfico real y el generado por distintos modelos diseñados para imitarlas.

El escenario elegido para estas pruebas fue aquel en donde los efectos de la existencia de LRD son más relevantes, es decir, el de altos niveles de ocupación y tamaños grandes de buffers [3] [8] [9], ya que se ha aducido [10] [11] que en transmisiones sensibles a retardos donde el *buffering* o encolado de la información no se permite, la correlación a largo plazo (o lo que es lo mismo, las componentes bajofrecuenciales del espectro de potencia) no poseen ninguna influencia, al menos sobre la calidad percibida por el propio servicio con naturaleza LRD. Las simulaciones se realizaron considerando un modelo de flujo de fluidos en donde la tasa de servicio de la cola es fija y se elige en función del grado de ocupación del servidor bajo el cual se desea evaluar el comportamiento del modelo. Los parámetros estimados son el tamaño de cola medio para un rango de ocupaciones altas y la probabilidad de pérdida de bit medida para cierta ocupación concreta. Para el primer parámetro se consideró una cola infinita en tanto que

para el segundo se realizaron pruebas con diversos tamaños de colas.

Así, en las figuras 7 y 8 se ofrecen los resultados de las mencionadas mediciones cuando se toma como muestra de análisis la señal "Star Wars".

Junto con las aproximaciones de los dos modelos Wavelet comentados, la figura incorpora también los resultados de dos modelos FARIMA sin sección de media móvil y con sección autorregresiva de orden 1 ($p=1, q=0$). La diferencia entre ambos modelos, ajustados mediante un procedimiento similar al convencionalmente empleado para diseñar filtros ARIMA, reside en que son programados para ajustar dos valores del parámetro H . Estos valores son elegidos para cubrir, aproximadamente, los límites del intervalo de valores de H calculados por los distintos estimadores. Así, para la serie "Star Wars" se observaron valores de H de 0.85 y 0.90.

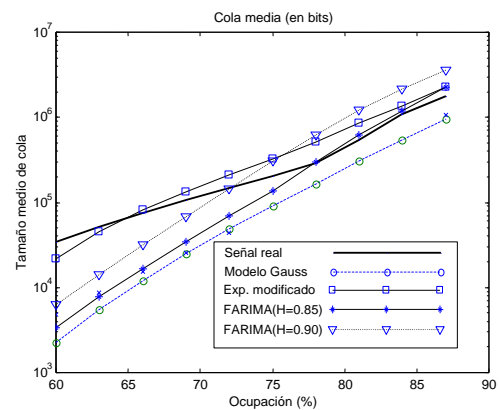


Figura 7. Tamaño medio de cola para los diversos modelos. Señal "Star Wars"

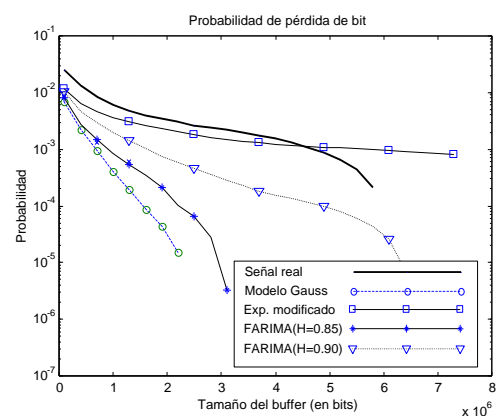


Figura 8. Probabilidad de pérdidas para una ocupación del 70%. Señal "Star Wars"

Las figuras y en especial la relativa a pérdidas prueban la capacidad del modelo Wavelet propuesto para aproximar el comportamiento de la realidad, mejorando incluso los resultados del modelo FARIMA más pesimista, es decir, aquel que supone una autosemejanza o LRD más intensa en la señal ($H=0.90$). En cualquier caso se ha de reseñar la importancia que puede llegar a tener un desajuste de H ,

como indica la divergencia entre los modelos FARIMA.

Estas mismas conclusiones se obtienen si, en lugar de "Star Wars" se analizan las otras series "Jurassic Park" y "Wurzburg" (Figuras 9,10,11 y 12). En todos los ejemplos el modelo propuesto proporciona estimaciones más pesimistas y más ajustadas a la realidad que el modelo Wavelet con aproximación gaussiana de coeficientes y que los modelos FARIMA.

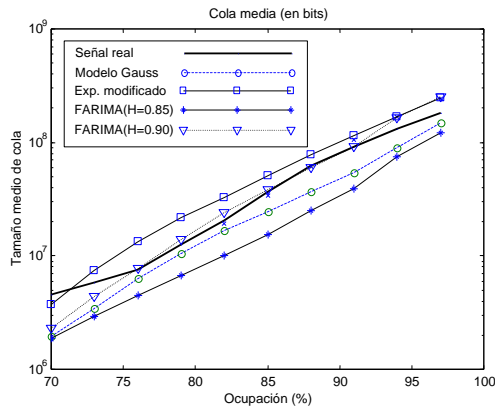


Figura 9. Tamaño medio de cola para los diversos modelos. Señal "Jurassic Park"

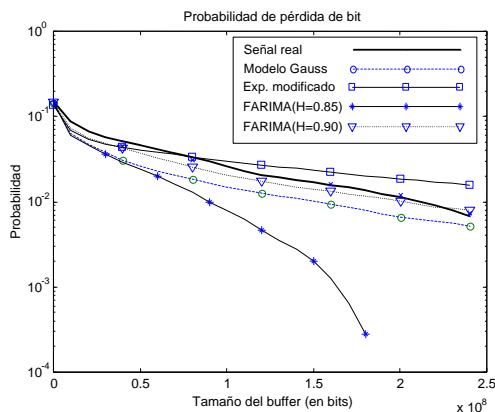


Figura 10. Pérdidas para una ocupación del 90%. Señal "Jurassic Park"

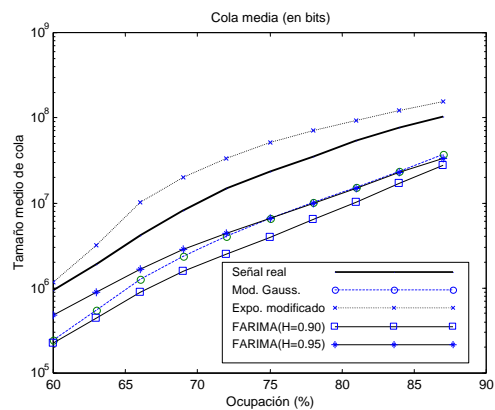


Figura 11. Tamaño medio de cola para los diversos modelos. Señal "Wurzburg"

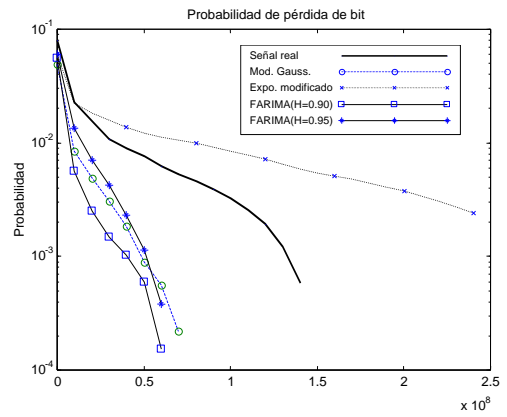


Figura 12. Pérdidas para una ocupación del 70%. Señal "Wurzburg"

Tras establecer estas pruebas con fuentes individuales, se pasa a evaluar la capacidad del modelo propuesto cuando se aplica para caracterizar tráfico agregado. Como muestra de dicho tráfico y a falta de un conjunto amplio de muestras homogéneas se consideró la suma de veinte réplicas de la señal "Jurassic Park". Para reducir el efecto a corto plazo de la correlación cruzada entre las señales se tomaron puntos aleatorios a la hora de considerar el inicio de la muestra.

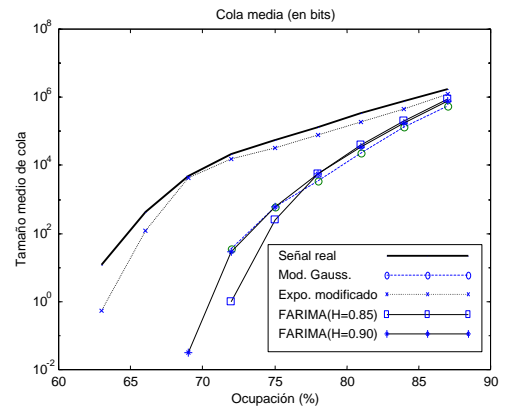


Figura 13. Cola media para el agregado de 20 muestras. Señal "Jurassic Park"

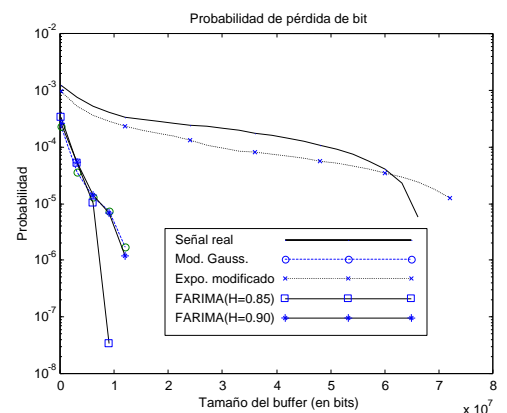


Figura 14. Pérdidas para el agregado de 20 muestras. Ocupación 80%. Señal "Jurassic Park"

Las figuras 13 y 14 corroboran los resultados obtenidos con las fuentes simples: observándose de nuevo el

mejor ajuste de pérdidas y retraso obtenidos con el modelo propuesto.

5. Conclusiones

En este trabajo se ha propuesto un modelo de tráfico de vídeo VBR en el dominio Wavelet. El principal interés de modelar en este dominio radica en la posibilidad de caracterizar procesos LRD mediante una caracterización de estadísticos de primer orden, típica de procesos incorrelados. Para ello se propone imitar la distribución de los coeficientes Wavelet de cada escala de tiempo mediante una modificación de la distribución exponencial. Mediante pruebas en colas, se prueba la capacidad que posee el modelo de ajustar el comportamiento de series de vídeo MPEG reales, ya que mejora la de los modelos basados en Wavelet que caracterizan los coeficientes mediante campanas gaussianas. Frente a modelos multinivel el modelado Wavelet presenta una gran parsimonia, evitando al tiempo la complejidad de estimación paramétrica de los procesos fractales, la otra familia habitualmente propuesta para imitar en el dominio del tiempo las características LRD.

En este trabajo se ha probado la inexistencia de correlación entre los coeficientes Wavelet pertenecientes al mismo nivel de dilación, sin cuestionarse la existencia de correlación entre coeficientes de niveles distintos. En [6] se modelan dichas correlaciones mediante filtros autorregresivos que incrementan la complejidad al modelo sin añadir ventajas substanciales, sobre todo en lo que se refiere a comportamiento en colas.

Por otro lado, se ha apuntado la naturaleza gaussiana del tráfico generado por los modelos presentados. Esto implica su incapacidad para ajustar en detalle la forma de la función de distribución, la cual es crítica a la hora de evaluar escenarios de bajos niveles de ocupación y *buffers* pequeños. En esas condiciones se debe incorporar al modelo (tanto a los FARIMA como a los aquí planteados) mecanismos que mejoren el ajuste de dicha distribución, como el ya clásico de proyectar la señal gaussiana sobre la composición de su propia distribución y la función de distribución inversa a imitar (método de la inversión).

Agradecimientos

Este trabajo ha sido financiado en parte por la Comisión Interministerial de Ciencia y Tecnología (CICYT), Proyecto N° TIC96-0743. Igualmente, queremos agradecer a Jorge Mata y Luis J. de la Cruz (Dpto. Telemática Aplicada, UPC) la cesión de la muestra "Jurassic Park" de tráfico de vídeo.

Referencias

[1.] Beran, J., Sherman, R., Taqqu, M.S., y Willinger, W., "Long-Range Dependence in Variable-Bit-Rate Video Traffic", *IEEE Transactions on Communications*, Vol. 43, N° 2/3/4, Abril, 1995, pp. 1566-1579.

[2.] Leland, W.E., Taqqu, M.S., Willinger, W., y Wilson, D.V., "On The Self-Similar Nature Of Ethernet Traffic" (Extended Version). *IEEE/ACM Transactions On Networking Communications*, Vol. 2, N° 1, Febrero, 1994, pp. 1-15.

[3.] Jelenkovic, P.R., Lazar, A.A., y Semret, N., "The effect of Multiple Time Scales and Subexponentiality in MPEG Video Streams on Queueing Behavior", *IEEE Journal on Selected Areas in Communications*, Vol. 15, N° 6, Agosto, 1997, pp. 1052-1071.

[4.] Casilari, E., Reyes, A., Díaz-Estrella, A., y Sandoval, F., "Heavy-Tailed Distribution of the Scene Duration in VBR Video", *Electronic Letters*, Vol. 35, N° 2, Enero, 1999, pp. 134-135.

[5.] San-Qi L., Chong, S., y Hwang, Ch.L., "Link Capacity Allocation and Network Control by Filtered Input Rate in High-Speed Networks", *IEEE/ACM Transactions on Networking*, Vol. 3, N° 1, Febrero, 1995, pp. 10-25.

[6.] Ma, S., y Ji, Ch., "Modeling Video Traffic in the Wavelet Domain", *Proc. of INFOCOM'98*, Vol. 1, San Francisco, 1998, pp. 201-208.

[7.] Taqqu, M.S., Teverovsky, V., y Willinger, W., "Estimators for Long-Range-Dependence: an empirical study", *Fractals*, Vol. 3, N° 4, 1995, pp. 785-788. Accesible como documento *postscript* en la dirección: <http://math.bu.edu/people/murad/articles.html>.

[8.] Casilari, E., Reyes, A., Díaz-Estrella, A., y Sandoval, F., "Characterization and Modelling of VBR Video Traffic", *Electronic Letters*, Vol. 34, N° 10, Mayo, 1998, pp. 968-969.

[9.] Casilari, E., Reyes, A., Díaz-Estrella, A., y Sandoval, F., "Classification and comparison of modelling strategies for VBR Video Traffic", *Proc. of International Teletraffic Congress (ITC-16)*, Vol. 3.b, Edimburgo, RU, Junio, 1999, pp. 817-826.

[10.] Heyman, D.P., y Lakshman, T.V., "What are the implications of Long-Range-Dependence for VBR-Video Traffic Engineering?", *IEEE/ACM Transactions on Networking*, Vol. 4, N° 3, Junio, 1996, pp. 301-317.

[11.] Ryu, B.K., y Elwalid, A., "The Importance of the Long-Range Dependence of VBR Video Traffic in ATM Traffic Engineering: Myths and Realities", *Proc. of ACM SIGCOMM'96*, Stanford, USA, Agosto, 1996, pp. 3-14.

[12.] Mandelbrot, B.B., *The Fractal Geometry of Nature*, Nueva York: Freeman and Company, 1977.

Un estudio por simulación del efecto de la correlación en el retardo medio del tráfico asíncrono en una red FDDI

José Carlos López Ardao Cándido López García Andrés Suárez González
Area de Ingeniería Telemática - Universidad de Vigo
ETSI Telecomunicación. Campus Universitario. 36200 Vigo (PONTEVEDRA)
Telf: 986-812176, Fax: 986-812116
E-mail: {jardao,candido,asuarez}@ait.uvigo.es

Abstract

Traffic modeling has been traditionally based almost exclusively on supposing independence in the processes that describe the random arrivals to a network. The fundamental reason for these suppositions is the analytical tractability. However, several recent traffic measurements studies have convincingly shown the presence in the traffic of the modern high-speed networks of a strong correlation that can extend over many time scales, property called long-range dependence (LRD). However, most of the works that study the impact of LRD on network performance are limited almost exclusively to single server queues that analyze the behavior of an ATM switch. Our main objective is to study, by means of simulation, a more complex case, like FDDI. Therefore, we study the impact of this long-term correlation, present in the interarrival times, on the mean waiting time.

Our main findings are that although this impact depends, basically, upon the intensity of the LRD and the network load, it can lead to significant differences in the mean waiting time, with regard to the independence assumption. On the other hand, we have also observed that the marginal distribution has a considerable impact on mean delay that depends upon its tail, when correlation is present.

1. Introducción

Uno de los principales problemas que se plantean a la hora de realizar una rigurosa evaluación de las prestaciones de una red de comunicaciones es la modelación del tráfico de entrada a la red, ya que el éxito del análisis depende en gran medida de lo representativos de la realidad que sean los modelos de tráfico utilizados [1, 2]

Históricamente, la modelación del tráfico tiene sus orígenes en la telefonía convencional, y se ha basado casi exclusivamente en suponer independencia en los tiempos entre llegadas (fundamentalmente modelos poissonianos), y duraciones exponenciales de los tiempos de uso de los recursos (ver, por ejemplo [2]). La razón más importante de tales suposiciones se halla en poder obtener modelos relativamente simples desde el punto de vista analítico. Sin embargo, el tráfico multimedia se caracteriza por una elevada variabilidad (*burstiness*) y una fuerte correlación positiva (ver, por ejemplo [3]), mucho más acentuadas que en el tráfico vocal, por lo que hoy se cuestiona enormemente la validez de los modelos más tradicionales, y en concreto las hipótesis de independencia.

Ya desde el trabajo de Kleinrock a mediados de los 60 [4], quizá el primer estudio serio que abordó el tema, la existencia de dependencia temporal ha sido objeto de estudio por parte de numerosos autores (por ejemplo, [3] y referencias allí), destacando el descubrimiento del enorme impacto que puede llegar a tener sobre las prestaciones de un sistema de colas. A la vista de este hecho, han ido apareciendo nuevos

modelos de tráfico con estructuras de correlación más o menos complejas, que se han aplicado a casos en que el modelo del sistema de comunicaciones en estudio permite mantener la tratabilidad analítica. En cualquier caso, estos modelos (fundamentalmente markovianos) desprecian la correlación a partir de determinada separación en el tiempo, aunque ésta puede ser aumentada a costa de complicar el modelo con parámetros adicionales. Por ello, habitualmente se denominan modelos de correlación a corto plazo (*Short-Range Dependence - SRD*).

Sin embargo, un trabajo presentado en ACM SIGCOMM'93 [5] (posteriormente se publicó una versión extendida [1]) mostraba, tras la observación de exhaustivas mediciones realizadas sobre una red Ethernet en Bellcore (NJ, USA), que el tráfico (número de tramas en la red por unidad de tiempo) era de naturaleza "autosimilar" (*self-similar*) o fractal. Aunque más adelante trataremos con mayor detalle la autosimilaridad y los modelos que permiten caracterizarla, de momento basta decir que implica la existencia de una acusada correlación a largo plazo (*Long-Range Dependence - LRD*).

Posteriormente, el mismo carácter autosimilar ha sido observado en redes WAN [6, 7], en el tráfico debido al *World Wide Web* (WWW) [8], o incluso en el tráfico de vídeo de tasa variable (*Variable Bit Rate - VBR*) [9, 10]. Todos estos hallazgos han contribuido enormemente a un replanteamiento muy significativo de la modelación estocástica del tráfico de entrada y del análisis de prestaciones, ya que, a pesar de que cada contribución individual a la correlación pueda ser muy pequeña, su efecto acumulado puede llegar a tener un drástico impacto sobre las prestaciones de la red.

A modo de resumen, podría decirse que la aparición de los modelos autosimilares en este campo ha resultado fundamental, debido a su capacidad de exhibir dependencia a largo plazo sobre todas las escalas temporales mediante el uso de pocos parámetros (modelación “parsimoniosa”). Como consecuencia de ello, resulta necesario realizar análisis de prestaciones con modelos autosimilares que permitan disponer de resultados y argumentos que revelen el impacto de la estructura de correlación (a largo y a corto plazo) sobre las prestaciones de las redes [11].

No obstante, y aunque el panorama ha cambiado de forma sustancial en los últimos años, Erramilli *et al.* [12] destacaban recientemente que la mayoría de los estudios realizados en este sentido se limitaban a colas de servidor único, resultando necesario el análisis de prestaciones de redes más complejas con este tipo de tráfico. En este trabajo consideramos uno de estos casos: las redes FDDI, y más concretamente, su protocolo de control de acceso al medio (MAC).

El resto del artículo se organiza de la siguiente forma. En la sección 2 se hace un breve repaso a los conceptos básicos relacionados con la LRD y la autosimilaridad. A continuación, en la sección 3 se describen las principales características de los procesos autosimilares más habitualmente utilizados en el campo de la modelación estocástica del tráfico de las redes. En la sección 4 se comentan los aspectos más relevantes del protocolo MAC de FDDI, dejando para la sección 5 la descripción del modelo de simulación, tanto de la red como del tráfico de entrada, que ha sido utilizado para obtener los resultados comentados en la sección 6. Finalmente, en la sección 7 se resumen las principales conclusiones del estudio.

2. Conceptos básicos

Pese a la estrecha relación existente entre los conceptos de “dependencia a largo plazo” y “autosimilaridad”, éstos no son equivalentes. En esta sección se establece con precisión la definición de ambos conceptos y su relación.

2.1. Dependencia a largo plazo

Se dice que exhiben dependencia o correlación a largo plazo (LRD) aquellos procesos cuya función de autocorrelación es no sumable:

$$\sum_{k=0}^{\infty} r(k) = \infty \quad (1)$$

Tal es el caso de los procesos cuya función de autocorrelación tiene una caída asintótica de tipo hiperbólico:

$$\lim_{k \rightarrow \infty} r(k) = ck^{-\beta} \quad 0 < \beta < 1 \quad (2)$$

Por oposición, se dice que exhiben dependencia o correlación a corto plazo (SRD) aquellos procesos cuya función de autocorrelación es sumable:

$$\sum_{k=0}^{\infty} r(k) < \infty$$

Tal es el caso de los procesos usados tradicionalmente para modelar el tráfico de las redes de comunicaciones, que poseen típicamente funciones de autocorrelación con una caída asintótica de tipo exponencial:

$$r(k) \approx r^k \quad \text{cuando } k \rightarrow \infty, 0 < r < 1$$

2.2. Autosimilaridad de series estacionarias

Se dice que una serie estacionaria X_k es **exactamente autosimilar** si, para todos los niveles de agregación m , se cumple que:

$$X_k \stackrel{d}{=} m^{1-H} X_k^{(m)} \quad (3)$$

donde la “d” indica igualdad en las distribuciones de orden finito, $X_k^{(m)}$ es la serie agregada de la serie original X_k , resultante del promedio sobre bloques no solapados de tamaño m , y H es el parámetro de autosimilaridad, o también **parámetro de Hurst**, en reconocimiento al trabajo de H.E. Hurst [13]

Es decir, una secuencia es exactamente autosimilar si la agregación preserva su distribución. También se dice que la serie X_k es **asintóticamente autosimilar** si (3) se mantiene de forma asintótica cuando $m \rightarrow \infty$. No obstante, habitualmente se utiliza otra definición menos estricta, debida a Cox [14], que involucra exclusivamente a los momentos de segundo orden.

Así, una serie estacionaria en sentido amplio, con varianza finita, se dice **exactamente autosimilar de segundo orden** con parámetro $H=1-\beta/2$ ($0 < \beta < 1$) si su función de autocorrelación es:

$$r(k) = \frac{1}{2} \left[(k+1)^{2-\beta} - 2k^{2-\beta} + (k-1)^{2-\beta} \right] \stackrel{\Delta}{=} g(k)$$

y, por tanto, se comporta asintóticamente como (2). Además, ello implica que la función de autocorrelación de las series agregadas, $r^{(m)}(k)$, es igual a la de la serie original para cualquier $m > 1$, es decir,

$$r^{(m)}(k) = r(k) = g(k)$$

De la misma forma, habitualmente se dice que una serie estacionaria en sentido amplio es **asintóticamente autosimilar de segundo orden** con parámetro $H=1-\beta/2$ ($0 < \beta < 1$) si se verifica:

$$\lim_{m \rightarrow \infty} r^{(m)}(k) = g(k)$$

Asimismo, en [15] se demuestra que una serie estacionaria en covarianza, con varianza finita y que posea una función de autocorrelación con una caída hiperbólica como la expresada en (2), es asintóticamente autosimilar de segundo orden.

En definitiva, el parámetro H posee una gran importancia en los procesos autosimilares, ya que es suficiente para modelar la estructura de correlación a largo plazo, cuyo efecto se hace más acusado a medida que aumenta H (entre 0.5 y 1).

3. Procesos autosimilares

Los principales procesos autosimilares estacionarios que se han venido utilizando en el campo de la modelación del tráfico de entrada son el ruido gaussiano fraccionario (FGN), los procesos ARIMA fraccionario (FARIMA) y el proceso de cuentas $M/G/\infty$.

3.1. Los procesos FGN

Un proceso FGN (*Fractional Gaussian Noise*) es el proceso resultante de considerar los incrementos de ruido browniano fraccionario (FBM) (proceso autosimilar de tiempo continuo, gaussiano, no estacionario, pero con incrementos estacionarios).

Es decir, si $B_H(t)$ es un proceso FBM, entonces la secuencia de incrementos, $X_t = B_H(t) - B_H(t-1)$, es un proceso FGN.

Los procesos FGN son estacionarios, gaussianos, de media cero y exactamente autosimilares de segundo orden. Al ser gaussianos, son también exactamente autosimilares en el sentido de la definición (3).

A pesar de su rígida estructura de correlación, los procesos FGN son una buena aproximación de la agregación de procesos LRD gaussianos [16], e incluso son una primera aproximación razonable aunque no sean gaussianos [17].

3.2. Los procesos FARIMA

Los procesos ARIMA fraccionarios (*Fractional ARIMA* o FARIMA) son una generalización de los procesos ARIMA (*Auto-Regresive Integrated Moving Average*) estándar de Box y Jenkins [18], que se han hecho enormemente populares en el campo de la modelación estocástica debido a su sencillez y flexibilidad.

Un proceso X_t perteneciente a la familia ARIMA(p,d,q) satisface la ecuación

$$\phi_p(B) (1-B)^d X_t = \theta_q(B) \varepsilon_t \quad (4)$$

donde d es un entero, ε_t es una colección de variables i.i.d. (denominadas “innovaciones”), $\phi_p(B)$ y $\theta_q(B)$ son polinomios de grado p y q en B , respectivamente, y B es el operador “retardo”, definido como

$$B^j X_t = X_{t-j}$$

La ecuación (4) puede generalizarse permitiendo que d sea un número real cualquiera, dando lugar a los procesos FARIMA [19, 20]. Para $0 < d < 1/2$, la función de autocorrelación presenta una caída hiperbólica como la expresada en (2); lo que implica que los procesos FARIMA son asintóticamente autosimilares de segundo orden [14], siendo $H=d+1/2$. Además, en el caso de que las innovaciones sean gaussianas, el proceso es también gaussiano y, por tanto, asintóticamente autosimilar.

Cabe resaltar que, para índices de correlación altos, los procesos FARIMA(p,d,q) son muy similares a los correspondientes FARIMA(0,d,0), que son los procesos FARIMA más simples que exhiben LRD.

Una de las principales ventajas de esta familia de procesos es que son mucho más flexibles que los procesos FGN, en cuanto a la modelación simultánea de la dependencia a corto y largo plazo de una serie temporal; principalmente, porque los procesos FGN poseen una estructura de correlación demasiado rígida para poder capturar un amplio rango de correlaciones a corto plazo encontradas en la práctica.

A este respecto, Hosking [20] destaca que la simple adición de una componente AR(1) o MA(1) a un proceso FARIMA(0,d,0), que da como resultado procesos FARIMA(1,d,0) y FARIMA(0,d,1), posee mucha más flexibilidad.

3.3. Procesos M/G/∞

Otro proceso autosimilar de interés es el proceso de ocupación (número de usuarios) de un modelo de colas $M/G/\infty$ [14], ya que, entre otras razones, se ha barajado como explicación de la presencia de autosimilaridad en el tráfico de las redes WAN [6].

En este modelo de colas, los usuarios llegan según un proceso de Poisson de tasa λ a una cola con infinitos servidores, cuyos tiempos de servicio son variables i.i.d. según una distribución general S de tiempo continuo, y de media finita. Se puede demostrar [21] que el proceso X_t (número de usuarios en el sistema en el instante t) es un proceso de Poisson de media $\lambda E[S]$ que es asintóticamente autosimilar de segundo orden si y sólo si la varianza de S es infinita, lo que se puede obtener mediante distribuciones de cola pesada (*heavy-tailed*), como la Pareto.

4. El protocolo MAC de FDDI

FDDI (*Fiber Distributed Data Interface*) es un conjunto de normas desarrolladas por el subcomité X3T9.5 de ANSI (*American National Standards Institute*), que se ocupa de la normalización de redes locales de alta velocidad.

Las redes FDDI utilizan como medio de transmisión un anillo de fibra óptica con una capacidad de 100 Mbps. El protocolo de control de acceso al medio [22] es un protocolo de testigo temporizado, desarrollado por R. M. Grow [23], con el objeto de soportar tráfico de tiempo real (síncrono), para el que resulta necesario poder controlar el tiempo máximo entre dos oportunidades consecutivas de transmisión, sin impedir el acceso a tráfico menos prioritario (asíncrono).

Los requisitos del tráfico síncrono se satisfacen asignando una porción del ancho de banda total a

cada estación, que será usado tras la recepción del testigo por parte de cada estación.

El ancho de banda remanente será usado para transmitir tráfico asíncrono. Para ello, cada estación posee un temporizador que contabiliza el tiempo de la última rotación del testigo (TRT). Tras recibir el testigo, la estación compara el valor de este temporizador con un valor objetivo preestablecido (TTRT). Si $TRT < TTRT$, la diferencia de tiempo puede usarse para transmitir tramas asíncronas. De no ser así, se dice que “el testigo llega tarde”, y no se permite la transmisión de tramas asíncronas.

Mediante este esquema de asignación del ancho de banda, el tiempo medio de rotación del testigo es igual a TTRT, y su valor máximo viene limitado por $2 \cdot TTRT$ [24].

El protocolo MAC de FDDI se puede modelar como un sistema de sondeo con disciplina de servicio limitado, con límite variable. Si tal límite es infinito (es decir, $TTRT = \infty$), entonces el protocolo se comporta como una disciplina de servicio exhaustivo.

5. El modelo de simulación

Haciendo uso de un simulador de eventos discretos [25], se analizará el retardo medio (tiempo medio transcurrido desde que una trama llega a una estación y comienza a ser transmitida), utilizando el siguiente escenario de red:

- 10 nodos en configuración simétrica;
- cada estación recibe el tráfico agregado procedente de varias redes locales (como Ethernet);
- tráfico asíncrono con un único nivel de prioridad;
- tamaño de las colas ilimitado.

El modelo de tráfico se describe mediante dos procesos estocásticos: uno que modela las llegadas de las tramas a cada estación, y otro que modela el tamaño de éstas.

Respecto al proceso de llegadas, ya hemos dicho que supondremos que cada nodo de la red FDDI se halla conectado a varias redes locales, y que es en el proceso de cuentas (número de tramas por unidad de tiempo, para una escala temporal dada) donde se ha observado autosimilaridad o dependencia a largo plazo. Así, habitualmente, los procesos de llegadas se describen en términos de procesos de cuentas, en lugar de tiempos entre llegadas.

No obstante, el uso de procesos de cuentas como modelo de tráfico presenta algunos inconvenientes, como la subjetividad en la elección de la escala temporal, o el problema de decidir cómo se reparten en el tiempo las llegadas dentro de cada intervalo. Para evitar todos estos problemas resultaría

enormemente útil el poder hacer uso de procesos de tiempos entre llegadas, por lo que resulta necesario analizar la relación existente entre la estructura LRD del proceso de tiempos entre llegadas y la del proceso de cuentas.

En el análisis realizado en [26] se observa la existencia de una dualidad entre los procesos de cuentas y los procesos de tiempos entre llegadas derivados de los primeros, en cuanto a que su estructura de correlación a largo plazo presenta características enormemente similares. Por tanto, ello nos permite modelar las llegadas mediante procesos de tiempos entre llegadas, sin para ello renunciar al hecho de que los procesos de cuentas resultantes exhiban autosimilaridad.

En cuanto al tamaño de las tramas, en la sección 5.1 se justifica la decisión de utilizar tamaños de trama constantes. Finalmente, en la sección 5.2 se describe el modelo de tráfico utilizado en el presente estudio.

5.1. Impacto de la distribución en el caso independiente

En [26, 27] se ha mostrado empíricamente, en el caso de utilizar procesos aleatorios puros, que dada una distribución de tiempos entre llegadas, el retardo medio depende exclusivamente de los dos primeros momentos de la distribución del tamaño de las tramas, independientemente de otros parámetros como utilización, coeficientes de variación o TTRT.

En cambio, dada una distribución del tamaño de las tramas, la distribución del tiempo entre llegadas sí posee un impacto significativo sobre el retardo medio, dependiendo en gran parte del tipo de “cola” de la distribución de tiempos entre llegadas, y de modo que cuanto menos pesada es la cola, mayor es el impacto producido sobre el retardo medio.

En conclusión, en dicho estudio se pone de relieve que los dos primeros momentos no son suficientes para caracterizar el proceso de tiempos entre llegadas, pudiendo suponerse razonablemente que la distribución entre llegadas domina sobre la de los tamaños de las tramas, en cuanto a su impacto sobre el retardo medio. Cabe esperar, además, que este carácter dominante sea mayor si el tráfico exhibe correlación. Por esta razón, nos centraremos en la modelación del proceso de llegadas.

Por otro lado, para mitigar el inevitable efecto que introduciría la varianza del tamaño de las tramas, dificultando el estudio de los comportamientos que realmente se desean analizar, hemos decidido utilizar tamaños de tramas constantes.

5.2. El modelo de tráfico

A la hora de elegir el proceso autosimilar más adecuado, descartamos a priori los procesos $M/G/\infty$ puesto que, como ya dijimos, nuestro modelo de tráfico se basará en procesos de tiempos entre llegadas en lugar de procesos de cuentas.

Por otro lado, como resulta fundamental la posibilidad de modelar simultáneamente tanto la estructura de correlación a corto como a largo plazo, descartamos FGN, que sólo permite modelar la LRD, y nos quedamos con los procesos FARIMA, que aunque son sólo asintóticamente autosimilares, permiten una mayor flexibilidad a este respecto.

Además, resulta también esencial poder variar la distribución marginal del proceso sin que ello afecte a su estructura de correlación. Se ha demostrado analíticamente que este requisito puede alcanzarse en lo que respecta a la estructura LRD si el proceso es gaussiano [28], por lo que debe utilizarse un proceso FARIMA gaussiano (es decir, con innovaciones gaussianas).

En cuanto a la estructura SRD, se desconoce el efecto debido al cambio de distribución, pero en [26] se muestra empíricamente que aunque la magnitud de las correlaciones disminuye levemente, ello no supone una objeción significativa en el análisis realizado.

Cabe decir que, a pesar de su enorme flexibilidad, los procesos FARIMA han sido poco usados en los modelos de tráfico, debido fundamentalmente a que los métodos de generación sintética propuestos en la literatura son muy costosos computacionalmente. Como solución a este inconveniente, en [26] se propone un nuevo método de generación que permite obtener directamente una buena aproximación de un proceso FARIMA(p,d,q) gaussiano con un coste computacional claramente inferior a los métodos habitualmente utilizados, y que, por tanto, será el usado para generar las trazas sintéticas utilizadas en este estudio.

Por otro lado, para evitar el manejo de modelos altamente parametrizados, creemos suficiente utilizar un proceso FARIMA(1,d,0), de forma que podamos analizar el impacto de cada tipo de correlación mediante un único parámetro: d para la LRD, y el coeficiente AR, α_1 , para la SRD.

Además, para poder suponer que el modelo no varía, independientemente del número de fuentes FARIMA(1,d,0) (redes locales) conectadas a un nodo FDDI, nos basamos en que la agregación de procesos asintóticamente autosimilares de segundo orden da lugar a otro proceso asintóticamente autosimilar de segundo orden cuyo parámetro de Hurst es igual al mayor de los procesos agregados. Este resultado se completa con el obtenido en [30] para el proceso FARIMA(1,d,0). En dicho trabajo, se muestra que la agregación del tráfico procedente de N fuentes FARIMA(1,d,0) independientes con idénticos parámetros d y α_1 , y con media μ y

varianza σ^2 , da lugar a un nuevo proceso FARIMA(1,d,0) con idénticos parámetros d y α_1 , media $N\mu$ y varianza $N\sigma^2$, es decir, el coeficiente de variación del proceso agregado se verá reducido por \sqrt{N} .

Por otro lado, ya que la disminución del TTRT, por un lado reduce la capacidad máxima disponible, y por otro lado hace que la longitud media del período ocupado aumente (aumentando así el número de correlaciones que tienen efecto sobre las prestaciones), es claro que tal disminución provocará un aumento del retardo medio. Por esta razón, y dado que sólo se cursará tráfico asíncrono, elegimos un valor del TTRT lo suficientemente elevado como para suponer servicio exhaustivo. A modo de resumen, se muestra en la Tabla 1 el resto de parámetros del modelo y los escenarios base de los experimentos de simulación.

Tabla 1: Escenario de experimentos de simulación

Tiempos entre llegadas	Proceso FARIMA(1,d,0)
Tamaño de las tramas	Cte. e igual a 1230 bytes
LRD	$H = 0.5 - 0.7 - 0.9$
SRD	$\alpha_1 = 0 - 0.3 - 0.6 - 0.9$
Distr. marginal llegadas	Unif./Exp.–Logn.–Pareto
Coef. variación llegadas	$CV = 0.25 - 0.5 - 1$
Factor de utilización	$\rho = 0.7 - 0.9$

6. Resultados

Resumimos a continuación los principales resultados obtenidos, con el objetivo de evaluar el impacto de la estructura de correlación, tanto a corto como a largo plazo, sobre el retardo medio.

A partir del impacto de los parámetros que gobiernan la estructura de correlación, α_1 y H , puede observarse en las Tablas 2 a 5 que el incremento de éstos provoca un aumento del retardo medio que, en valor relativo, se hace más apreciable cuando:

- Aumenta CV
- Aumenta ρ
- La cola de la distribución se hace más ligera

Además, se observa que la LRD contribuye a reforzar el efecto de la SRD y viceversa. No obstante, cuando se da el peor caso del resto de parámetros (dando lugar a los retardos medios más elevados), estos efectos se ven atenuados, tal y como se muestra en la Fig. 1, donde se muestra que el incremento relativo del retardo medio al variar la magnitud de la SRD para α_1 desde 0 a 0.9, usando una distribución lognormal y $\rho=0.9$. Puede observarse que el aumento de H y CV intensifican el impacto de la SRD, excepto cuando $H=0.9$ y $CV=1$.

Tabla 2: Incremento relativo del retardo medio entre un proceso aleatorio puro ($\alpha_1=0$) y un proceso AR(1) con $\alpha_1=0.9$

ρ	CV	Exp./Uni.	Lognor.	Pareto
0.9	1	864 %	732 %	276 %
	0.5	398 %	284 %	107 %
	0.25	62 %	53 %	21 %
0.7	1	343 %	163 %	-
	0.5	49 %	8.3 %	-
	0.25	-	-	-

Tabla 3: Incremento relativo del retardo medio para FARIMA(1,d,0) con H=0.7, con α_1 desde 0 a 0.9.

ρ	CV	Exp./Uni.	Lognor.	Pareto
0.9	1	1771 %	1580 %	657 %
	0.5	900 %	648 %	212 %
	0.25	115 %	95 %	29 %
0.7	1	765 %	365 %	-
	0.5	107 %	15 %	-
	0.25	-	-	-

Tabla 4: Incremento relativo del retardo medio para FARIMA(1,d,0) con H=0.9, con α_1 desde 0 a 0.9.

ρ	CV	Exp./Uni.	Lognor.	Pareto
0.9	1	384 %	418 %	775 %
	0.5	718 %	777 %	847 %
	0.25	574 %	478 %	53 %
0.7	1	1291 %	1133 %	-
	0.5	446 %	47 %	-
	0.25	-	-	-

Tabla 5: Incremento relativo del retardo medio entre un proceso aleatorio puro (H=0.5) y un proceso FARIMA(0,d,0) con H=0.9.

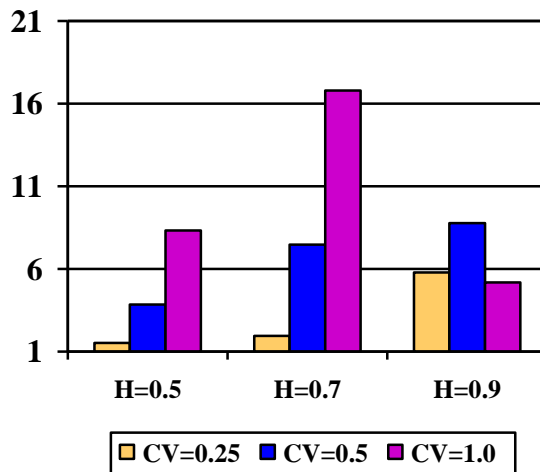
ρ	CV	Exp./Uni.	Lognor.	Pareto
0.9	1	92048 %	61607 %	4366 %
	0.5	9227 %	4790 %	343 %
	0.25	103 %	81 %	16 %
0.7	1	938 %	278 %	-
	0.5	31 %	3.8 %	-
	0.25	-	-	-

Como principal conclusión, podemos decir que, aunque en general no se puede prescindir de la LRD ni la SRD en la modelación, ello depende fundamentalmente del resto de parámetros

Por ejemplo, podemos observar que para una utilización moderada ($\rho = 0.7$), cuando CV=0.25 o la distribución es Pareto, el efecto introducido por la correlación es prácticamente inapreciable, hecho que significamos con un guión en las tablas anteriores.

Sin embargo, cuando la utilización es más elevada, ($\rho = 0.9$), en el caso de tiempos entre llegadas exponenciales se obtiene un retardo medio de 0.5935 msg., mientras que ese mismo caso, tan sólo modificado con la introducción de una acusada correlación, tanto a largo como a corto plazo, da

Figura 1: Incremento relativo del retardo medio variando α_1 desde 0 a 0.9 para $\rho=0.9$ y distribución lognormal.



como resultado un retardo medio de algo más de 2.6 segundos !!!, es decir, unas 4500 veces más.

En cuanto a la distribución marginal de los tiempos entre llegadas, ésta posee un enorme impacto sobre el retardo medio, que puede agravarse drásticamente debido a la correlación. Así, por ejemplo, en el caso sin correlación se obtiene una diferencia máxima del 70%, entre las distribuciones exponencial y Pareto con CV=1; mientras que cuando $H = 0.9$ y $\alpha_1=0.9$, la distribución exponencial da lugar a un retardo medio 20 veces superior frente a la Pareto. E incluso la distribución lognormal da lugar a un retardo 13 veces superior.

Puede observarse, pues, que el impacto debido a la distribución viene determinado fundamentalmente por el tipo de cola exhibido por la distribución, que, como vimos, puede incluso reducir dramáticamente el impacto debido a la correlación.

Para finalizar este estudio, sería de gran interés comparar la contribución al retardo medio de cada estructura de correlación analizada. Puede verse que, en todos los casos, los mayores retardos medios se dan cuando la LRD es acusada (H=0.9), aumentando con el valor de α_1 . A continuación se hallan los otros dos casos con $\alpha_1=0.9$ (H=0.7 y H=0.5); seguidos de los de H=0.7 con $\alpha_1=0.6$ y $\alpha_1=0.3$. A éstos siguen el proceso AR(1) con $\alpha_1=0.6$, y FARIMA(0,d,0) con H=0.7, para finalizar con los procesos AR(1) para $\alpha_1=0.3$ y los procesos aleatorios puros (H=0.5 y $\alpha_1=0$).

Cabe destacar, sin embargo, que los procesos AR(1) con $\alpha_1=0.9$ dan lugar a retardos medios muy similares, aunque superiores, a los obtenidos con H=0.7 y $\alpha_1=0.6$; y que lo mismo ocurre con los procesos AR(1) con $\alpha_1=0.6$, para los que resultan retardos medios similares a los obtenidos con procesos FARIMA(0,d,0) con H=0.7, aunque también ligeramente superiores. No obstante, en los dos escenarios que dan lugar a los retardos medios mayores en ambas situaciones ($\rho=0.9$, CV=1 y distribución exponencial o lognormal), éstos resultan ser ligeramente superiores para las estructuras con

$H=0.7$, sin duda debido a que, en dichos casos, el considerable aumento de la longitud media del período ocupado hace que aumente el número de correlaciones que interactúan y, por tanto, que la correlación a largo plazo cobre mayor importancia.

De estos hechos se pueden extraer interesantes conclusiones a partir de la observación de sus funciones de autocorrelación. Como puede verse en la Fig. 2, aproximadamente hasta el coeficiente de orden 5, las autocorrelaciones del proceso AR(1) (con $\alpha_1=0.6$) son mayores que las de FARIMA(0,d,0) (con $H=0.7$), a partir del cual las del proceso AR comienzan a caer exponencialmente a cero (hecho que ocurre en torno al coeficiente de orden 10). En cambio, la caída hiperbólica del proceso FARIMA hace que la magnitud de las correlaciones caigan muy lentamente (0.013 para el coeficiente de orden 1000). Así, desde el punto de vista del análisis del impacto de la correlación, si se usase un proceso que ajustase sólo la estructura SRD del proceso FARIMA para, digamos, los 5 primeros coeficientes, éste subestimaría considerablemente el retardo medio, pues la magnitud de las correlaciones del proceso que daría lugar a un retardo medio similar es claramente superior (basta ver que el primer coeficiente de autocorrelación es 0.6 para el proceso AR y tan sólo 0.25 para el FARIMA). Conclusiones similares podrían extraerse de la observación de las funciones de autocorrelación en el otro caso. Por otro lado, si comparamos las funciones de autocorrelación de un proceso FARIMA(0,d,0) para $H=0.9$, cuyo primer coeficiente de autocorrelación es 0.66, con la de un proceso

FARIMA(1,d,0) con $H=0.7$ y $\alpha_1=0.9$, para el que el primer coeficiente de autocorrelación es próximo a la unidad (ver Fig. 3), podemos observar que si ignoramos la correlación a largo plazo, el error cometido al estimar el retardo medio puede ser de una magnitud más que considerable. Basta ver, por ejemplo, que usando una distribución exponencial

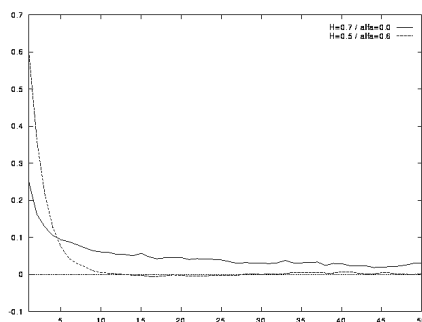


Figura 2: Funciones de autocorrelación de FARIMA(0,d,0) con $H=0.7$ y AR(1) con $\alpha_1=0.6$

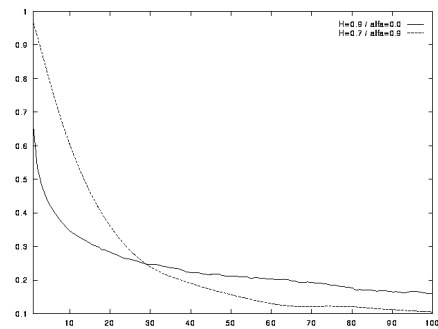


Figura 3: Funciones de autocorrelación de FARIMA(0,d,0) con $H=0.9$ y FARIMA(1,d,0) con $H=0.7$ y $\alpha_1=0.9$

para $\rho=0.9$, el proceso que posee una menor magnitud en las correlaciones a corto plazo da lugar a un retardo medio casi 20 veces superior al obtenido con el otro proceso, incluso aunque éste exhibe una LRD moderada.

7. Conclusiones

En este artículo se analiza el retardo medio en ciertos escenarios de una red FDDI cuando el tráfico de entrada, y más concretamente el proceso de tiempos entre llegadas, exhibe correlación, tanto a corto como a largo plazo. Para modelar la LRD, se usan procesos autosimilares FARIMA(1,d,0) gaussianos, que permiten modelar simultáneamente la SRD.

La conclusión principal que se puede extraer de este estudio es que, en general, no se puede prescindir de la LRD en la modelación, ni despreciar el efecto debido a la magnitud de la correlación a más corto plazo, aunque ello depende fundamentalmente del resto de parámetros. Así, en aquellos casos en que el retardo medio es mayor debido a la contribución del resto de parámetros (utilización y CV elevados, y distribuciones de cola no pesada), el impacto de la estructura de correlación puede llegar a tener efectos drásticos (retardos medios hasta 4500 veces superiores a los obtenidos con procesos aleatorios puros). Sin embargo, para valores moderados de los citados parámetros (utilización 0.7; con $CV=0.25$ o con distribuciones de cola pesada), el efecto introducido por la estructura de correlación es prácticamente inapreciable.

Por otro lado, se ha observado que la contribución individual de cada parámetro del modelo de tráfico (ρ , H , α_1 , CV y distribución marginal) al retardo medio se ve intensificada a medida que aumenta la del resto de parámetros. Sin embargo, alcanzado cierto umbral (generalmente próximo al peor caso para cada parámetro), el impacto de cada contribución individual se ve reducido.

En cuanto a la distribución marginal, ésta posee un enorme impacto sobre el retardo medio, determinado fundamentalmente por el tipo de cola exhibido por la distribución. Así, a medida que la cola de la distribución se hace más pesada, los retardos medios disminuyen, a veces de forma considerable.

Referencias

- [1] W.E. Leland, M.S. Taqqu, W. Willinger, y D.V. Wilson. "On the self-similar nature of Ethernet traffic (extended version)". IEEE/ACM Trans. on Networking, 2(1):1-15, Feb. 1994.
- [2] V. Frost y B. Melamed. "Traffic modeling for telecommunications networks". IEEE Comm. Mag., 32:70-80, 1994.
- [3] K.W. Fendick, V.R. Saksena, y W. Whitt. "Dependence in packet queues". IEEE Trans. on Communications, 37:1173-1183, 1989.
- [4] L. Kleinrock. "Communication Nets". McGraw-Hill, New York, 1964.
- [5] W.E. Leland, M.S. Taqqu, W. Willinger, y D.V. Wilson. "On the self-similar nature of Ethernet traffic". Computer Communication Review, 23:183-193, 1993. Este artículo fue presentado en ACM SIGCOMM '93.
- [6] V. Paxson y S. Floyd. "Wide-area traffic: The failure of Poisson modeling". IEEE/ACM Trans. on Networking, 3:226-244, Jun. 1995. Una versión más corta de este artículo fue presentada en ACM SIGCOMM '94.
- [7] S. Klivansky, S. Mukherjee, y C. Song. "Factors contributing to self-similarity over NSFNet". Georgia Institute of Technology (preprint), 1994.
- [8] M.E. Crovella y A. Bestavros. "Self-similarity in World Wide Web traffic: Evidence and possible causes". IEEE/ACM Transactions on Networking, 5(6):835-846, Dic. 1997.
- [9] M.W. Garrett y W. Willinger. "Analysis, modeling and generation of self-similar VBR video traffic". En Proc. ACM SIGCOMM '94, pp. 269-280, Londres, 1994.
- [10] J. Beran, R. Sherman, M.S. Taqqu, y W. Willinger. "Long-range dependence in variable bit rate video traffic". IEEE Trans. on Communications, 43(2/3/4):1566-79, 1995.
- [11] W. Willinger. "Traffic modeling for high-speed networks: Theory versus practice". En F.P. Kelly y R.J. Williams (editores), Stochastic Networks, volumen 71 de IMA Volumes in Mathematics and its Applications, pp. 395-409. Springer Verlag, 1995.
- [12] A. Erramilli, O. Narayan, y W. Willinger. "Fractal queueing models". En Frontiers in Queueing, pp. 245-269. CRC Press, Inc., 1997.
- [13] H.E. Hurst. "Long-term storage capacity of reservoirs". Trans. Am. Soc. Civil Engineers, 116:770-799, 1951.
- [14] D.R. Cox. "Long range dependence: A review". En H.A. David y H.T. David (editores), Statistics: An Appraisal, pp. 55-74. Iowa State University Press, 1984.
- [15] B. Tsybakov y N.D. Georganas. "On self-similar traffic in ATM queues: Definitions, overflow probability bound, and cell delay distribution". IEEE/ACM Transactions on Networking, 5(3):397-409, Jun. 1997.
- [16] G. Samorodnitsky y M.S. Taqqu. "Stable Non-Gaussian Random Processes: Stochastic Models with Infinite Variance". Chapman & Hall, 1994.
- [17] M.S. Taqqu y V. Teverovsky. "On estimating the intensity of long-range dependence in finite and infinite variance time series". En R. Adler, R. Feldman y M.S. Taqqu (editores), A Practical Guide to Heavy Tails. Birkhauser, 1998.
- [18] G.E.P. Box, G.M. Jenkins, y C. Reinsel. "Time Series Analysis: Forecasting and Control". Prentice Hall, Englewood Cliffs, NJ, 1994.
- [19] C.W.J. Granger y R. Joyeux. "An introduction to long-range time series models and fractional differencing". Journal of Time Series Analysis, 1:15-30, 1980.
- [20] J.R.M. Hosking. "Fractional differencing". Biometrika, 68(1):165-176, 1981.
- [21] M. Parulekar y A.M. Makowski. "Tail probabilities for a multiplexer with self-similar traffic". En Proc. IEEE INFOCOM '96, San Francisco, CA, Mar. 1996.
- [22] ANSI X3.139-1987. "Information Systems-Fiber Distributed Data Interface (FDDI)-Token Ring Media Access Control (MAC)". También ISO 9314-2:1989.
- [23] R.M. Grow. "A timed token protocol for local area networks". En Proc. Electro '82, 1982.
- [24] M.J. Johnson. "Proof that timing requirements of the FDDI token ring protocol are satisfied". IEEE Transactions on Communications, 35:620-625, Jun. 1987.
- [25] C. López García. "Una Aproximación Metodológica al Análisis del Retardo en Redes de Alta Velocidad con Protocolos de Control de Acceso al Medio de Testigo Temporizado". Tesis Doctoral, UPM Madrid, 1995.
- [26] J.C. López Ardao. "Contribución al Análisis del Impacto de la Correlación en las Prestaciones de las Redes de Alta Velocidad". Tesis Doctoral, Universidad de Vigo, 1999.
- [27] J.C. López Ardao, A. Suárez González, y C. López García. "On the effects of long-range dependence on the performance of FDDI". En Proc. 7th. IEEE Workshop on Computer-Aided Modeling, Analysis and Design of Communication Links and Networks (CAMAD'98), Sao Paulo, Aug. 1998.
- [28] C. Huang, M. Devetsikiotis, I. Lambadaris, y A.R. Kaye. "Modeling and simulation of self-similar variable bit rate compressed video: A unified approach". En Proc. ACM SIGCOMM '95, pp. 114-125, Cambridge, MA USA, 1995.
- [29] Y. Fan y N.D. Georganas. "On merging and splitting of self-similar traffic in high-speed networks". En Proc. ICC '95, Seúl, Aug. 1995.
- [30] A. Adas y A. Mukherjee. "Delay-jitter bound and statistical loss bound for heterogeneous correlated traffic - architecture and equivalent bandwidth". Technical Report GIT-CC-96-05, Georgia Institute of Technology, Abr. 1996.

Técnicas eficientes de filtrado y análisis de tráfico para la monitorización continua de redes de comunicaciones

¹J.J. Ruiz, ²E. Magaña, ³J. Aracil, ⁴J. Villadangos
Departamento de Automática y Computación
Universidad Pública de Navarra
Campus de Arrosadía s/n, 31006 Pamplona
Telf: 948168904, Fax: 948169281

Email: ¹jr18315@zurron.upna.es, {²eduardo.magana, ³javier.aracil, ⁴jesusv,}@upna.es

Abstract

This paper presents an efficient traffic filtering and analysis architecture for network monitoring. Opposed to the usual network monitoring architectures that provide simultaneous filters as requested by managers (packet filters), we propose a different approach that aims at minimizing CPU load by avoiding unnecessary filter duplicates. Such architecture makes it possible to optimize several parallel filters execution and thus is suitable for continuous network monitoring in which it is necessary to keep track of hundreds of filters. This architecture has been implemented in a network-monitoring tool called PROMIS whose main features are detailed in this paper.

1. Introducción

Las redes corporativas se han convertido en uno de los elementos estratégicos de cualquier compañía de medio y gran tamaño. Las tecnologías de red evolucionan rápidamente, haciendo que las pequeñas redes locales aisladas se conviertan en grandes sistemas de información multiservicio. Incluso se está produciendo la convergencia entre las redes telefónicas y de datos, tendiendo a un acceso común para voz (VoIP), datos, imagen y vídeo. Así, puesto que estas redes de datos constituyen una parte fundamental de cualquier actividad empresarial es preciso asegurar su disponibilidad mediante un sistema de monitorización de red [3][5].

Estos sistemas de monitorización de red disponen de elementos activos que recogen el tráfico de la red en modo promiscuo (sondas) y su funcionalidad se agrupa en tres grandes grupos: realización de capturas de tráfico, programación de alarmas y recogida de estadísticas como, por ejemplo, número de paquetes IP que produce una máquina. Para realizar estas funciones es necesario disponer de filtros de paquetes en las sondas que monitorizan la red. Esta necesidad viene dada por el altísimo grado de concurrencia de este tipo de sistemas, en el entorno de alta velocidad y gran volumen de tráfico generado que presentan las actuales redes de comunicaciones. Así, un sistema de monitorización típico puede estar recogiendo estadísticas específicas como los paquetes/sg entre dos máquinas a nivel IP, mientras simultáneamente se verifican situaciones de alarma y se realizan las capturas programadas. En este escenario es imprescindible disponer de un filtro que sólo deje pasar los paquetes IP de la máquina que se pretende monitorizar, para realizar el cálculo de estadísticas a partir de estos paquetes solamente y no a partir del tráfico total en bruto, que cargaría innecesariamente al

proceso de cálculo de estadísticas con un excesivo número de eventos.

Observamos entonces que la técnica de filtrado de paquetes determina en gran medida las prestaciones de un sistema de monitorización. En el caso de redes de medio/gran tamaño es práctica común la realización de un gran número de capturas, alarmas o recogida de estadísticas. Cada una de estas operaciones depende de un filtro para seleccionar los paquetes que le conciernen. Así, las técnicas de filtrado y análisis de tráfico constituyen un aspecto esencial en el diseño de sistemas de monitorización de red.

Precisamente, este artículo propone una arquitectura de filtrado y análisis de tráfico destinada a su utilización en un sistema de monitorización de redes de comunicaciones. La aplicación de la técnica propuesta resulta en la optimización de series de filtros simultáneos. La idea intuitiva de esta nueva técnica es la utilización de los niveles de los protocolos de red para la organización del filtrado, de forma que se aprovechan filtrados comunes en niveles inferiores aunque difieran los filtrados a niveles más altos. Como resultado se logran dos objetivos fundamentales en un sistema de monitorización de red. Por un lado, la flexibilidad necesaria en cuanto a la programación de filtros, capturas y recogida de estadísticas específicas. Por otro lado, la eficiencia necesaria para que el sistema permita la monitorización continua, sin interrupción, de redes de alta velocidad.

El resto del artículo se organiza de la siguiente forma: en la sección 2 se expone el estado del arte en sistemas de filtrado y la motivación para presentar uno nuevo. En la sección 3 se comentan los problemas típicos a los que se enfrenta una implementación de filtrado de protocolos y en la sección 4 se expone la arquitectura del sistema de monitorización PROMIS que hace uso de este núcleo

de filtrado. Las secciones 5 y 6 se centran en la especificación de la arquitectura de filtrado y rendimiento del sistema, para terminar en la sección 7 con las conclusiones.

2. Motivación

El filtrado de protocolos en un sistema de monitorización de redes de comunicaciones se puede realizar desde el kernel del sistema operativo o desde un proceso de usuario. En el primer caso el código es más difícil de desarrollar y mantener, mientras que en el segundo caso se produce una sobrecarga de procesamiento y una bajada del rendimiento, debido al incremento de cambios de contexto y llamadas al sistema que supone.

Una solución de equilibrio que se utiliza normalmente en sistemas UNIX es el *packet filter*, que permite una programación de filtros desde el nivel de usuario con una parte del kernel que filtra paquetes según los criterios especificados al nivel de usuario, resultando el proceso eficiente. Esta parte del kernel es la encargada de demultiplexar paquetes. El *packet filter* aísla el kernel de los detalles de implementación de los protocolos. La interfaz que provee el *packet filter* otorga facilidades para transmisión y recepción de paquetes, e información y control de estado. Algunos ejemplos de *packet filter* son el BPF de sistemas BSD[1] y el CSPF [2], que fue la primera tentativa. Otros sistemas que permiten el acceso de los procesos de usuario a los paquetes en el nivel de enlace son el DLPI de Solaris y el Snoop de sistemas Irix.

A continuación se presentan las características de estos sistemas de *packet filter*:

- CSPF(CMU/ /Stanford Packet Filter): utiliza un árbol de expresiones booleanas, en el que los nodos representan operaciones booleanas y las hojas campos del paquete a comparar. Sus limitaciones son que no puede tratar cabeceras de tamaño variable y trabaja a 16bits.
- BPF (BSD Packet Filter): realiza el filtrado a nivel de kernel según unos filtros definidos a nivel de usuario y copia solo la parte del paquete solicitada. El filtrado está basado en registros y proporciona un buffer en el que se van almacenando los paquetes recibidos junto con su timestamp y longitud, disminuyendo el número de llamadas al sistema que ha de realizar la aplicación. El filtrado se basa en un grafo dirigido en el que cada nodo representa un campo del paquete, según el resultado del test se atraviesa una rama o la otra y sólo hay dos hojas que representan los valores true/false para todo el filtro.

Ambos filtros son equivalentes computacionalmente, pero el BPF está optimizado

para la arquitectura de máquinas basadas en registros.

Estos *packet filter*, aunque proporcionan una herramienta útil para el análisis parcial del tráfico que circula por la red, no son de aplicación en el caso de monitorización de redes de medio/gran tamaño. En el caso de análisis de redes que nos ocupa en este artículo se pretende monitorizar la red de forma global, lo que supone miles de filtros simultáneos y procesado de todos los paquetes de la red. En estos entornos, con enlaces de alta velocidad, un *packet filter* produce una carga excesiva del kernel por los posibles filtros duplicados, que conlleva la pérdida de paquetes y, por tanto, fallos graves de monitorización.

Para ilustrar este fenómeno de duplicación definimos *captura* como un conjunto de filtros de los diferentes campos de cada trama (por ejemplo, paquetes con una dirección IP destino determinada por el puerto TCP 80 –WWW-), como se muestra en la Fig. 1. Se definen tres capturas que comparten algunos filtrados intermedios: por ejemplo, la segunda captura supone comprobar en los paquetes si poseen cierta dirección MAC₁ origen, cualquier dirección MAC destino, cierta dirección IP₃ origen y que además sea un paquete TCP. Se observa como la tercera captura comparte el filtro de dirección IP₃ origen con la segunda captura, entre otros filtros comunes. Así, el diseño de estas capturas se puede enfocar de dos maneras distintas:

- Comprobar que cada paquete satisfaga cada uno de los filtros de cada captura, de forma independiente entre cada captura, como indica la Fig. 2. Esta es la técnica de filtrado de los *packet filter* CSPF y BPF, que resultaría en la aplicación de 12 filtros.
- O bien, hacer que se compruebe que cada paquete satisfaga cada uno de los filtros, pero de forma que si una serie de capturas tienen en común un mismo filtro, esta comprobación sea común para todas esas capturas. De esta forma, la comprobación de que un paquete cumple o no un filtro determinado se realiza una sola vez para todo ese conjunto de capturas, como se indica en la Fig. 3. Resultan 6 filtros, la mitad que en los anteriores.

Captura1	MAC ₁ origen	MAC ₂ destino	IP ₃ origen	IP ₄ destino	UDP
Captura2	MAC ₁ origen	IP ₃ origen	TCP		
Captura3	MAC ₂ destino	IP ₃ origen	IP ₄ destino	UDP	

Fig. 1: Capturas propuestas

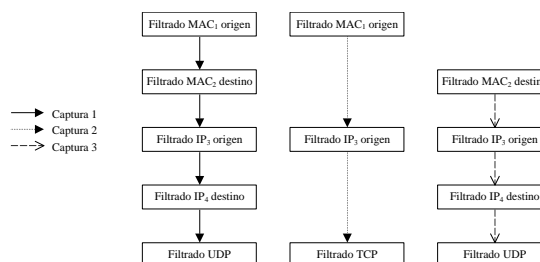


Fig. 2: Capturas por filtrado mediante CSPF y BPF

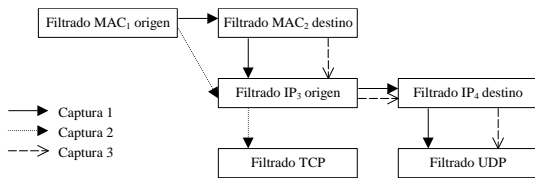


Fig. 3: Capturas por filtrado en árbol

Comparando ambas estrategias se observa que la segunda opción minimiza el número de comprobaciones para cada paquete y un mismo número de filtrados. Por ejemplo, si se quiere realizar un cierto número X de capturas, de forma que cada una de ellas tengan los mismos N filtros, es decir, repetimos X veces la misma captura. En la primera estrategia se tiene una cifra de $N \times X$ comprobaciones en los filtros para cada paquete, mientras que en la segunda estrategia se tienen únicamente N comprobaciones.

La técnica propuesta se basa en la segunda estrategia y se ha seleccionado una disposición de los diferentes filtros en forma de una estructura en árbol, de modo que se identifica cada filtro con un nodo del árbol. Cada nodo representa una comprobación de cierto campo de determinado protocolo, observándose la estructura lógica de niveles de protocolos en el diseño del árbol. Si distintos filtrados comparten etapas de filtrado comunes, compartirán los nodos del árbol evitando la realización de filtrados duplicados como en los otros sistemas de *packet filter*.

3. Problemas de Sistemas de Filtrado y Análisis de Tráfico de Red

Todos los sistemas de captura de paquetes, ya sean basados en *packet filter* o al nivel de usuario, muestran una serie de problemas [4] o limitaciones entre los que destacan los que se citan a continuación:

- La precisión del Timestamp (marca temporal con la que se ha recibido el paquete): depende del sistema sobre el que funcione, del orden de centésimas de segundos para un PC, o de microsegundos para una SUN Sparc. En todo caso, se garantiza que cada paquete recibido posee un timestamp mayor o igual que el paquete anterior.
- Pérdidas de paquetes: se puede producir a dos niveles. En el ámbito de la tarjeta de red, cuando no se hacen lecturas sobre la tarjeta a suficiente velocidad y esta ha de sobrescribir los paquetes recibidos en su buffer, normalmente de tamaño pequeño, del orden de Kbytes. El otro ámbito se produce en el buffer intermedio, cuando el proceso de filtrado y análisis no es lo suficientemente rápido.
- Generación de paquetes fantasma, por fallos de la implementación del soporte de red. Por ejemplo, en tasas alta de carga se ha detectado como en Linux se capturan paquetes de tamaño menor al mínimo permitido en una

Ethernet (64 bytes). En realidad, estos paquetes no existen sobre la red y son creados por el driver de la tarjeta de red cuando se intenta mandar paquetes a la red desde la misma máquina, estando la tarjeta en modo promiscuo.

Estos aspectos se han tenido en cuenta en la implementación de la técnica de filtrado propuesta, evitando o minimizando sus efectos sobre los resultados finales. Dicha técnica de filtrado es el núcleo del sistema de monitorización distribuido PROMIS [7], diseñado por los autores de este trabajo, que se presenta brevemente en el siguiente apartado.

4. Sistema PROMIS

El sistema de monitorización PROMIS está compuesto de sondas y consolas. Las sondas están implementadas en PCs con Linux como sistema operativo, y son las encargadas del filtrado y análisis de tráfico de la red, además de servir la información procesada a la consola. Se coloca una sonda por cada segmento de red que se desee monitorizar. La consola provee la interfaz de usuario, a través del cual el gestor de la red puede acceder a la información recogida por las sondas. Esta información puede ir desde gráficos de monitorización en tiempo real de un determinado parámetro de tráfico hasta la recepción de capturas, pasando por la recepción de alarmas ante problemas detectados en las redes. Los parámetros que se pueden obtener pueden ser bytes/sg, paquetes/sg o utilización media, globales o filtrados por máquina, protocolo, servicio y/o aplicación. Además se lleva cuenta de las "top N" máquinas que más tráfico generan y la matriz de tráfico entre ellas. Se posee un mecanismo flexible de definición de alarmas en función de umbrales de determinado parámetro, y también se pueden definir capturas de parámetros para analizarlas con posterioridad. El sistema se completa con una serie de informes accesibles desde la Web con los parámetros resumidos de evolución de la red a lo largo de la última hora, día, semana y mes.

5. Arquitectura de filtrado

5.1. Arquitectura de la sonda

El procesamiento de cada paquete o trama que se recibe por la red se realiza mediante el software instalado en la sonda conectada a esa red. El software está desarrollado en lenguaje C, sobre sistema operativo Linux, y comprende cinco procesos concurrentes y dos memorias compartidas:

- Un proceso "*lector*", que se encarga de recoger los paquetes de la red, y el proceso "*master*", que se encarga del análisis de dichos paquetes. Además existen tres procesos "*servidores*", que son los encargados de comunicar la sonda con la consola y de suministrarle los datos y parámetros que solicite.
- Las dos zonas de memoria reservada se utilizan para el intercambio de datos entre los diferentes procesos. Una de ellas se implementa para el

paso de los paquetes que recoge el proceso "lector" hacia el proceso "master", actuando como buffer intermedio que permite almacenar paquetes en periodos de pico de tráfico, y de esta forma evitar la pérdida de paquetes comentada en el apartado 3. La otra memoria compartida es la utilizada por el proceso "master" para almacenar los parámetros ya analizados para que los procesos servidores puedan disponer de ellos.

El proceso "master" es el que realiza el filtrado y análisis del tráfico propiamente dicho, y es en la zona de memoria de este proceso donde se localiza el árbol de filtrado. El árbol de filtrado, aunque funciona internamente en este proceso, accede a la segunda memoria compartida para almacenar los parámetros calculados y hacerlos así accesibles a los procesos servidores.

5.2. Arbol de filtrado

La técnica de filtrado y análisis propuesta se basa en una estructura en árbol. Se organiza por niveles de profundidad, actuando todos los nodos del mismo nivel sobre el mismo campo de la trama recibida, es decir, actúan en la misma posición del paquete (*offset*). Conforme se desciende por el árbol, recorriendo la trama campo a campo, se comprueba el contenido de los campos. La información obtenida será filtrada por los nodos del nivel correspondiente a ese campo. Cuando el filtrado sea positivo para un determinado nodo se puede seguir descendiendo por el árbol a través de ese nodo.

Dada la disposición en árbol, se provee de una absoluta libertad a la hora de configurar el conjunto de filtros de una determinada captura. Además, se tiene la suficiente flexibilidad como para poder actualizar el árbol en tiempo de ejecución, pudiendo añadir nuevos nodos-filtros mientras se siguen procesando tramas recogidas de la red. En cada uno de los nodos-filtros es posible definir una serie de nodos-parámetro, que contendrán la información necesaria para tomar los parámetros requeridos en esa captura. Estos nodos sólo son recorridos y actualizados en caso de que se cumpla el filtro al que están asociados.

Como se puede observar en la Fig. 4, se distingue entre filtrado obligatorio y filtrado opcional en los distintos niveles. El primer caso se refiere a que en un nivel determinado se obliga a un paquete a que cumpla uno, y sólo uno, de los filtros definidos en ese nivel. El segundo hace referencia a que existen niveles en los que se definen estructuras nodo-filtro que no tienen función de filtrado, con lo que se puede pasar directamente al nivel inferior. De esta forma es posible la combinación de conjuntos de filtros en los que no se tiene que especificar un filtro por cada uno de los niveles.

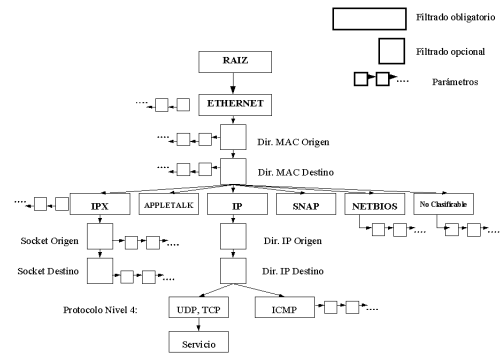


Fig. 4: Ejemplo de árbol de filtrado sobre red Ethernet

La necesidad de fijar niveles de filtrado obligatorio se debe a que es necesario obtener forzosamente información del campo de la trama que se está chequeando en ese momento para poder seguir descendiendo por los nodos del árbol. Por ejemplo, es necesario comprobar el campo situado a partir del byte 13 de una trama Ethernet para saber qué protocolo encapsula la trama. El esquema del árbol de la Fig. 4 representa un conjunto de filtrados que se pueden realizar sobre una red Ethernet. Para optimizar la flexibilidad del árbol, no se ha puesto el nodo-filtro Ethernet como raíz de la estructura, sino que se ha añadido por encima de este nodo un nivel más. Esto permite que bajo el nodo raíz se puedan disponer varios nodos, que coincidirían con diferentes protocolos de nivel de enlace, como podrían ser Token Ring, Token Bus, ATM, Fast Ethernet, Gigabit Ethernet... De esta forma, la misma sonda se podría conectar en diferentes redes sin necesidad de modificar la estructura del árbol, y por tanto, sin modificar el software.

Para implementar el árbol se ha definido una estructura de datos única, de forma que todos los nodos del árbol, sin importar el nivel en el que se encuentren, se configuren mediante la misma estructura. Esto puede producir un menor rendimiento en el uso de la memoria del sistema, ya que existen nodos cuyos parámetros no ocupan completamente la zona de memoria asignada de los campos donde se almacenan. Pero el hecho de utilizar este tipo de estructura única simplifica y agiliza, en cuanto a velocidad, la construcción, borrado y recorrido del árbol de filtrado.

Mediante la ayuda de cuadrados, que indican cada nodo, y flechas, que indican el enlace de nodos (en términos de memoria) mediante punteros, en la Fig. 5 se representa la forma del árbol que queda con la utilización de la estructura de nodo anterior. Sin embargo, existen cuatro niveles en el árbol Ethernet donde la disposición de nodos comentada no se satisface. El cambio de estructura se debe a la búsqueda de una disposición alternativa que optimice la velocidad de recorrido del árbol en los niveles de dirección MAC origen, MAC destino, IP origen e IP destino.

La estructura de parámetro es la encargada de llevar los contadores de los bytes o paquetes que cumplen el filtro asociado y que también se representa en la Fig. 5. Como en las estructuras de nodo, se intenta que con un solo tipo de estructura se puedan implementar todos los parámetros definidos en el sistema. La estructura de parámetro puede ir unida a una estructura de nodo o a otra estructura de parámetro, como puede observarse en la Fig. 5.

5.2.1. Posible estructura del árbol de filtrado en una red Ethernet

El descenso de nivel por el árbol coincide con un desplazamiento por los diferentes campos que conforman la trama Ethernet.

- Nivel 0: La raíz del árbol. Este nodo no realiza ningún tipo de filtrado y tan sólo sirve como soporte de todos los niveles que se creen bajo él.
- Nivel 1: Filtrado de tipo de red LAN. Desde este nivel se pueden especificar las diferentes ramas para cada uno de los protocolos Ethernet, Token Ring, ATM... Este nodo es de filtrado obligatorio, ya que hay que saber que protocolo de nivel de enlace se está soportando.

A partir de este nivel la configuración del árbol se ha implementado únicamente para pilas de protocolos sobre Ethernet.

- Nivel 2: Filtrado de dirección MAC origen. Cada nodo de este nivel, exceptuando si existe el nodo de “no filtrado”, tendrá en su identificador una dirección MAC. Este nivel es de filtrado opcional, con lo que se puede incluir el nodo de “no filtrado”.
- Nivel 3: Filtrado de dirección MAC destino. Igual que el anterior, pero chequeando el campo de la dirección destino de la trama.
- Nivel 4: Filtrado por protocolo de red. Cada uno de los nodos indica un protocolo de red. En su función de filtrado existe un algoritmo que chequea la estructura de la trama, identificando el tipo de protocolo definido a continuación en la trama. Dependiendo del protocolo identificado, el puntero que recorre la trama se desplaza un *offset* determinado para situarse en el primer campo de la cabecera de red.

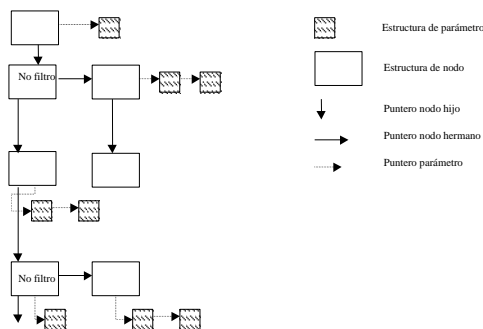


Fig. 5: Esquema de árbol

Dado que es necesario el conocimiento de dicho protocolo, este nivel es de filtrado obligatorio.

- Nivel 5: A partir de este nivel hay que distinguir entre los posibles protocolos de nivel de red que se hayan definido. Por ejemplo, en:

1. Protocolo IP: Filtrado de dirección IP origen. En cada nodo se pone una dirección IP diferente. En este nivel también se admite el nodo de “no filtrado”, dado que el nivel es de filtrado opcional.
2. Protocolo IPX: Filtrado de punto de acceso al servicio origen. Al igual que el protocolo IP es de filtrado opcional.

- Nivel 6: También hay que distinguir entre protocolos:

1. Protocolo IP: Filtrado de dirección IP destino, con las mismas propiedades que el nivel anterior.
2. Protocolo IPX: Filtrado de punto de acceso a servicio destino. Al igual que el protocolo IP es de filtrado opcional.

Este nivel permite realizar filtrados para propósito de filtrado de túneles.

- Nivel 7: A partir de este nivel se podría implementar, por ejemplo, el protocolo IP. En este nivel se distingue entre protocolos TCP y UDP, y los paquetes ICMP que circulan por la red.

- Nivel 8: Para aquellos paquetes TCP y UDP se filtra el puerto por el que se manda, es decir, se filtra el servicio.

Este es un ejemplo de árbol para una sonda sobre redes Ethernet. Se comprueba que la topología utilizada permite una gran flexibilidad para incrementar, ya sea el número de niveles de profundidad, ya sea el número de nodos dentro de un mismo nivel. Además, dado que cada nodo es independiente de sus hermanos y sólo depende de su nodo padre, es posible la realización de subárboles a partir de cualquier nodo, teniendo estos nuevos árboles la configuración que se desee.

5.2.2 Creación y borrado de nodos en el árbol de filtrado

El proceso master es el encargado, además de analizar las tramas tomadas de la red, de mantener actualizado el árbol de filtrado. Si se desea crear una nueva captura, los procesos servidores proporcionan toda la información de dicha captura al proceso master a través de la Memoria Compartida 2.

El sistema de creación de nodos es el mismo para todos los niveles, al igual que el sistema de creación de parámetros, de tal forma que el algoritmo utilizado no depende de la información que contenga la directriz suministrada por los procesos servidores ni del nivel en el que se vaya a situar el nuevo nodo. El descenso por el árbol se realiza mediante llamadas a sí misma de la función anterior, de modo recursivo. Dicha función es capaz de decidir si es posible la creación de un nodo de “no filtrado” en un determinado nivel, así como decidir cuándo se llega

al punto de comenzar a introducir nodos de parámetro.

Sin embargo, para optimizar la velocidad de recorrido del árbol existen cuatro excepciones a la regla anterior. Dichas excepciones se sitúan en el nivel 2 y 3 del árbol Ethernet (que coinciden con los niveles de filtrado de direcciones MAC), y el nodo 5 y 6 del árbol Ethernet con protocolo de red IP (que coinciden con los niveles de filtrado de direcciones IP). En este nivel la estructura cambia, de forma que se realiza una discriminación previa del identificador de nodo para almacenarlo de una forma más ordenada, con el fin de disminuir el tiempo de búsqueda horizontal del nodo en el nivel. En estos cuatro niveles la estructura queda de la forma que indica la Fig. 6.

En estos niveles se realiza una clasificación previa de los nodos, almacenándolos en uno de los 256 grupos, dependiendo del valor numérico del último byte significativo del identificador (se utiliza el último byte por que es el que más varianza sufre). De esta forma, la comprobación de si el nodo está ya creado se realiza entre un número menor de nodos. Dentro de cada subnivel horizontal, el ordenamiento es igual que en el resto de los niveles, y el árbol que se crea bajo este nodo sigue el criterio general anteriormente comentado, sin las cuatro excepciones.

La necesidad de estas cuatro excepciones está en la generación de la matriz de tráfico y el listado de máquinas presentes en la red de forma dinámica. Para realizarlos es necesario mantener información continua del tráfico entre todo par de máquinas, es decir, se posee un filtro definido por cada combinación de máquina origen y destino. Al recorrer el árbol, si esta gran cantidad de información estuviera almacenada de forma “estándar”, el coste de tiempo necesario en la generación dinámica de la matriz y los listados haría que el proceso master dedicase mucho tiempo al análisis de cada paquete. El cambio del algoritmo de almacenamiento de nodos optimiza la velocidad de análisis de cada paquete en estos niveles.

Si se desea borrar una captura, el proceso servidor proporciona al proceso master la misma información que le había suministrado en la creación de dicha captura a través de la Memoria Compartida 2.

5.2.3. Recorrido del árbol de filtrado

El recorrido del árbol se realiza de un modo recursivo “mixto”. Esto quiere decir que dentro de la función de recorrido, que se llama a sí misma para descender por el árbol de filtrado, existen cuatro excepciones, donde el recorrido no se realiza de forma recursiva sino de forma directamente implementada en la misma función.

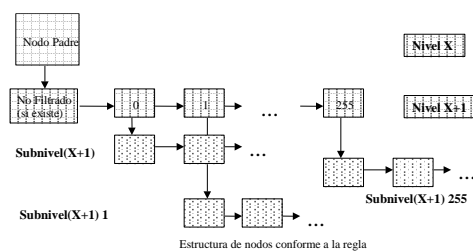


Fig. 6: Estructura de los niveles no estándar

Estas cuatro excepciones coinciden con las cuatro excepciones de la creación del árbol de capturas, y como aquéllas, se han desarrollado para optimizar la velocidad de recorrido del árbol.

El algoritmo de recorrido sobre el árbol Ethernet sigue unas reglas generales:

1. Si la función de filtrado devuelve un valor verdadero, la siguiente acción es seguir el recorrido del árbol por el nodo hijo, si existe. Si no existe, se sigue el recorrido por el puntero a la estructura de parámetro.
2. Como el algoritmo es recursivo, cuando la función a la que ha invocado el nodo en el que se encuentre en ese momento termina su ejecución, se sigue el recorrido del árbol por el puntero a la estructura de parámetro.
3. En cada nivel el paquete cumplirá la función de filtrado de un solo nodo (si se llega a cumplir la función de alguno). El recorrido horizontal se continúa hasta que se encuentre un nodo que cumpla uno de los filtros. Si está definido el nodo de “no filtrado”, que todos paquetes cumplen siempre, es posible que se pueda cumplir la función de otro nodo más en ese mismo nivel. Debido a esta posible circunstancia, se fuerza a que tras la vuelta al nodo de “no filtrado”, se recorra el nivel en el que se encuentre situado de forma horizontal para comprobar si hay o no otro nodo en el que el paquete cumpla la función de filtrado.

Las tres reglas anteriores se pueden comprender observando la Fig. 7, donde la numeración indica el orden de recorrido en el árbol.

La necesidad de creación y mantenimiento dinámico de la matriz de tráfico y el listado de direcciones MAC e IP de la red ha hecho necesario que se puedan generar nodos-filtros de forma automática, es decir, sin necesidad de que un proceso servidor haga una petición de creación de captura. De esta forma, con cada paquete que se captura de la red, se realiza un proceso de comparación de la información de sus campos de dirección con los identificadores de los nodos de los niveles 2, 3, 5 y 6. Como la información contenida en estos niveles (en forma de estructuras de nodo) puede llegar a tener un gran volumen, en términos de memoria, si se sigue el algoritmo “estándar”, que es recursivo, las tareas que debe realizar el sistema operativo cuando realiza los cambios de contexto hace que se pierda una gran cantidad de tiempo y memoria.

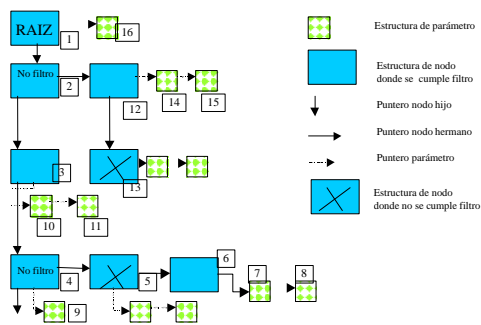


Fig.7: Esquema del recorrido del árbol

Por ello, para el chequeo y actualización de estos niveles se utiliza un algoritmo lineal, cuyo comportamiento y resultados son equivalentes a los que se obtendrían con el algoritmo recursivo, pero sin el malgaste de tiempo y memoria debidos a los cambios de contexto. No se utiliza un algoritmo lineal en todo el recorrido del árbol porque la estructura del filtrado es muy flexible y el algoritmo lineal es muy rígido, y no puede moldearse a todas las posibles combinaciones de filtros que se puedan dar.

6. Rendimiento de la arquitectura

Se han realizado una serie de experiencias para comprobar el rendimiento de la arquitectura descrita en una red de área local Ethernet 10BaseT. La sonda está instalada sobre un PC 133 MHz, 32 MB RAM y 1.5GB de disco duro, con una tarjeta de red 3COM 3c509 ISA. En la red de área local existen 39 máquinas conectadas.

En la Fig. 8 se observa el porcentaje de uso de CPU de la aplicación en función del número de estructuras-nodo que se van creando en el árbol de filtrado para el peor caso, que se ha generado con las condiciones que se explican a continuación. Dado que en la red en la que se toman las medidas no existe una cantidad suficiente de máquinas para crear una gran cantidad de nodos en el árbol, se ha implementado un algoritmo que permite cambiar la dirección MAC de una máquina determinada a una arbitraria. De esta forma se simula la existencia de miles de máquinas en la red.

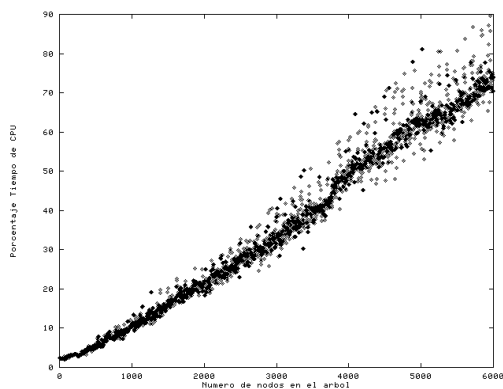


Fig. 8: Porcentaje de uso de CPU en función del número de nodos para el peor caso

Se realiza una secuencia de transmisión a cada uno de estas máquinas “virtuales” de 1000 paquetes a una velocidad media de 3 Mbps. Dado que en el experimento se mantiene la dirección MAC origen constante y lo que se cambia es la dirección MAC destino, en el proceso de actualización de la matriz de tráfico estos nodos de filtrado de MAC no hacen uso del algoritmo de optimización del recorrido del árbol sino que utilizan el estándar.

Por tanto, todos los nodos generados de esta forma aparecen en el mismo subnivel del árbol y su recorrido no está optimizado. La razón de no haber aplicado la optimización es que en el funcionamiento ordinario de una red una máquina se comunica con un número reducido de máquinas, por lo que este efecto no sería apreciable.

De esta forma se observa cómo el porcentaje de uso de CPU se incrementa de forma lineal conforme se eleva el número de nodos en el árbol de filtrado. Esto se debe a que se obliga al nuevo paquete que llega de la red a realizar el recorrido de mayor longitud, con lo que con cada nuevo paquete se incrementa dicha longitud. Este es, por tanto, el peor caso que no puede ocurrir en una red ordinaria. Se observa que para una cantidad de nodos en el árbol cercana a los 6000, se llega a acaparar prácticamente todo el uso de CPU. Cuando se llega a este punto, el tiempo de análisis de cada paquete es superior al tiempo entre llegadas medio de nuevos paquetes, con lo que se comienzan a almacenar estos últimos en la Memoria Compartida I, pudiéndose llegar al punto de que ésta se sature y se comiencen a perder paquetes.

En la Fig. 9 se aprecia el caso que podría corresponder a una red con una carga de tráfico más real. De nuevo, existen miles de máquinas pero ahora el tráfico de una máquina se produce con un número limitado de máquinas destino, en concreto 40, con una carga media de 3 Mbps. Se aprecia como ahora el porcentaje de uso de CPU es asumible, no superando el 10% para 6000 nodos-filtro. Los resultados de la optimización introducida en la estructura del árbol se comprueban con esta experiencia.

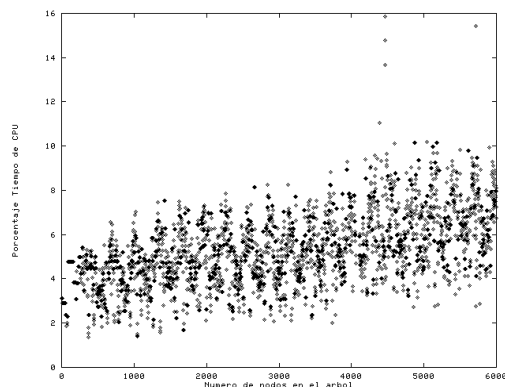


Fig. 9: Porcentaje de uso de CPU en función del número de nodos para una situación de carga de red real

Ante un aumento significativo del número de nodos – filtro en el árbol el incremento de uso de CPU es levemente apreciable (con un aumento de 5000 filtros sólo se aumenta aproximadamente un 3% de uso de CPU).

En la Fig. 10 se representa la evolución del tiempo de ocupación de CPU ante una variación lineal de la carga existente en la red. En este caso, para la generación de la carga de tráfico controlada se han utilizado unos programas cliente-servidor situados en dos máquinas con lo que todos los paquetes de la red proceden de una de estas dos máquinas.

Cuando se ha realizado la medida de esta experiencia, el árbol estaba formado por 1810 nodos. Se observa que el sistema evoluciona de forma aproximadamente lineal ante un aumento de la carga, sin que a altas cargas de la red el coste computacional sea excesivamente alto.

Sin embargo, se comprueba que el coste para una carga de 3 Mbps y aproximadamente 1800 nodos no es la misma entre la experiencia de la Fig. 8 y la Fig. 10: en la Fig. 8 se observa que se utiliza aproximadamente un 20% del tiempo de uso de CPU, mientras que en la Fig. 10 el tiempo de uso es de aproximadamente 6 - 8%. Esta diferencia es producida por la obligación que se tiene en la primera experiencia a realizar el recorrido de longitud máxima en el árbol. En cambio, en la segunda, al tener todos los paquetes con los que se genera el tráfico las mismas direcciones origen y destino, y no estar los nodos correspondientes a éstas en el extremo de los niveles, sino en posiciones intermedias, para analizar estos paquetes no se debe hacer un recorrido por el árbol tan extenso.

En cambio, las experiencias de la Fig. 9 y Fig. 10 representan situaciones similares de tráfico, y por tanto, coinciden los porcentajes de uso de CPU en ambas.

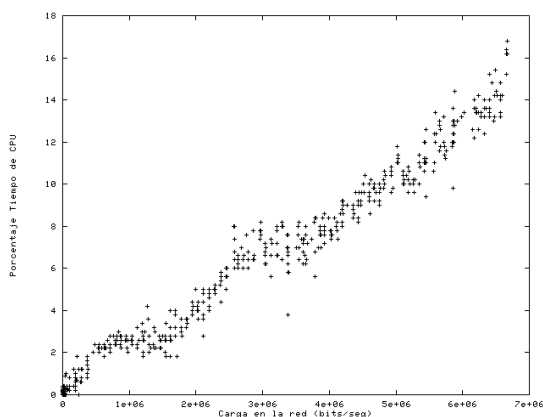


Fig. 10: Porcentaje de uso de CPU en función de la carga de la red

7. Conclusiones

Uno de los elementos fundamentales en el diseño de un sistema de monitorización continua de redes de comunicaciones es la forma en la que se capturan paquetes de la red, se filtran sus protocolos y se analizan. Se ha presentado una alternativa que trata de aprovechar los filtrados a niveles de protocolos comunes con el fin de minimizar el uso de CPU. Este punto de la duplicación de filtros es lo que hace a los *packet filter* poco apropiados para sustentar un sistema de monitorización de tráfico de red en el que se hace necesario llevar miles de filtros simultáneos.

En la arquitectura propuesta se ha optado por una estructura en árbol en la que cada nodo representa una comprobación de cierto campo de determinado protocolo. Si distintos filtrados comparten etapas de filtrado comunes, se comparten los nodos del árbol, evitando la realización de filtrados duplicados como en otros sistemas de *packet filter*. De esta forma se ha conseguido una arquitectura eficiente para el caso de sistemas que realicen miles de filtrados simultáneos, como puede ser el de un sistema de monitorización continua de red.

Para aumentar la flexibilidad de la estructura en árbol se han utilizado algoritmos de creación, borrado y recorrido basados en la recursividad. Sin embargo, para optimizar la velocidad de creación de ciertos filtros muy utilizados como son los de filtrado de máquinas, se ha introducido una estructura rígida, con un recorrido lineal, que agiliza el proceso de mantenimiento de la matriz de tráfico y los listados de direcciones.

Referencias

- [1] Steven McCanne, Van Jacobson, "The BSD Packet Filter: A New Architecture for User-level Packet Capture". Winter USENIX Conference Proceedings, San Diego, CA, Jan.1993.
- [2] Jeffrey C. Mogul, "The Packet Filter: An Efficient Mechanism for User-level Network Code". Proceedings of 11th Symposium on Operating Systems Principles, Austin, TX, Nov.1987, ACM, pp-39-51
- [3] David R. Boggs, Jeffrey C. Mogul, Christopher A. Kent, "Measured Capacity of an Ethernet: Myths and Reality". SIGCOMM'88 proceedings, Stanford, CA, Aug.1998, ACM
- [4] Jeffrey C. Mogul, "Efficient Use of Workstations for Passive Monitoring of Local Area Networks". SIGCOMM'90 proceedings, Philadelphia, PA, Sept.1990, ACM
- [5] Vern Paxson, "Automated Packet Trace Analysis of TCP Implementations", ACM SIGCOMM'97, September 97, Cannes, France.
- [6] "The New Generation of Network Monitoring Systems", ATG's Communications & Networking Technology Guide Series, 1997
- [7] E. Magaña, J. Aracil, J. Villadangos, "PROMIS: A Reliable Real-time Network Management Tool for Wide Area Networks", Proceedings of IEEE Euromicro'98, Vasteras, Suecia, Agosto 1998.

Disciplina de Planificación Basada en Rondas con Disminución de la Distorsión del Patrón de Tráfico

José M^a Quinteiro, Alfredo Sarmiento, Ángelo Santana, Francisco Cabrera

Área Temática: Tecnología Avanzada de Redes
Dpto. de Ingeniería Telemática. Universidad de Las Palmas GC
Campus Universitario de Tafira, Pab. C - 205
35017 Las Palmas GC- Las Palmas
Tel: 928 451236, Fax: 928 451243
E-mail: jquinteiro@cma.ulpgc.es

Abstract

In this paper we address the problem of traffic pattern distortions in packet-switching integrated-services networks. There are mainly two solutions to address this problem: the first one is based on controlling the traffic distortion within the network by using nonwork-conserving service disciplines, such as frame based schemes and round robin schemes; the second one accounts for traffic distortion during scheduling by using work-conserving disciplines where the packets are transmitted according to the deadline-ordered scheduling policy. This mechanism is an evolution of round robin scheduling that take into account the arrival time of a cell without using time-stamping; the most attractive feature is a reduction of local delays and traffic pattern distortions of low priority cells while the service of high priority cells is about the same.

1 Introducción.

La única forma de restaurar el patrón de tráfico en el interior de las redes es empleando esquemas de trabajo no conservativos [9]. Estos esquemas, si bien consiguen controlar la distorsión del patrón de las conexiones, y por tanto son capaces de proporcionar prestaciones de variación del retardo limitado, tienen el inconveniente de que incrementan el retardo medio de los paquetes e infrautilizan la capacidad del canal de salida. Disciplinas de este tipo son **Stop-and-Go** [3] o **HRR** (*Hierarchical Round Robin*) [4] entre otras.

Cuando se utilizan esquemas de trabajo conservativo hay que recurrir a disciplinas basadas en marcas de tiempo ya que éstas tienen la flexibilidad de proporcionar los servicios requeridos. Ejemplos de este tipo ampliamente referenciados en la literatura son **WFQ** (*Weighted Fair Queueing*) [1] [5], **VirtualClock** [7] [2], o **Dinamic Priority Queueing** [6].

Aquí se presenta un novedoso esquema de gestión de *buffers* basado en rondas, de trabajo conservativo y de tasa garantizada que tiene en cuenta el patrón de llegada de dos clases de tráfico. La característica más destacable es su simplicidad, teniendo una complejidad asintótica a efectos de caudal de salida de únicamente $O(1)$. El diseño consta de una cola para cada clase de tráfico: una cola para el tráfico prioritario y otra para el no prioritario, aunque el esquema es generalizable para soportar más tipos de tráficos. La política de servicio garantiza una tasa reservada a cada uno de los dos

tipos de tráfico en función de la relación de servicio utilizada. Para evaluar el comportamiento de esta propuesta se realizará una comparativa con políticas similares del tipo *round-robin* y **HOL** (*Head Of the Line*) las cuales se han adaptado para que también garanticen el caudal de tráfico. Como se demostrará, estos tres esquemas garantizan la tasa si bien a largo plazo proporcionan distintas prestaciones.

Comenzaremos describiendo los mecanismos de colas empleados en el estudio, así como sus algoritmos de funcionamiento. En la sección 2 se presenta el método analítico empleado en la nueva política de cola. También se comprueba que los resultados en estado estacionario obtenidos con el modelo analítico verifican a los proporcionados mediante las simulaciones. En la sección 3 se someten a simulaciones las tres políticas de colas analizadas en distintas condiciones de carga y de relaciones de servicio. Finalmente, en la sección 4 se detallan las conclusiones obtenidas.

1.1.1 Descripción del mecanismo.

Los distintos esquemas de planificación se muestran en las siguientes figuras:

- Planificadores tradicionales con dos clases de tráfico: Esta solución consta de dos colas, una para el tráfico de prioritario (*C1 Clase 1*) y otra para el tráfico no prioritario (*C2 Clase 2*), las cuales son atendidas por un planificador. Existen unos contadores que controlan la cantidad de células que se han servido durante un intervalo. El tráfico *C1* y *C2* tienen unas tasas reservadas de N_{C1} y N_{C2} células,

respectivamente. Cuando el sistema está a plena carga, la tasa garantizada al tráfico TR es $(N - N_{C2})C$, mientras que la tasa garantizada al tráfico C2 es $N_{C2}C$, siendo C el caudal de salida del servidor y $N = N_{C1} + N_{C2}$ el intervalo. El esquema es de trabajo conservativo; de tal forma que cuando un tipo de tráfico ha superado su tasa reservada y no existen células del otro tipo, se le permite el servicio, si no, tendrá prioridad el tipo de tráfico que aún no haya hecho uso de todo el ancho de banda reservado. Para llevar a cabo la tarea de selección se emplean dos técnicas:

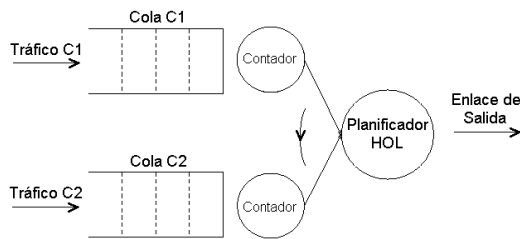


Figura 1 Disciplina basada en HOL.

- La primera sigue una política **HOL**, siendo la cola de mayor prioridad la del tráfico C1, véase la Figura 1. En cada ranura de tiempo, primero se sondea la cola del tráfico C1 y si éste no hace uso del canal, entonces se sondea el tráfico C2. Con esta solución, el caudal recibido a corto plazo por el tráfico C2 es muy sensible a la relación de servicio de C1.

SI ((Existen células de C1) **Y** ((C1 no ha utilizado toda la tasa) **O** (No existen células de C2)))

Transmitir la célula de la cabeza de la cola C1.
Incrementar el contador de la tasa de C1

SI NO

SI (Existen células de C2)
Transmitir la célula de la cabeza de la cola C2.
Incrementar el contador de la tasa de C2.

SI ((C1 ha utilizado toda la tasa) **Y** (C2 ha utilizado toda la tasa))

Restaurar los contadores de la tasa.

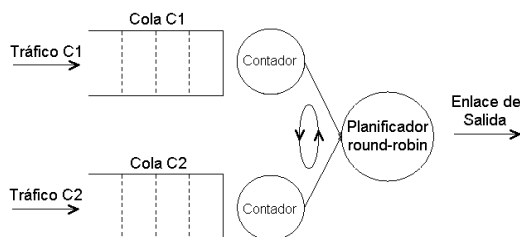


Figura 2 Disciplina basada en RR.

- La segunda solución emplea una política **RR** (*round-robin*), de tal forma que en una ranura

primero se sondea a una determinada cola, mientras que en la siguiente ranura se sondea a la otra cola. El esquema se muestra en la Figura 2.

SI (Ronda de C1)

Proxima ronda de C2.

SI ((Existen células de C1) **Y** ((C1 no utilizado toda la tasa) **O** (No existen células de C2)))

Transmitir la célula de la cabeza de la cola C1.
Incrementar el contador de la tasa de C1.

SI NO

SI (Existen células de C2)
Transmitir la célula de la cabeza de la cola C2.

SI NO

Incrementar el contador de la tasa de C2.
Próxima ronda de C1.

SI NO

Próxima ronda de C1.

SI ((Existen células de C2) **Y** ((C2 no ha utilizado toda la tasa) **O** (No existen células de C1)))

Transmitir la célula de la cabeza de la cola C2.
Incrementar el contador de la tasa de C2.

SI NO

SI (Existen células de C1)
Transmitir la célula de la cabeza de la cola C1.

SI NO

Incrementar el contador de la tasa de C1.
Próxima ronda de C2.

SI ((C1 ha utilizado toda la tasa) **Y** (C2 ha utilizado toda la tasa))

Restaurar los contadores de la tasa.

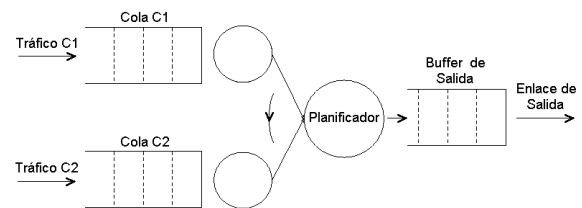


Figura 3 Disciplina BO.

- Planificador **BO** (*Buffer-Output*): Como se muestra en la Figura 3 el planificador consta de dos colas: una para el tráfico C1 y otra para el tráfico de C2, las cuales están conectadas al *buffer* de salida. $N = N_{C1} + N_{C2}$ es el tamaño del *buffer* de salida, siendo N_{C1} y N_{C2} el número máximo de células de cada clase de tráfico que puede haber en el *buffer* de salida. Mientras las dos clases estén activas, el caudal de tráfico garantizado al tráfico C1 y C2 es, respectivamente, $(N - N_{C2})C$ y $N_{C2}C$, siendo C el caudal de salida del servidor. En cada ranura el planificador primero sondea la cola C1 e introduce en el *buffer* de salida todas las células posibles, con un máximo de $N - N_{C2}$ células; seguidamente, sondea la cola C2 e introduce en el *buffer* de salida un máximo de una célula. De esta forma, el tráfico C1 puede ocupar todo el espacio asignado en un único ranura; con lo que obtiene un menor retardo y no se ve afectado por posibles ráfagas del tráfico C2.

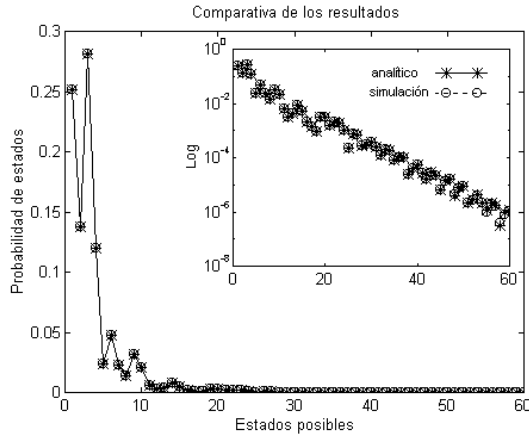


Figura 4 Comparativa de las probabilidades de estados analíticos y de simulación.

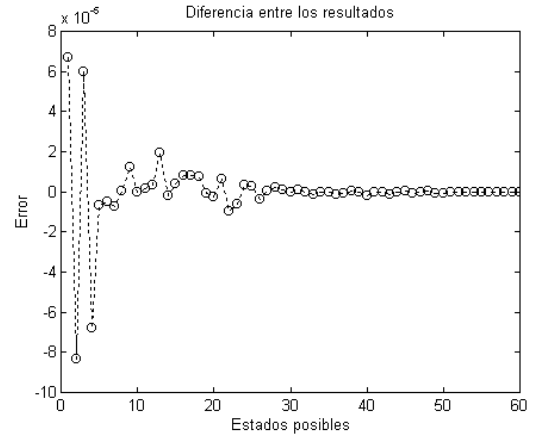


Figura 5 Diferencia entre las probabilidades de estados analíticos y de simulación.

MIENTRAS ((Existen células de C1) **Y** (Queda espacio reservado para C1 en el buffer de salida))

Introducir una célula de C1 en el buffer de salida.
Incrementar el contador de C1.

SI ((Existen células de C2) **Y** (Queda espacio reservado para C2 en el buffer de salida))

Introducir una célula de C2 en el buffer de salida.
Incrementar el contador de C2.

Transmitir la célula de la cabeza de el buffer de salida.
Decrementar el contador del tipo de célula.

2 Validación y resultados numéricos.

En esta sección se compararán los resultados analíticos con los obtenidos mediante simulación. Por simplicidad asumiremos que las llegadas de células de ambas clases de tráfico al planificador se producen de acuerdo con sendas distribuciones geométricas, de parámetro α para el tráfico C1, y β para el tráfico C2. Asimismo asumimos que $N_{C2} = 1$ y que los tamaños de los *buffers* de entrada son, respectivamente, M_1 y M_2 . De esta forma, en cada ranura de tiempo el estado del planificador queda descrito por el vector (N_1, N_2, K) , donde:

N_1 = "Número de células C1 en el planificador"

N_2 = "Número de células C2 en el planificador"

K = "Posición que ocupa la (única) célula C2 en el *buffer* de salida". En el caso de que no haya ninguna célula C2 en este *buffer* se toma $K=0$.

Nuestro objetivo es determinar la distribución en el equilibrio de (N_1, N_2, K) . De la descripción del funcionamiento del planificador, se sigue que la sucesión de sus estados constituye una cadena de Markov. Para obtener la matriz de transiciones de esta cadena, observamos que en cada ranura de tiempo, el planificador opera en dos fases: una primera fase en la que se produce la reubicación de células desde los *buffers* de entrada hacia el *buffer* de salida, y la salida hacia el canal del primer cliente de este *buffer*; y una segunda fase en la que se permite el acceso a los *buffers* individuales del tráfico procedente de cada canal.

En la primera fase las transiciones son completamente deterministas y pueden resumirse del modo siguiente:

$(0,0,0)$	\rightarrow	$(0,0,0)$	
$(N_1,0,0)$	\rightarrow	$(N_1-1,0,0)$	$N_1 > 0$
$(0,N_2,0)$	\rightarrow	$(0,N_2-1,0)$	$N_2 > 0$
$(N_1,N_2,0)$	\rightarrow	(N_1-1,N_2,N_1)	$0 < N_1 \leq N_{C1}$
$(N_1,N_2,0)$	\rightarrow	(N_1-1,N_2,N_{C1})	$N_{C1} < N_1 \leq N_{C1} + M_1$
$(N_1,N_2,1)$	\rightarrow	$(N_1,N_2-1,0)$	$N_1, N_2 > 0$
(N_1,N_2,K)	\rightarrow	$(N_1-1,N_2,K-1)$	$N_2 > 0, N_1 \geq K-1$

En la segunda fase, dado que las llegadas se producen con distribución geométrica, en cada ranura de tiempo puede ocurrir:

- Que llegue una célula de cada clase, con probabilidad $\alpha\beta$
- Que llegue sólo una célula, de clase C1, con probabilidad $\alpha(1-\beta)$
- Que llegue sólo una célula, de clase C2, con probabilidad $(1-\alpha)\beta$
- Que no llegue ninguna célula, con probabilidad $(1-\alpha)(1-\beta)$

De esta forma, en función del estado del planificador al comienzo de la segunda fase, se producirían las siguientes transiciones desde el estado (N_1, N_2, K) :

Estado final	Probabilidad	Estado inicial
(N_1+1, N_2+1, K)	$\alpha\beta$	$N_1 < N_{C1} + M_1, N_2 < N_{C2} + M_2, \forall K$
(N_1+1, N_2, K)	$\alpha(1-\beta)$	$N_1 < N_{C1} + M_1, N_2 < N_{C2} + M_2, \forall K$
(N_1, N_2+1, K)	$(1-\alpha)\beta$	$N_1 < N_{C1} + M_1, N_2 < N_{C2} + M_2, \forall K$
(N_1, N_2, K)	$(1-\alpha)(1-\beta)$	$N_1 < N_{C1} + M_1, N_2 < N_{C2} + M_2, \forall K$
(N_1+1, N_2, K)	α	$N_1 < N_{C1} + M_1, N_2 = N_{C2} + M_2, \forall K$
(N_1, N_2, K)	$1-\alpha$	$N_1 < N_{C1} + M_1, N_2 = N_{C2} + M_2, \forall K$
(N_1, N_2+1, K)	β	$N_1 = N_{C1} + M_1, N_2 < N_{C2} + M_2, \forall K$
(N_1, N_2, K)	$1-\beta$	$N_1 = N_{C1} + M_1, N_2 < N_{C2} + M_2, \forall K$
(N_1, N_2, K)	1	$N_1 = N_{C1} + M_1, N_2 = N_{C2} + M_2, \forall K$

Combinando ambas fases es sencillo construir la matriz de transiciones entre estados. A

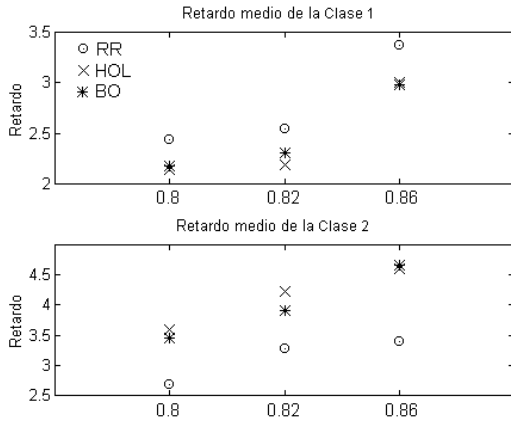


Figura 6 Retardo medio para una relación de servicio 3:1.

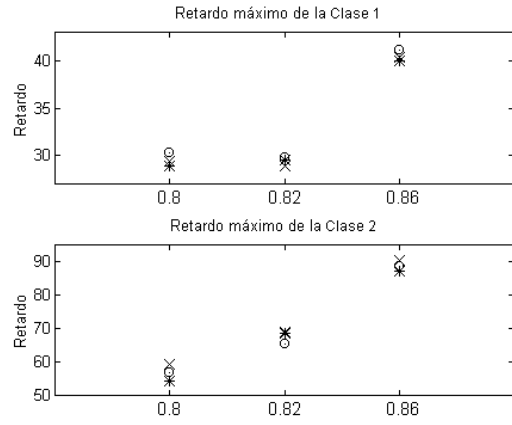


Figura 7 Retardo máximo para una relación de servicio 3:1.

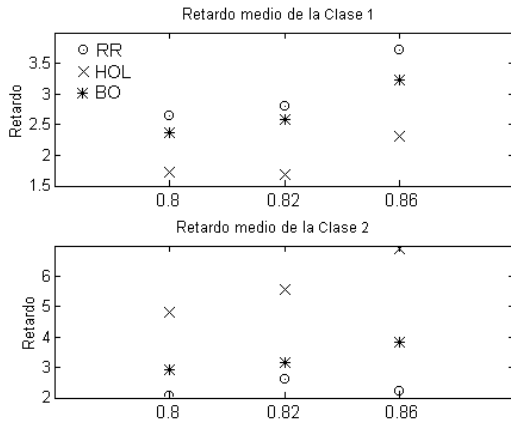


Figura 8 Retardo medio para una relación de servicio 30:10.

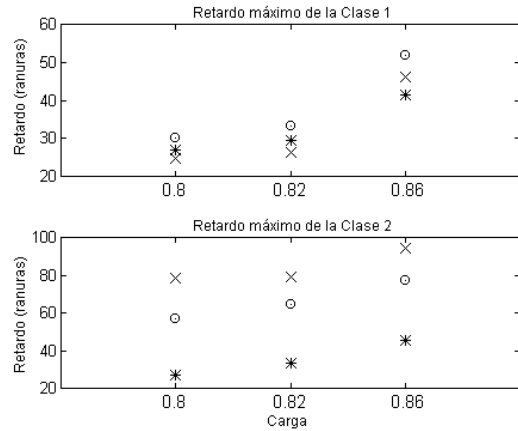


Figura 9 Retardo máximo para una relación de servicio 30:10.

modo de ejemplo, si partimos del estado (1,1,0), tras pasar directamente en la primera fase al estado (0,1,1), en la segunda fase con probabilidad $\alpha\beta$ se transitaría a (1,2,1), con probabilidad $\alpha(1-\beta)$ a (1,1,1), con probabilidad $(1-\alpha)\beta$ a (0,2,1), y con probabilidad $(1-\alpha)(1-\beta)$ a (0,1,1). Sea T esta matriz de transiciones. Si llamamos π al vector de probabilidades en el equilibrio (los elementos de π son las probabilidades $\pi_{(N_1, N_2, K)}$ de que el sistema esté en cada uno de los estados posibles), este vector puede obtenerse resolviendo el sistema:

$$\pi T = \pi$$

$$\sum_{N_1, N_2, K} \pi_{(N_1, N_2, K)} = 1$$

En la Fig. 4 se representan gráficamente las probabilidades en el equilibrio obtenidas analíticamente de esta forma, conjuntamente con la estimación de dichas probabilidades obtenidas simulando dicho sistema en un intervalo de 10^8 ranuras. El tamaño de las colas de la disciplina **OB** es de 4 células para la cola de salida y de 2 células

para las colas C1 y C2; por lo que se obtienen 60 estados posibles.

El ejemplo se ha realizado con dos fuentes de distribución geométrica con probabilidades de llegada 0.5 y 0.25 para el tráfico C1 y C2, respectivamente.

A partir de las probabilidades de estados analíticas se pueden obtener fácilmente otras medidas de interés tales como las probabilidades de pérdidas o los retardos máximos en el multiplexor.

3 Simulaciones.

Las simulaciones se llevaron a cabo con el simulador de tiempo discreto YATS de redes ATM desarrollado por la Dresden University of Technology [8]. Para dicho simulador se diseñaron las clases de objetos de red que implementan los esquemas aquí analizados.

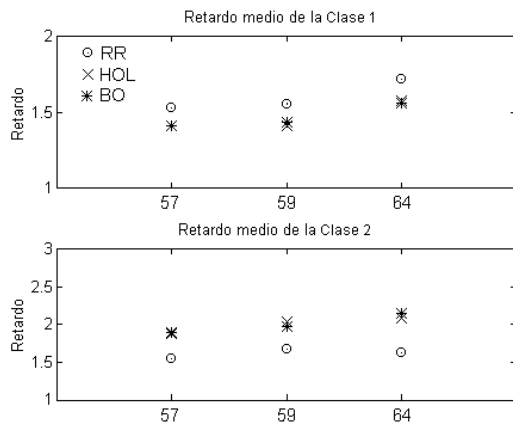


Figura 10 Retardo medio para una relación de servicio 3:1.

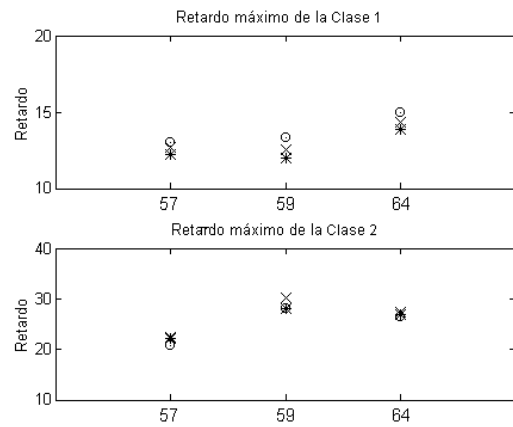


Figura 11 Retardo máximo para una relación de servicio 3:1.

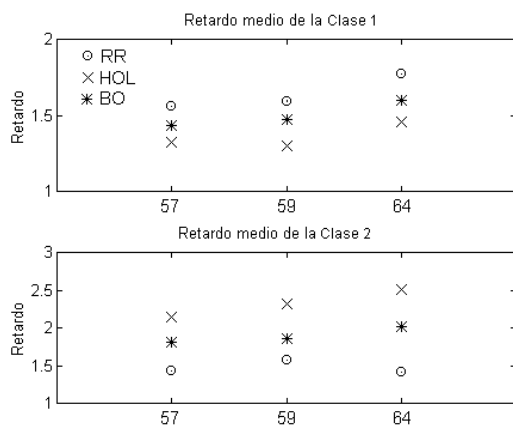


Figura 12 Retardo medio para una relación de servicio 30:10.

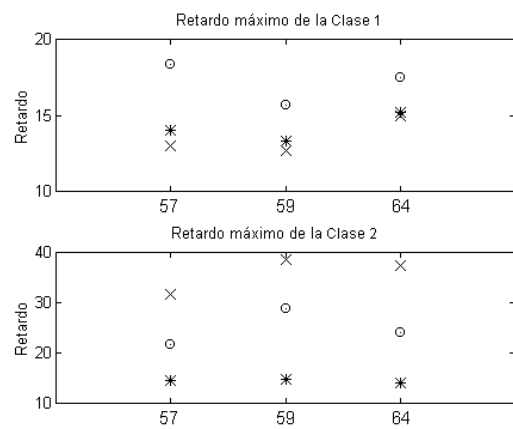


Figura 13 Retardo máximo para una relación de servicio 30:10.

Las simulaciones tienen por objeto comprobar el comportamiento de los esquemas bajo distintas condiciones de carga y relaciones de servicio. Las simulaciones se llevaron a cabo durante 10^8 ranuras. El tamaño de las colas es lo suficientemente grande como para que el sistema sea estable y no se produzcan pérdidas. Para cumplir la tasa garantizada, por ejemplo con una relación 3:1, la relación de servicio de los planificadores sin cola a la salida es de 2 células C1 de cada 3 atendidas; mientras que el planificador **OB** permite un máximo de 2 células C1 y 1 célula C2 en el *buffer* de salida.

3.1 Modelo de tráfico.

Las fuentes utilizadas en las simulaciones son del tipo de tráfico a ráfagas con estados ON-OFF geoméricamente distribuido.

El tráfico de las fuentes tiene un modelo ON-OFF. Durante la fase de silencio no se transmiten células, en cambio, en la fase activa se transmiten células sin separación entre las células. Las duraciones de las fases ON y OFF están distribuidas

geoméricamente con una longitud media B e I , respectivamente. Siendo B la longitud media de la ráfaga de células e I la longitud media de la fase OFF.

Se ha utilizado un modelo similar al empleado por *Chao* en [CHA91] aunque con colas infinitas y cargas de tráfico diferentes. El tráfico está compuesto por seis fuentes ON-OFF de tráfico C1 y otras tantas de tráfico C2. Mientras no se diga lo contrario el número medio de células en el estado ON es la unidad, al igual que la distancia entre las células.

3.2 Comportamiento a alta carga.

La relación del caudal de tráfico generado por ambas clases es de 3:1. Las simulaciones se realizaron con una carga inicial del 80 %; o sea, una carga de 0.6 y 0.2, respectivamente. Para apreciar el comportamiento de los distintos esquemas cuando un tipo de tráfico aumenta el caudal, se analizó el efecto del incremento de la tasa de cada una de las clases de tráfico en un 11.1 %; pasando a generar un

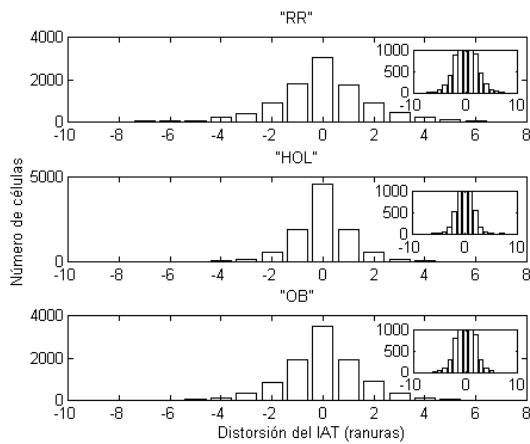


Figura 14 Distribución IAT de la conexión C1.

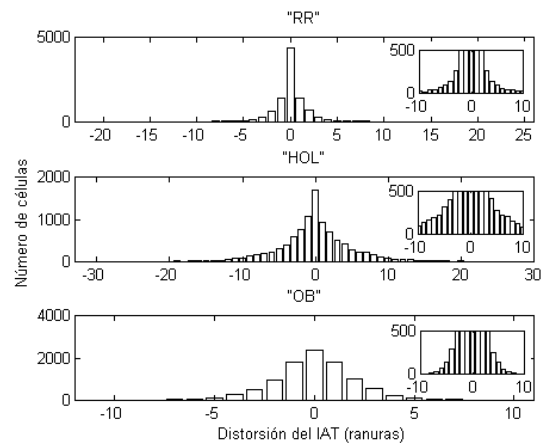


Figura 15 Distribución IAT de la conexión C2.

caudal del 86.6 % cuando el incremento es del tráfico C1 y del 82.2 % cuando el incremento es del tráfico C2.

En las Figuras 6-9 se representan los resultados de las simulaciones para las tres cargas: 0.8, 0.82 y 0.86. Por otra parte, para evaluar la importancia de la relación de servicio se experimentó con las relaciones 3:1 y 30:10. Un previsible primer inconveniente de aumentar la relación de servicio (aunque sea en la misma proporción) es que las dos clases de tráfico tendrán un mayor comportamiento a ráfagas. Se observa que en todos los casos, el retardo medio de la política **OB** es un valor intermedio entre el de **RR** y **HOL**. Además, con la política **OB** se consigue el menor retardo máximo del tráfico C2, mientras que el retardo máximo del tráfico C1 continúa teniendo un valor cercano o inferior al de las otras propuestas. Como cabría esperar, con **RR** se obtiene el menor retardo medio de la clase C2, a expensas de un mayor retardo de la clase C1. En cambio, **HOL** proporciona unas prestaciones opuestas a las de **RR**. Así mismo, cuando aumenta la relación del servicio se consiguen mayores diferencias entre ambas clases de tráfico.

3.3 Comportamiento a media carga.

El tipo de tráfico es el mismo que en el caso anterior, salvo que el caudal agregado de la clase C1 y C2 es 0.42 y 0.14, respectivamente; lo que genera una carga total del 57.1 %. Al igual que en el caso anterior, para apreciar el comportamiento de los distintos esquemas cuando un tipo de tráfico aumenta el caudal, se analizó el efecto del incremento de la tasa de cada una de las clases de tráfico en un 16.6 %; pasando a generar un caudal del 64.2 % cuando el incremento es del tráfico C1 y del 59.5 % cuando el incremento es del tráfico C2.

Se observa en las Figuras 10-13 que con estas cargas la relación de servicio no juega un papel tan

importante como en alta carga; de hecho, el retardo máximo no sufre una gran variación en las tres relaciones de servicio. Además, la diferencia en el servicio proporcionado por las tres propuestas no es tan destacado como en el caso anterior, aunque siguen manteniendo el mismo comportamiento.

3.4 Efecto del tráfico con ráfagas.

Esta simulación presenta las discrepancias en el comportamiento de los tres esquemas cuando el número medio de células en la fase ON del tráfico C1 es de 1, 2 y 3 células, el caudal total es del 80 % y la relación de servicio de 3:1.

Tabla 1. Retardos medios en función de la ráfaga.

B	Clase	RR	HOL	OB
1	C1	2.4347	2.1362	2.1823
	C2	2.6864	3.5822	3.4439
2	C1	3.2963	2.8966	3.0985
	C2	9.6326	10.9652	10.2923
3	C1	10.7940	10.1939	10.3215
	C2	31.3394	33.5888	33.1107

La Tabla 1 muestra que con esta carga y patrón de tráfico los tres esquemas no presentan diferencias sustanciales, si bien se observa como aumenta en una proporción superior el retardo de la clase C2, mientras que no se penaliza fuertemente el tráfico causante de las ráfagas. Además, destaca la importancia de la relación de servicio, ya que al aumentar la duración de la ráfaga los retardos medios aumentan en una proporción 3:1.

3.5 Distorsión del patrón de tráfico.

Por último se evaluará la distorsión en el patrón de tráfico de una conexión de las clases C1 y C2. Las simulaciones se llevaron a cabo durante 10⁶

ranuras con una carga del 80% y una relación de servicio 30:10.

En las Figuras 14 y 15 se representa la distribución de las diferencias de los IAT (*Inter Arrival Time*) de 10^4 células a la salida del multiplexor con los IAT a la entrada. Estas curvas ponen de relieve nuevamente que **OB** ofrece un mejor servicio al tráfico C2. Aunque la disciplina **OB** tiene una menor probabilidad de que la distorsión en el patrón sea nula; tiene la ventaja de que la distorsión global del patrón de tráfico está delimitada por una curva más estrecha, lo que implica que este esquema es capaz de proporcionar una variación del retardo más ajustada.

Tabla 2. Distorsión de las diferencias absolutas del IAT.

Conexión		RR	HOL	OB
C1	Media	1.3364e+0	8.0690e-1	1.0761e+0
	Varianza	1.9062e+0	1.0523e+0	1.2558e+0
	Máxima	11	11	10
C2	Media	1.3634e+0	3.7478e+0	1.5738e+0
	Varianza	4.9268e+0	1.7265e+1	2.2205e+0
	Máxima	26	33	12

4 Conclusiones.

De las simulaciones se deduce la importancia que tiene la relación de servicio en esta propuesta. Aunque con las distintas relaciones de servicio se garantiza el caudal de tráfico, el servicio proporcionado por los diferentes esquemas es muy diferente a corto y largo plazo. Cuando la escala de la relación de servicio es baja, **OB** tiene un comportamiento similar al ofrecido por **HOL**; mientras que cuando aumenta la relación de servicio, el comportamiento de **OB** se aproxima al de **RR** pero con la ventaja de conseguir una disminución en el retardo máximo de la clase C2 cercana al 50% y sin que por ello se vea considerablemente afectado ni los retardos medios de dicha clase ni las prestaciones de la clase C1. Este comportamiento se debe a que el planificador tiene la posibilidad (en función de la carga y de la relación de servicio) de tener en cuenta en cada ranura el patrón de llegada de la clase no prioritaria, siendo el *buffer* de salida el encargado de almacenar el histórico de las llegadas, característica de la que carecen los esquemas *round-robin* tradicionales.

Finalmente se ha comprobado que **OB** genera una menor distorsión límite del patrón de tráfico; con las consiguientes ventajas que esto conlleva: un menor comportamiento a ráfagas y la posibilidad de ofrecer servicios con variación del retardo más estrictos.

Referencias.

- [1] A. Demers, S. Keshav and S. Shenkar, "Analysis and simulation of a fair queueing algorithm", in Proceedings of the ACM SIGCOMM'89, pp. 3-12. Also in J. Internetworking Res. and Experience, pp. 3-26, Oct. 1990.
- [2] Norival R. Figueira and Joseph Pasquale, "An upper bound on delay for the virtualclock service discipline", IEEE/ACM Transactions on Networking, Vol. 3, No. 4, August 1995.
- [3] S. J. Golestani, "A stop-and-go queueing framework for congestion management", in Proceedings of the ACM SIGCOMM'90, Philadelphia, PA, Sept. 1990, pp. 8-18.
- [4] C. R. Kalmanek, H. Kanakia and S. Keshav, "Rate controlled servers for very high-speed networks", in IEEE Global Telecommun. Conf., San Diego, CA, Dec. 1990, pp. 300.3.1-300.3.9.
- [5] Abbay K. Parekh and Robert G. Gallager, "A generalized processor sharing approach to flow control in integrated services networks: the single-node case", in Transactions on Networking, IEEE/ACM JUN'93.
- [6] Jing-Fei Ren, Jon W. Mark and Johnny W. Wong, "A dynamic priority queueing approach to traffic regulation and scheduling in B-ISDN", GLOBECOM '94.
- [7] L. Zhang, "VirtualClock: a new traffic control algorithm for packet switching networks", in Proc. of ACM SIGCOMM'90, Philadelphia, PA, Sept. 1990, pp. 19-29. Also in ACM Trans. Comput. Syst., vol. 9, pp. 101-124, May 1991.
- [8] YATS is available in <http://www.ifn.et.tu-dresden.de/TK/yats/yats.html>.
- [9] Hui Zhang, "Service Disciplines For Guaranteed Performance Service in Packet-Switching Networks", Proceedings of the IEEE, 83(10), Oct 1995.

Capítulo 3B: Sistemas Multimedia

Ubicación de Contenidos Multimedia con Velocidad de Representación Variable en Servidores Multidisco.

A. García Martínez (U. Carlos III de Madrid)

Diseño y evaluación de un sistema para el transporte de vídeo en redes industriales mediante MMS.

V.M. Sempere Payá, J. Mataix Oltra, E. Tordera Santamatilde (U. Politécnica de Valencia)

Particionado Hardware/Software para Sistemas Multimedia.

J. P. Castellano, D. Sanchez y A. Suarez (U. Las Palmas)

Sistema Multicanal de Audio Digital sobre protocolo IEEE-1394.

J. Soler Lucas, F.J. Cortés Santaolalla, M.A. Pérez Costero (Instituto Tecnológico de Aragón)

Desarrollo de un Servidor de Vídeo MPEG sobre ATM nativo y JAVA.

J.M. Fornés y J.A. Gómez Argudo (U. Sevilla)

Visión general de las técnicas de marcas de agua (watermarking) y sus aplicaciones.

J. Unzilla, E. Jacob, A. Ferro (U. País Vasco)

TV Interactiva Terrestre - El proyecto ACTS iTTi.

J. A. Carral, F. Scalise, R. Kopp, L. Lauer, O. Souloumiac (U. Alcalá)

Ubicación de Contenidos Multimedia con Velocidad de Representación Variable en Servidores Multidisco

Alberto García Martínez
Area de Ingeniería Telemática Universidad CARLOS III de Madrid
Avda. de la Universidad 30, 28911 Leganés (MADRID)
Telf: 91-6249180, Fax: 91-6249430
E-mail: alberto@it.uc3m.es

Abstract

Performance in Video-On-Demand servers largely depends on an efficient access to the stored contents. In this article we compare several strategies for distributing multimedia contents among disk array servers. Constant Time Length Declustering, based on the distribution on different disks of fragments that account for the playback of a cycle, has been considered an appropriate data distribution strategy for the service of Variable Bit Rate streams. However, no quantitative results have been presented to support this assert. We have developed a simulator to compare Constant Time Length Declustering with common strategies like data striping, or data interleaving with a fixed striping unit. The results obtained from several experiments prove that Constant Time Length Declustering offers superior performance in most scenarios.

1. Introducción¹

Las aplicaciones que se agrupan bajo la denominación de multimedia se están extendiendo con gran fuerza en terrenos como el ocio, la educación o las telecomunicaciones. De entre todas las aplicaciones multimedia, queremos destacar el futuro prometedor del servicio bajo demanda de contenidos almacenados de forma no volátil. El término Vídeo Bajo Demanda (cuyo acrónimo, VBD, utilizaremos a partir de ahora) es una denominación genérica del servicio de contenidos como vídeos, catálogos multimedia para la compra a distancia, noticias, servicios de información, etc., que han sido generados con anterioridad a su representación, y que se sirven bajo demanda del cliente en un intervalo relativamente corto desde su petición.

En un sistema de VBD uno o varios servidores suministran contenidos multimedia (audio, vídeo, imágenes, texto, etc.) a un gran número de clientes que los solicitan de forma interactiva. Nuestra investigación se centra en el proceso de transferencia de información entre los dispositivos de almacenamiento y la red, que tiene lugar en el servidor. La provisión de un servicio con una calidad determinada al menor coste posible por cliente exige una gestión cuidadosa de los recursos implicados en este proceso. Tomando como punto de partida la especificación del sistema (contenidos a servir, número de clientes que se desea servir, calidad de servicio, etc.), las decisiones técnicas que debemos concretar afectan a:

- La política de planificación de los recursos del servidor (incluyendo cómo se realiza la lectura

de los datos presentes en los sistemas de almacenamiento, cómo se envían los datos por la red, y cómo se gestiona la memoria utilizada en este proceso).

- La política de admisión de clientes.
- La política de ubicación de contenidos en los dispositivos de almacenamiento.

Este último factor tiene especial importancia en servidores diseñados para atender a un gran número de clientes, y que emplean para ello *agrupaciones de discos (disk arrays)*.

Es interesante destacar que las técnicas de compresión comúnmente empleadas para los contenidos multimedia generan bloques de información de tamaños diferentes para la representación de tiempos iguales, con el objetivo de reducir el ancho de banda y la capacidad de almacenamiento. Se dice de estos contenidos que tienen una *velocidad de representación variable*. Un estudio realista de las estrategias de ubicación de contenidos en varios discos deberá considerar la problemática asociada a la existencia de estas variaciones en la velocidad de representación de los contenidos servidos.

En este artículo estudiaremos la conveniencia, en lo que a rendimiento se refiere, del empleo de una técnica de ubicación de contenidos multimedia de reciente propuesta, *distribución entrelazada por duración constante*. Esta técnica, frente a las estrategias clásicas de ubicación de contenidos, presenta características que pueden hacerla más interesante para el servicio de contenidos multimedia con velocidad de representación variable. No obstante, hasta la fecha no se han realizado estudios que avalen esta suposición. En nuestro desarrollo, presentaremos datos experimentales, basándonos en un simulador de un sistema de VBD multidisco, que demuestran que el

¹ Este estudio ha sido financiado parcialmente por el programa CICYT TIC97-0438

rendimiento de distribución entrelazada por duración constante es superior para un vasto conjunto de contenidos a servir.

El artículo está estructurado de la siguiente manera: La segunda sección se detiene en aspectos generales del diseño de la arquitectura software de un servidor de VBD que son de interés para el planteamiento del problema considerado. En el apartado 3 nos centramos en la problemática de la ubicación de contenidos en almacenamiento secundario, dedicando una especial atención a la descripción y al análisis de las técnicas más utilizadas para la distribución de datos en varios discos, y a la descripción de distribución entrelazada por duración constante. A continuación, presentamos el simulador utilizado. La sección 5 describe los experimentos realizados y el análisis de los mismos. Terminamos con las conclusiones.

2. Planificación y control de admisión en servidores de Vídeo Bajo Demanda

La planificación de los recursos disponibles en el servidor y el control de admisión de nuevos clientes son dos procesos fundamentales en la transferencia de datos desde el servidor de VBD al cliente.

2.1. Planificación de los recursos

La opción de enviar antes de su representación toda la información requerida por el cliente resulta poco atractiva, ya que retrasaría el comienzo de la representación, y reduciría la concurrencia en el servidor. Además, se requerirían grandes cantidades de memoria y gran capacidad de almacenamiento en el cliente. Por estos motivos, es preferible el envío frecuente de pequeños fragmentos que permitan mantener la representación durante un corto intervalo de tiempo. A los contenidos servidos de esta forma se les conoce como *flujos*.

La gran mayoría de los servidores de VBD se basan en el paradigma de *envío iniciado por el servidor*: una vez que un cliente solicita un servicio y es aceptado, el servidor es responsable de enviar al cliente la cantidad de datos que este requiera, según una periodicidad establecida.

El uso de algoritmos cíclicos es una estrategia aceptada como conveniente para la planificación del servicio de contenidos multimedia. Un algoritmo cíclico se basa en el establecimiento de un ciclo común para el servicio de todos los clientes. En cada ciclo, el servidor deberá enviar un fragmento de datos con información suficiente para mantener la representación de todos los clientes servidos sin que en ellos aparezcan discontinuidades. Supondremos que el tiempo de ciclo tiene una duración constante.

El proceso de envío de datos a los clientes se realiza en dos fases: La transferencia de datos de disco a memoria, y la transferencia de datos de memoria a la red. Los elementos que limitan la velocidad de

ambas operaciones son la interfaz de red y, sobre todo, los dispositivos de almacenamiento. La planificación de los recursos tendrá como objetivo, consecuentemente, la obtención del máximo rendimiento de los discos.

La eficiencia en la lectura de un disco está determinada por la relación entre tres componentes: el tiempo de búsqueda, el tiempo de lectura y el tiempo de transferencia [17]. El *tiempo de búsqueda* depende del número de pistas recorridas hasta colocar la cabeza lectora sobre la pista de destino. Los valores pueden ir desde 1 ms para un trayecto de una pista, hasta 20 ms si es preciso atravesar todo el disco. Una vez que la pista correspondiente ha sido alcanzada, es necesario esperar a que la rotación del disco haga aparecer el primer sector de la secuencia a leer. Este tiempo, el *tiempo de rotación*, depende de la velocidad de rotación de los discos (son típicas 5400 o 7200 rpm). La velocidad de transferencia varía de acuerdo con la zona del disco considerada, siendo mayor en las zonas más externas, ya que estas contienen un número mayor de sectores para aprovechar la mayor longitud lineal recorrida en la lectura. Si se desea acceder al disco de forma eficaz, conviene incrementar en lo posible la importancia relativa del tiempo de transferencia, frente al tiempo de búsqueda y de rotación.

Los algoritmos cíclicos de planificación son muy apropiados para la planificación del acceso a disco, ya que permiten garantizar el cumplimiento de los plazos de envío de datos a los clientes, con eficiencia, y de forma sencilla que otros algoritmos de planificación como RMT o EDF. Dependiendo del orden establecido en el servicio a los clientes, podemos distinguir dos grandes familias de algoritmos cíclicos para la planificación del acceso a disco:

- *Planificador cíclico de rodaja de tiempo*. El orden en el que se sirven los contenidos no varía de un ciclo a otro.
- *Planificador cíclico SCAN*. Los clientes se sirven en el orden determinado por la posición que ocupan en el disco los fragmentos a leer. El planificador cíclico C-SCAN (*Circular-SCAN*) es una variación en la que sólo se lee del disco cuando el brazo se mueve de la zona más exterior a la más interior.

En comparación con el planificador cíclico de rodaja de tiempo, los planificadores cíclicos de tipo SCAN permiten obtener un mayor rendimiento, ya que el tiempo de búsqueda acumulado a lo largo de un ciclo se reduce. A cambio, los clientes pueden sufrir un retraso en el comienzo de la representación [9].

El tamaño de los fragmentos de datos leídos del disco (o, de forma complementaria, el tiempo del ciclo de servicio) es un parámetro esencial en el rendimiento del servidor de VBD. Cuanto más

grandes sean los fragmentos, más eficiente será el acceso al dispositivo de almacenamiento, ya que el tiempo empleado en búsquedas y rotaciones tendrá una menor importancia relativa. Nótese que para contenidos comprimidos, el tamaño de los fragmentos varía con el ciclo y el cliente considerados.

2.2. Control de admisión

Los sistemas de VBD preservan la calidad de servicio a través de mecanismos de control de admisión. Cada vez que un nuevo cliente desea incorporarse, el servidor comprueba la disponibilidad de recursos como el tiempo de servicio en el disco, el tiempo de servicio en la red, o la cantidad de memoria. Es importante destacar que son el mecanismo de control de admisión, junto con la estrategia de planificación empleada (y no sólo el segundo) los factores que determinan el rendimiento final del sistema.

Para los clientes *intolerantes*, aquellos que no aceptan ninguna discontinuidad en el servicio, el servidor de VBD debe ofrecer garantías deterministas respecto a la calidad de servicio. Podemos identificar dos familias dentro de los algoritmos de control de admisión para clientes intolerantes:

- *Algoritmos de control de admisión basados en caso peor.* La evaluación de recursos se realiza en un escenario de caso peor. Por ejemplo, para el servicio del disco, se emplearán estimaciones de caso peor para el tiempo de búsqueda, rotación y transferencia.
- *Algoritmos de control de admisión mediante simulación.* En esta caso, la decisión se basa en la predicción del comportamiento futuro del sistema [5]. Cuanto más precisa sea la estimación, mayor número de clientes podrá ser admitido. Para obtener una predicción ajustada a la realidad, es imprescindible incorporar un modelo detallado de los dispositivos de almacenamiento presentes en el sistema.

3. Ubicación de contenidos multimedia en dispositivos de almacenamiento

En este apartado vamos a analizar distintas formas de ubicar un contenido en una o varias unidades de almacenamiento, y las consecuencias asociadas. Supondremos que este análisis se realiza para un servidor de VBD de clientes intolerantes que emplea un planificador cíclico. En el primer punto discutiremos la problemática asociada a la ubicación de contenidos en un único disco. A continuación describiremos y analizaremos las estrategias más populares para la distribución de contenidos en varios discos. Finalmente, presentaremos *distribución entrelazada por duración constante*.

3.1. Ubicación de contenidos multimedia en un disco

Existen diferentes modelos de distribución en un disco de la información correspondiente a un contenido, pero los más apropiados son los basados en *ubicación contigua*. En este caso, todos los bloques correspondientes al mismo contenido son almacenados de forma consecutiva. El rendimiento obtenido en la lectura secuencial de contenidos es superior al de ubicación aleatoria, pero sufre de fragmentación externa. Para solucionar este problema, se puede dividir el contenido en fragmentos de datos de tamaño suficiente para mantener la representación durante un ciclo; los bloques correspondientes a cada uno de estos fragmentos se almacenarán de forma contigua. A esta estrategia la llamamos *ubicación localmente contigua*, y, con un planificador cíclico, permite obtener un rendimiento equivalente al que ofrece ubicación contigua.

3.2. Estrategias clásicas de ubicación de contenidos en varios discos

Cuando los servidores disponen de varios discos, es necesario encontrar estrategias para la distribución de los contenidos que permitan un acceso eficiente. Los factores que tienen una mayor influencia en el rendimiento de un servidor multidisco son:

- *La eficiencia en el uso de cada disco.* Es importante reducir la importancia relativa de los tiempos de búsqueda y de rotación para incrementar el tiempo dedicado a la transferencia de datos. Conviene tener en cuenta que contenidos con velocidades de representación reducidas pueden ser causantes de una disminución en el rendimiento en la lectura de datos.
- *La equidad en la distribución de la carga entre todos los discos.* Encontramos dos posibles inconvenientes para la consecución de este objetivo:

Por un lado, una alta variabilidad en la cantidad de datos requerida por cliente y por ciclo puede dar lugar a repartos poco equitativos de carga entre los discos en un ciclo dado (y variaciones en la carga de un disco en ciclos distintos).

Por otro lado, es posible que algunos discos almacenen una parte mayor del contenido que otros. Al dividir un contenido en fragmentos que se distribuyen entre varios discos, es posible que algunos discos cuenten con más fragmentos que los otros (si la distribución trata de ser equilibrada, la diferencia máxima será de un fragmento). Los discos que contengan una cantidad mayor de datos también tendrán una carga mayor.

Los dos factores arriba mencionados, eficiencia en el uso de cada disco y equidad en el reparto de la carga, vienen determinados principalmente por

- La estrategia de distribución de datos.
- El algoritmo de planificación: tiempo de ciclo, organización de los accesos, etc.
- Las características de los contenidos servidos: velocidad de representación, patrón de variación del tamaño de las peticiones, etc.

Una vez presentados los factores que determinan el rendimiento, podemos describir las estrategias más populares para la distribución de contenidos en varios discos [9]:

- *Distribución independiente.* Se utilizan discos separados, cada uno con su propio controlador. Cada contenido se almacena por entero en un único disco. Esta configuración permite aumentar la capacidad del sistema y el número de peticiones servidas concurrentemente, al compararla con el uso de un único disco, pero no el ancho de banda para servir una petición.
- *Distribución entrelazada.* En la distribución entrelazada (*interleaving*), un contenido es dividido en fragmentos que son almacenados en discos distintos. Cada disco operará independientemente y podrá contar con su propio controlador. Para esta configuración, podemos definir la *unidad de distribución (striping unit)*, [15], como la cantidad de datos lógicamente contiguos que se almacenan de forma físicamente contigua en un único disco. En una agrupación de discos con distribución entrelazada, la unidad de distribución, múltiplo del número de bytes por sector, determina cómo se descompone una petición lógica en peticiones que afectan a discos distintos.
- *Distribución uniforme* - En la distribución uniforme (*striping*), un contenido se reparte entre todos los discos, a razón de un bit o un byte en cada uno. Su comportamiento es equivalente a un único disco con una densidad de información incrementada en un factor igual al número de discos considerados. Los discos, idénticos, deben estar sincronizados. La alta coordinación exigida se suele conseguir gestionando el conjunto a través de un único controlador hardware. Para el análisis del rendimiento, esta estrategia se puede considerar como una distribución entrelazada con una unidad de distribución con tamaño igual a un bit o byte (aunque el tamaño de las peticiones deba ser múltiplo de un sector).

El rendimiento de estas estrategias lo analizaremos de forma cualitativa estableciendo una clasificación en grupos de comportamiento similar, a los que aplicaremos los criterios presentados al comienzo

del apartado. Los grupos se establecen en función de la relación entre el tamaño del contenido, el tamaño de la petición correspondiente a un ciclo, y el tamaño de la unidad de distribución.

- *Distribución entrelazada con unidades de distribución similares o superiores al tamaño de los contenidos.* En este grupo encontramos a la distribución independiente. El rendimiento obtenido cuando la popularidad en la elección varía de unos contenidos a otros, una situación muy frecuente, es muy inferior al que se podría esperar de la suma del ancho de banda de los discos implicados. Nótese que el número de clientes que pueden estar accediendo a la parte del contenido residente en un disco dado está limitado por el ancho de banda de un disco.

- *Distribución entrelazada con unidades de distribución del mismo orden o mayores que las peticiones, pero menores que el tamaño de los contenidos.* Los contenidos están distribuidos entre varios discos (posiblemente todos), aunque una petición concreta será servida por un disco o por un número reducido de discos. El ancho de banda de servicio de una petición es bajo, pero pueden servirse concurrentemente varias peticiones distintas. Aunque en un ciclo dado un cliente no pueda ser aceptado porque en un disco no hay ancho de banda para atender más peticiones, si algún disco de la agrupación dispone de ancho de banda, el cliente podrá ser atendido en un ciclo posterior.

La utilización de los discos será alta, ya que el número de movimientos de sus brazos es bajo. Pero en el reparto de la carga pueden surgir dificultades. Por un lado, algunos discos dispondrán de una mayor cantidad de contenidos que otros. Por el otro, el servicio de contenidos con velocidades de representación variable hace que para algunos discos, y en algunos ciclos, la carga sea muy elevada [18].

- *Distribución entrelazada con unidades de distribución de menor tamaño que las peticiones,* como por ejemplo distribución uniforme. En este caso una operación implicará la colaboración de muchos discos, o quizás a todos. En un instante dado, un alto número de discos atenderá una petición, y habrá un bajo número de peticiones distintas siendo atendidas concurrentemente.

Al compararlo con el caso anterior, vemos que cada disco realiza un número mayor de operaciones distintas, y por ello el tiempo dedicado a búsqueda y rotación aumenta. Para conseguir igual ancho de banda efectivo (cantidad de datos leídos por unidad de tiempo), es necesario incrementar el tiempo de ciclo, y consecuentemente la latencia en el comienzo del servicio del cliente. Pero la distribución de la carga es mejor que en caso anterior: Por un lado, la diferencia relativa entre la cantidad de

datos correspondiente a un mismo contenido que almacenan discos distintos es muy pequeña, porque lo es la unidad de distribución. Y por otro, en un mismo ciclo todos discos tienen cargas similares entre sí, ya que atienden de forma conjunta a las mismas peticiones.

Para un análisis cuantitativo del rendimiento que se puede obtener con las estrategias presentadas, es necesario, como ya hemos razonado, tener en cuenta los contenidos concretos a servir, la estrategia de planificación empleada, etc.

Un gran número de trabajos de investigación se han dedicado a la caracterización de las estrategias comentadas arriba, y al análisis de otros parámetros no considerados en esta escueta presentación: El número de discos entre los que se distribuye un contenido, la forma de repartir los contenidos entre distintos discos, la variación en la unidad de distribución para diferentes contenidos, etc. Destacamos el trabajo de [2], [10], [12], [13], [14] y [4]. La principal objeción que se puede achacar a los estudios mencionados reside en su limitación al sencillo caso en el que los contenidos tienen una velocidad de representación constante. Desarrollos que contemplan el servicio de flujos con velocidad variable de representación los encontramos en [7] y [18].

3.3. Distribución entrelazada por duración constante

Hasta este punto se ha supuesto implícitamente que la unidad de distribución debe tener un tamaño constante. Pero puede resultar de interés para el servicio de contenidos con velocidad de representación variable el acomodo secuencial en un único disco de la cantidad de datos necesaria para mantener la representación durante un intervalo constante de tiempo ([6], [1], [3], [16]).

En particular, con un planificador cíclico, puede resultar beneficioso el almacenamiento de forma contigua de la información necesaria para mantener la representación durante un ciclo. A esta estrategia de distribución la llamaremos *distribución entrelazada por duración constante* [8], y la referimos como Entrelazada-DC. Con este proceder los costes en tiempo incurridos por cliente y ciclo son exclusivamente los asociados a un movimiento de la cabeza de lectura de un solo disco. Por muy grande que sea la unidad de distribución, no se puede asegurar lo mismo para una distribución entrelazada por cantidad de datos constante. Este argumento augura un uso más eficaz del disco, uno de los factores que determinan el rendimiento; pero

no hay ninguna indicación de que la distribución de la carga resulte igualmente satisfactoria.

Cohen, Burkhard y Rangan [6] estudian la aplicación de Entrelazada-DC al servicio de contenidos de velocidad de representación variable, razonando a partir de consideraciones estadísticas sencillas que el reparto de la cantidad media de datos en cada disco tiende a ser similar. No obstante, no ofrecen ni demostración rigurosa, ni datos experimentales que avalen la afirmación. Además, su razonamiento es poco riguroso si se considera el servicio de clientes intolerantes, ya que las valoraciones estadísticas pierden peso en favor del análisis del caso peor que puede aflorar en un disco y ciclo dados. En el resto de los trabajos que estudian Entrelazada-DC, [1], [3], [16], no se realiza mención alguna al rendimiento obtenido.

En las circunstancias consideradas (flujos de velocidad variable y clientes intolerantes), el estudio analítico se topa con obstáculos difícilmente superables, dejando a la vía experimental como la más apropiada para valorar la bondad de Entrelazada-DC. Hasta lo que nosotros conocemos, la única evaluación cuantitativa de esta estrategia ha sido presentada en [8]; esta evaluación la completaremos en los siguientes apartados.

4. Descripción del simulador

Para la evaluación cuantitativa, hemos construido un simulador de un servidor de VBD multidisco. Cada disco representa un SEAGATE ST31200W [11], incluyendo el movimiento de la cabeza del disco, la rotación y la velocidad de transferencia, variable con la zona considerada.

Para los sistemas con distribución entrelazada, se ha supuesto que al principio de cada ciclo el gestor de disco divide las peticiones para cada cliente en subpeticiones que son atendidas de forma independiente por cada disco. La reordenación de los datos se producirá en el cliente [16], por lo que no se requiere ningún tipo de sincronización entre los dispositivos de almacenamiento.

El simulador utiliza información sobre el tamaño de los cuadros de películas reales codificadas según MPEG-1. Se han empleado cinco secuencias: StarWars [Garrett y Willinger, 94], un vídeo monocromo con una duración ligeramente superior a dos horas; y Asterix, MTV, Simpsons y Video [Rose, 95] vídeos en color, codificados con 12 bits por pixel, con una duración de unos 28 minutos los tres primeros, y 3 minutos y medio el último. También se permite el servicio de contenidos con velocidad de representación constante, definidos por su velocidad de representación y su duración.

Los contenidos se colocan de forma localmente contigua, con una separación de tamaño constante entre contenido y contenido. De esta forma, el primer contenido se sitúa en los sectores iniciales, y el último en los sectores finales.

Suponemos un planificador cíclico de tipo C-SCAN, con una duración del ciclo de 10 segundos. Para los experimentos mostrados, las pruebas de admisión se realizan con un mecanismo de control mediante simulación (aunque el simulador también permite seleccionar un mecanismo de control de admisión de caso peor).

Los valores utilizados para la variable aleatoria que modela la llegada de clientes (construida a partir de una distribución de Poisson más una constante) garantizan que el sistema se encuentra en saturación. Para la elección de los contenidos por parte de los clientes se emplea una variable aleatoria que sigue una distribución de Zipf con una desviación de 0,271, utilizada corrientemente en la definición de patrones de acceso a contenidos multimedia [19]. Los contenidos que están más cercanos a al exterior del disco (con mayores velocidades de transferencia) se suponen más populares.

El parámetro principal para la evaluación del rendimiento es la *velocidad de transferencia efectiva*, que definimos como el número total de bytes transferidos por segundo. Otras medidas popularmente empleadas, como el número de clientes servidos, no resultan apropiadas para la evaluación del servicio de contenidos heterogéneos.

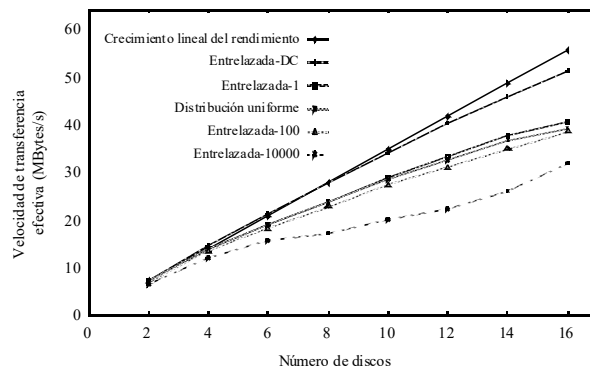
5. Resultados experimentales y análisis

Para hacer más legible el desarrollo, emplearemos la notación *Entrelazada-t* para referirnos a la distribución entrelazada por cantidad constante de datos con una unidad de distribución de *t* sectores (por ejemplo, Entrelazada-10000).

En primer lugar, estudiamos el rendimiento que se obtiene en el servicio de las cinco películas con una velocidad de representación variable referidas anteriormente, según distintas estrategias de distribución de los datos. Los resultados, para un número variable de discos pueden observarse en la figura 1. Es interesante desglosar la ocupación de disco para cada una de las estrategias de distribución de datos estudiadas (figura 2).

Entrelazada-10000, con gran tamaño de la unidad de distribución en comparación con los datos, presenta un rendimiento muy bajo debido al inadecuado reparto de los contenidos entre los distintos discos.

Tanto distribución uniforme, como Entrelazada-1, ambas con una unidad de distribución pequeña frente al tamaño de la petición, se ven fuertemente



penalizadas por el elevado porcentaje de tiempo dedicado a búsquedas y rotaciones.

Fig 1.- Velocidad de transferencia efectiva en función del número de discos para distintas estrategias de distribución de los datos

Entrelazada-100 ofrece un rendimiento intermedio entre Entrelazada-10000 y Entrelazada-1.

Entrelazada-DC destaca por el excelente rendimiento obtenido en las condiciones del experimento. En la figura 2 se observa que el tiempo dedicado a búsquedas y rotaciones es muy escaso, como resulta deseable. La diferencia con la extrapolación lineal del rendimiento de un disco se debe a que Entrelazada-DC no garantiza el reparto de los contenidos de forma completamente equitativa entre todos los discos, tal y como se puede observar en la figura 3. A pesar de ello, el rendimiento global es muy satisfactorio.

El experimento anterior parece indicar que Entrelazada-DC es la estrategia más interesante para la ubicación de los contenidos en un sistema basado en múltiples discos de todas las presentadas. No obstante, esta observación debe ser contextualizada, ya que algunas de las justificaciones que hemos aportado para explicar el comportamiento observado son muy dependientes del conjunto de contenidos servidos y de la distribución de las peticiones que generan los clientes. Por ello, creemos conveniente

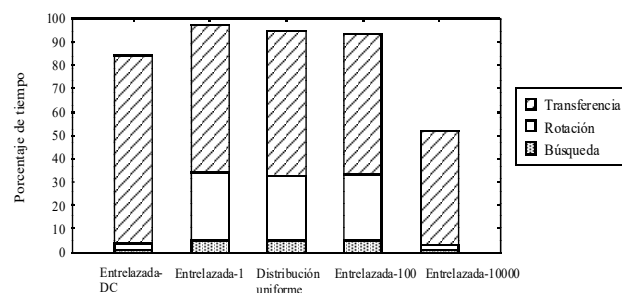


Fig 2.- Porcentajes medios de los tiempos de búsqueda, rotación y transferencia, para distintas estrategias de distribución de datos en un servidor compuesto por 16 discos

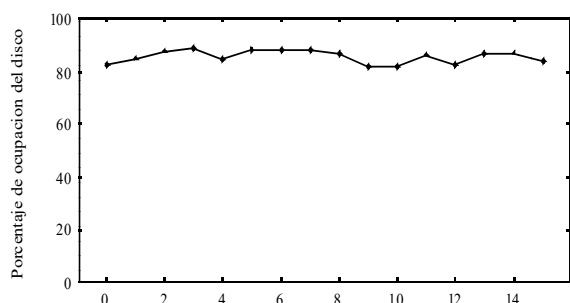


Fig. 3.- Porcentaje de uso para cada uno de los discos de un servidor compuesto por 16 discos con distribución Entrelazada-DC

comparar el comportamiento de las estrategias más relevantes con otros conjuntos de contenidos.

Para llevar a cabo este experimento, definimos los siguientes conjuntos de contenidos:

- Conjunto A- Las cinco películas ya presentadas
- Conjunto B- Las cinco películas sin StarWars
- Conjunto C- Sólo StarWars
- Conjunto D- Flujos con una velocidad de representación constante de 0,5 Mbits/s y una hora de duración
- Conjunto E- Conjunto formado por flujos muy heterogéneos: las cinco películas MPEG-1; flujos de audio a 64 Kbits/s con duración entre dos horas y 30 minutos; y flujos de texto de muy baja velocidad de representación (800 bits por minuto).

El experimento se realizará comparando las estrategias de distribución de datos representan las distintas opciones encontradas: Entrelazada-DC, Entrelazada-1 (cuyos resultados supondremos representativos de la distribución uniforme) y Entrelazada-10000. El servidor está compuesto por 16 discos. Los resultados obtenidos se observan en

Tabla 1. - Velocidad de transferencia efectiva para un sistema de 16 discos que implementa doble zona de memoria con distintos conjuntos de contenidos

Conjunto	Entrelazada -DC	Entrelazada -1	Entrelazada-10000
A	51,1	40,4	31,6
B (sin StarWars)	49,3	42,2	33,0
C (sólo StarWars)	58,6	43,3	42,4
D (flujos con velocidad de representación constante)	58,6	43,3	42,4
E (contenidos muy heterogéneos)	36,1	20,9	10,4

la tabla 1.

En general, Entrelazada-DC ofrece un rendimiento superior al obtenido con el resto de las técnicas. Y hacemos notar su buen comportamiento para el servicio de contenidos heterogéneos. La única excepción a la primacía en el rendimiento de Entrelazada-DC la encontramos en el servicio del conjunto C, para el que, casi sorprendentemente, Entrelazada-10000 resulta ser ligeramente mejor (en un 2,5%). En este caso, el contenido a distribuir (la película StarWars) tiene un gran tamaño, y la diferencia en la cantidad de datos asignada a los distintos discos es pequeña en valor relativo. Así, Entrelazada-10000 da lugar a una distribución de contenidos en los discos algo más equitativa que Entrelazada-DC. A la vista de este resultado, podemos afirmar que el uso de Entrelazada-t con unidades de distribución de gran tamaño puede ser interesante si los contenidos son grandes en relación con el producto de la unidad de distribución y el número de discos empleados.

Para todos los conjuntos considerados, el uso de unidades de distribución de pequeño tamaño ofrece un rendimiento inferior en comparación con Entrelazada-DC, debido al tiempo empleado en búsquedas y rotaciones. Para el conjunto E, con un número de peticiones por ciclo muy grande, al ser estas de pequeño tamaño, el tiempo medio dedicado a la rotación alcanza el 44,5%. No obstante, Entrelazada-1 resulta superior a Entrelazada-10000 cuando los contenidos no son muy grandes frente a la unidad de distribución y el número de discos, y la influencia de un inadecuado reparto de la carga se hace significativa.

6. Conclusiones

En este artículo se ha estudiado la influencia en el rendimiento de las estrategias de ubicación de contenidos multimedia en almacenamiento secundario. Se han presentado y discutido las estrategias que se utilizan corrientemente para la distribución de contenidos en agrupaciones de discos: distribución uniforme y entrelazada (con unidad de distribución de tamaño fijo). El parámetro primordial en la definición de estas estrategias es la unidad de distribución, que indica la cantidad de datos que reside de forma contigua en un disco. Nuestro análisis ha identificado en la utilización del disco y la distribución de la carga a los factores principales para comprender la variación del rendimiento en la lectura de datos. También se ha presentado la estrategia de distribución entrelazada por duración constante, propuesta recientemente, en la que la información correspondiente a la representación de un ciclo para un cliente reside en un único disco.

Los resultados obtenidos mediante simulación nos han permitido, por vez primera, comparar de forma cuantitativa el rendimiento de las estrategias arriba

mencionadas, para el servicio de clientes intolerantes que requieren contenidos con velocidad de representación variable. Nuestros resultados indican que la estrategia de distribución entrelazada por duración constante resulta ser, en la gran mayoría de los casos, la forma más conveniente de repartir los contenidos entre varios discos. Esta técnica ofrece interesantes ventajas respecto a otras opciones de uso más frecuente, como puede ser distribución uniforme (por lo general, ineficaz dado el elevado número de movimientos de los brazos del disco que implica) o distribución entrelazada por cantidad de datos constante (muy similar a distribución uniforme si el tamaño de la unidad de distribución es pequeño en comparación con la velocidad de representación del contenido; o con problemas de reparto de la carga si la unidad de distribución es de gran tamaño). No obstante, es necesario estudiar cuidadosamente el conjunto de contenidos servidos, para utilizar en cada momento la política de distribución más apropiada. En nuestro estudio presentamos algunos resultados que pueden ser de interés a la hora de escoger entre distribución uniforme, entrelazada por cantidad de datos constante, o entrelazada por duración constante.

Referencias

- [1] C. Bernhardt, E. Biersack. "The Server Array: A Scalable Video Server Architecture". Capítulo 5 en *High-Speed Networking for Multimedia Applications*. O. Spaniol, W. Effelsberg, A. Danthine, D. Ferrari. Editorial Kluwer. Marzo 1996.
- [2] S. Berson, S. Ghandeharizadeh, R. Muntz, X. Ju. "Staggered Striping in Multimedia Information Systems". Actas de ACM International Conference on Management of Data (SIGMOD), Minneapolis, EEUU, págs. 79-90. Mayo 1994.
- [3] M. M. Buddhikot, G. M. Parulkar. "Efficient Data Layout, Scheduling and Playout Control in MARS". *ACM Multimedia Systems*, 5(5), págs. 199-212. 1997.
- [4] E. Chang. "Storage and Retrieval of Compressed Video". Tesis, Universidad of California at Berkeley. 1996.
- [5] E. Chang, A. Zakhor. "Cost Analyses for VBR Video Servers". *IEEE Multimedia*, Vol. 3, n° 4, págs. 56-71. Invierno 1996.
- [6] A. Cohen, W. A. Burkhard, P. V. Rangan. "Pipelined Disk Arrays for Digital Movie Retrieval". Actas de International Conference on Multimedia Computing Systems (ICMCS), págs. 312-317. Tysons Corner, EEUU. Mayo 1995.
- [7] C. S. Freedman, D. J. DeWitt. "The SPIFFI Scalable Video-on-Demand System". Actas de ACM International Conference on Management of Data (SIGMOD), San Jose, EEUU, págs. 352-363. 1995.
- [8] A. García-Martínez, J. F. Conde, Á. Viña. "Efficient Memory Management in VOD Disk Array Servers Using Per-Storage-Device Buffering". Actas de IEEE Euromicro Workshop on Multimedia and Telecommunications, Euromicro '98, págs 551-558. Västerås, Suecia. Agosto 1998.
- [9] J. Gemmell, H. M. Vin, D. D. Kandlur, P. V. Rangan. "Multimedia Storage Servers: A Tutorial and Survey". *IEEE Computer*, 28(5), págs. 40-49. Mayo 1995.
- [10] S. Ghandeharizadeh, S. H. Kim. "Striping in Multi-Disk Video Servers". Actas de SPIE High-Density Data Recording and Retrieval Technologies Conference. Octubre 1996.
- [11] S. Ghandeharizadeh, J. Stone, R. Zimmermann. "Techniques to Quantify SCSI-2 Disk Subsystem Specifications for Multimedia". Technical Report TR 95-610. University of Southern California. 1995.
- [12] S. Ghandeharizadeh et al. "Mitra: A Scalable Continuous Media Server". *Multimedia Tools and Applications Journal*, 5(1), págs. 79-108. Ed. Kluwer Academic Publishers. Julio 1997.
- [13] J. Hsieh et al. "Performance of a Mass Storage System for Video-On-Demand". *Journal of Parallel and Distributed Computing*, 30(2), págs. 147-167. Noviembre 1995.
- [14] D. Jadav, A. Choudhary, P. B. Berra. "An Evaluation of Design Trade-Offs in a High-Performance Media-on-Demand Server". *ACM Multimedia Systems* 5(1), págs. 53-68. Ed. Springer Verlag. Enero 1997.
- [15] K. Keeton, R. H. Katz. "Evaluating Video Layout Strategies for a High-Performance Storage Server". *ACM Multimedia Systems*, 3(2), págs. 43-52. Ed. Springer Verlag. Mayo 1995.
- [16] J. Y. B. Lee. "Parallel Video Servers: A Tutorial". *IEEE Multimedia*, 5(2), págs. 20-28. Abril/Junio 1998.
- [17] C. Ruemmler, J. Wilkes. "An Introduction to Disk Drive Modeling". *IEEE Computer*, 27(3), págs. 47 -57. Marzo 1994.
- [18] P. J. Shenoy, H. M. Vin. "Efficient Striping Techniques for Multimedia File Servers". Actas de International Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV), págs. 25-36. Mayo 1997.
- [19] W. Tetzlaff, R. Flynn. "Block Allocation in Video Servers for Availability and Throughput". Actas de SPIE Multimedia Computing and Networking (MMCN), San José, EEUU. Enero 1996.

Diseño y evaluación de un sistema para el transporte de vídeo en redes industriales mediante MMS

Víctor M. Sempere Payá, Universitat Politècnica de València
Plaza Ferrándiz y Carbonell s/n - 03801 Alcoi (VALENCIA)
Tel: 96 554 62 99 / Fax: 96 652 84 09 / e-mail: vsempere@dcom.upv.es

Jorge Mataix Oltra, Universitat Politècnica de València
Camino de Vera s/n - 46020 València (VALENCIA)
Tel: 96 387 71 90 / Fax: 96 387 73 09 / e-mail: jmataix@eupbl.upc.es

Enric Tordera Santamatilde, Universitat Politècnica de València
C/ Polo y Peyrolón, 30 p.11 - 46021 València (VALENCIA)
Tel: 96 360 39 14 / e-mail: etordera@abonados.cplus.es

Abstract

This paper introduces a new video transmission system designed to work over the MAP (Manufacturing Automation Protocol) stack using the MMS (Manufacturing Message Specification) protocol. The transmission of images introduces powerful remote monitoring capabilities into industrial environments, while utilization of the versatile MMS protocol gives the system great flexibility and tight integration into the factory networks. Main characteristics of the system and the devised protocols are described. Then, a first implementation developed by the authors is presented, demonstrating the viability of the proposal. A preliminary study on the impact of video transmission over the network traffic is also included.

1. Introducción

La creciente complejidad de los procesos industriales exige una organización minuciosa de todos los elementos que los componen. Múltiples dispositivos deben realizar sus funciones de forma coordinada, compartiendo información que muchas veces se encuentra fuertemente distribuida. En este escenario cobra importancia el esquema CIM (Computer Integrated Manufacturing): la integración de las distintas funciones de producción mediante un sistema distribuido de procesamiento de información. Dicho sistema distribuido se materializa en una estructura telemática que permite el flujo de información entre todos los elementos involucrados en la producción. MAP (Manufacturing Automation Protocol) es una de las arquitecturas de comunicaciones aceptada por las grandes empresas como estándar para sus redes industriales. Y, sobre MAP, trabaja MMS (Manufacturing Message Specification) como protocolo de intercambio de mensajes entre dispositivos más utilizado.

Los procesos industriales necesitan de medios para monitorizar y supervisar el correcto funcionamiento de cada una de sus fases. La observación visual es una forma intuitiva, cómoda y directa de supervisión. Resulta interesante, pues, disponer de un mecanismo de monitorización que permita ver en cada instante cómo está funcionando cada subsistema. Ahora bien, debe ser un mecanismo integrado en la estructura telemática existente para no romper el esquema CIM. En este artículo se propone un sistema de transmisión de vídeo sobre

redes industriales con protocolos orientados al intercambio de mensajes como el MMS de MAP, de forma que se integra la monitorización visual en los sistemas telemáticos de producción y se da un paso adelante en la consecución del objetivo CIM. La transmisión de vídeo en entornos industriales es hasta ahora un área poco desarrollada, y las soluciones comerciales existentes son sistemas cerrados que además suponen grandes inversiones económicas. Por ello cobra especial interés el desarrollo de un sistema sencillo y abierto (siguiendo el estándar MMS/MAP).

El resto de este artículo está estructurado en 4 apartados más. El apartado 2 es una breve introducción a MAP y MMS, la base de protocolos sobre los que funciona el sistema de transmisión de vídeo. En el apartado 3 se procederá a detallar las características del sistema propuesto. El apartado 4 recoge los resultados obtenidos con un primer prototipo de laboratorio desarrollado por los autores. En el apartado 5 se comentan las conclusiones finales del artículo y las líneas futuras de trabajo. Las referencias citadas a lo largo del artículo aparecen detalladas al final del mismo.

2. MAP y MMS

Las redes industriales empezaron siendo soluciones telemáticas cerradas, de forma que la interconexión de elementos de diferentes fabricantes resultaba complicada y poco eficiente. MAP (Manufacturing Automation Protocol) es una propuesta pionera de las grandes empresas industriales [3] para establecer un estándar de referencia que sustituya a las múltiples soluciones anteriores y posibilite una

interconexión sin fisuras entre los dispositivos de todos los fabricantes. Actualmente, MAP es una solución ampliamente aceptada por la industria y existe gran cantidad de productos de redes que son compatibles con ella.

MAP es una pila de protocolos que sigue el modelo de referencia OSI (Open Systems Interconnection) de ISO (International Standard Organization). Para el nivel de aplicación se han definido diversos protocolos, siendo MMS (Manufacturing Message Specification) el protocolo de aplicación que gestiona la comunicación entre dispositivos industriales programables mediante el intercambio de mensajes. El protocolo MMS presenta cada dispositivo de cara al resto de la red siguiendo el modelo VMD (Virtual Manufacturing Device), una abstracción que homogeneiza el acceso a los distintos recursos. Cada VMD dispone de diversos objetos virtuales que representan las capacidades reales accesibles del dispositivo asociado. Mediante los mensajes MMS, un VMD puede requerir servicios sobre los objetos de otro VMD de una forma homogénea, actuando después el dispositivo real según sus características únicas.

Existen diversas clases de objetos de VMD, siendo los más significativos las variables, los dominios y las invocaciones de programa. Las variables representan valores de diversos tipos, que en cada dispositivo reflejarán un parámetro de funcionamiento concreto. Por ejemplo, en un sensor de temperatura puede haber una variable VMD que almacene el valor concreto de temperatura en cada instante. Las variables pueden ser leídas y modificadas mediante los servicios VMD correspondientes. Los dominios son paquetes de datos, conjuntos de bytes que pueden ser interpretados de múltiples formas. En algún dispositivo puede haber un dominio que represente un programa, mientras que otro disponga de un dominio para almacenar datos. MMS facilita diversos servicios que permiten transferir dominios entre VMD's. Por último citaremos las invocaciones de programa, que son una abstracción para representar el estado lógico en que se encuentra un programa en una determinada máquina (parado, ejecutándose, en pausa...).

MMS se apoya sobre la capa ACSE (Association Control Service Element) de ISO para establecer la comunicación entre dispositivos a través de conexiones denominadas asociaciones de aplicaciones (AA).. Existe numerosa bibliografía donde obtener información más detallada sobre los estándares MAP y MMS. [3] [6] [7] [8] [9] [10]

3. Descripción del sistema

3.1. Diseño del sistema

En el sistema propuesto distinguimos entre dos tipos distintos de entidades:

- Entidades *MAPCam*: son capaces de capturar vídeo y enviar esta información.
- Entidades *MAPView*: son las que reciben la información de vídeo y la utilizan de la forma correspondiente a cada aplicación concreta.

Las entidades de ambos tipos deben ser modeladas como VMD's, e interactuarán entre ellas mediante los mensajes MMS adecuados. En una red industrial MAP puede haber múltiples entidades MAPCam y MAPView, alojadas en múltiples dispositivos conectados al bus común. Un dispositivo puede albergar más de una entidad. Por ejemplo, un PC conectado a una cámara puede contener una entidad MAPCam que proporcione vídeo al resto de la red mientras otra entidad MAPView, en el mismo PC, recibe vídeo de otros elementos. La estructura general del sistema sigue un esquema cliente/servidor, con las entidades MAPCam actuando como servidores y atendiendo las peticiones de las entidades clientes MAPView.

Respecto a la transmisión de imágenes, se han considerado dos modos de funcionamiento: secuencias de vídeo e imagen fija. La transmisión de secuencias de vídeo proporciona a las entidades MAPView un seguimiento preciso y constante de las imágenes que producen las MAPCam, a costa de una mayor utilización de ancho de banda. En algunas ocasiones, unas imágenes fijas en determinados instantes son suficientes. Por ejemplo, para monitorizar la posición de una compuerta (abierta o cerrada) basta una sola imagen, mientras que una secuencia de vídeo no hace más que proporcionar información redundante. En tales casos se opta por la transmisión de imágenes fijas en instantes determinados, lo cual proporciona una reducción del ancho de banda utilizado y por tanto un mejor aprovechamiento de los recursos de la red.

La transmisión de imágenes de vídeo se caracteriza por el elevado volumen de datos generado. Con el fin de reducir la tasa binaria de transmisión requerida, el sistema utilizará mecanismos de compresión y descompresión de datos. Para la transmisión de secuencias, se ha elegido el estándar MPEG (Moving Pictures Experts Group) de ISO (International Standard Organization), dada su aceptación como el estándar de compresión de vídeo de calidad [11] [12] [15]. En el caso de la transmisión de imágenes fijas, el tipo de compresión a utilizar será el denominado JPEG (Joint Photographic Experts Group).

El vídeo, digitalizado y comprimido mediante MPEG, se convierte en un flujo binario constante. Es necesario, por tanto, un mecanismo que permita enviar un flujo de bytes a través de la red manteniendo una cadencia adecuada para conseguir la calidad de imagen deseada. Como se ha comentado previamente, el sistema de transmisión de vídeo se apoya sobre el estándar MMS, orientado al intercambio de mensajes y no a los flujos de

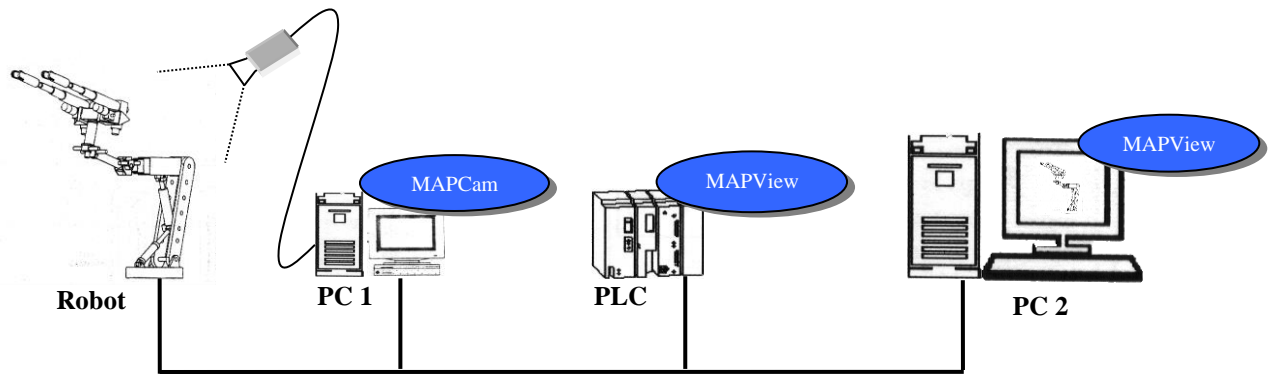


Fig. 1. Ejemplo de escenario de transmisión de vídeo

información continua. Será necesario extender de alguna manera los servicios ‘estándar’ MMS para incorporar un mecanismo capaz de enviar información a un ritmo sostenido. La base más adecuada para estas extensiones son los servicios de transferencia de dominios, ya que son los que más similitudes guardan con el objetivo perseguido.

En la Fig. 1 se observa la actuación de diversos dispositivos implicados en la transmisión de vídeo. El PC 1 implementa una entidad MAPCam que captura vídeo de la cámara de monitorización del robot y lo sirve a las entidades MAPView situadas en el PLC (Programable Logic Control) y el PC 2. Un operario puede ver en la pantalla del PC 2 los movimientos del robot, supervisando el funcionamiento del mismo. Y en el PLC puede procesarse la imagen recibida, de forma que determine el comportamiento de otros dispositivos conectados al PLC.

El sistema de transmisión de vídeo necesita de nuevos servicios no incluidos en MMS. La forma de incorporar estas extensiones será una interfaz a nivel de aplicación que hará de intermediaria entre la aplicación de usuario y la capa MMS. Las peticiones del usuario serán procesadas y expresadas en función de los servicios MMS existentes, y del mismo modo la capa intermedia generará nuevas indicaciones hacia el usuario partiendo de la información que reciba de la capa MMS. Podemos denominar esta nueva capa como VMS (Video Message Specification), de forma que la pila de protocolos MAP quedaría como indica la Fig. 2.

3.2. Objetos MMS en los VMD's MAPCam y MAPView

En los VMD's correspondientes a las entidades MAPCam se definirán diversos objetos MMS que permitirán el funcionamiento del sistema. Cierta número de variables permitirán conocer el estado en que se encuentra el dispositivo de captura de vídeo asociado, así como representar los diferentes parámetros de funcionamiento:

- Variable STATUS: Representa el estado del sistema.
- Variable COMMAND: Las indicaciones y respuestas de los nuevos servicios VMS se

transmitirán a través de MMS en forma de lecturas y escrituras sobre esta variable.

- Variables de parámetros MPEG/JPEG
- Variables de control del dispositivo de captura de vídeo

El VMD MAPCam contará con al menos un dominio: el dominio IMAGE. Se trata de un bloque de datos que representará el fragmento actual de vídeo MPEG, o bien la última imagen fija JPEG capturada. El dominio IMAGE es como un *buffer* por el que va pasando la información de vídeo a transmitir hacia MAPView. Con la periodicidad adecuada, MAPCam deberá ir renovando el contenido del dominio con nuevos datos y enviándolo hacia MAPView mediante los servicios MMS de transferencia de dominios. Notemos que durante estas transferencias MAPCam actúa de cliente y MAPView de servidor, según la nomenclatura de los servicios MMS.

Por lo que respecta a los VMD MAPView, presentarán una estructura de objetos MMS simétrica respecto a los MAPCam. Tendrán variables equivalentes para poder almacenar los valores remotos leídos, así como un dominio IMAGE donde recibir los fragmentos de vídeo.

3.3. Protocolo de comunicaciones VMS

APLICACIÓN	
	VMS
	MMS (ISO 9506)
	ACSE (ISO 8649)
7	
6	Presentación (ISO 8822)
5	Sesión (ISO 8326)
4	Transporte clase 4 (ISO 8072)
3	Red (ISO 8348)
2	LLC (ISO 8802/2) MAC (ISO 8802/4 o ISO 8802/3)
1	Nivel físico

Fig. 2. Pila de protocolos MAP con la extensión VMS para transmisión de vídeo

La comunicación entre las entidades MAPCam y MAPView sigue el protocolo VMS. Es un protocolo orientado a conexión, que funciona sobre la base ofrecida por el ACSE y los servicios MMS.

3.3.1. Establecimiento

Es el primer paso de todo protocolo orientado a conexión. Dicha conexión se da siempre entre entidades MAPCam y MAPView, nunca entre entidades del mismo tipo. La parte iniciadora será una entidad MAPView. Una entidad MAPCam puede recibir peticiones de múltiples entidades MAPView, y mantener conexiones abiertas con todas ellas, mientras que cada entidad MAPView sólo puede estar conectada a una única MAPCam en cada instante. En esta fase se establece una asociación de aplicaciones (AA) entre una aplicación MAPView y otra MAPCam, según la terminología ACSE (ver Tabla 1).

Primitiva	Origen	Descripción
AA-Associate	MAPView	Solicitud de establecimiento de la conexión
AA-Accept	MAPCam	Aceptación de la solicitud de establecimiento
AA-Reject	MAPCam	Rechazo de la solicitud de establecimiento

Tabla 2. Primitivas relativas a la lectura/modificación de parámetros

Primitiva	Origen	Descripción
GetVariableList	MAPView	Solicita una lista de las variables contenidas en la entidad MAPCam.
VariableList	MAPCam	Respuesta a GetVariableList conteniendo la lista de variables.
Read	MAPView	Lectura de una variable.
Write	MAPView	Escritura sobre una variable.
Ack	MAPCam	Reconocimiento positivo de una operación de lectura/escritura.
Nack	MAPCam	Rechazo de una lectura/escritura.

3.3.2. Lectura y modificación de parámetros

Las operaciones descritas a continuación son las que permiten a las entidades MAPView conocer el estado de las MAPCam y programar los distintos parámetros que determinan la transmisión de vídeo. En general, estos servicios se corresponden con total exactitud a los servicios MMS de lectura y escritura de variables (ver Tabla 2).

Primitiva	Origen	Descripción
Frame	MAPView	Solicita una imagen de MAPCam.
DomainTransfer	MAPCam	Transferencia del dominio IMAGE, que contiene la imagen JPEG o el fragmento MPEG actual.
VideoOn	MAPView	Solicitud de vídeo continuo.
VideoOff	MAPView MAPCam	Solicitud/Indicación de finalización de la transmisión de vídeo.

Las entidades MAPView pueden enviar una petición *GetVariableList* a la correspondiente MAPCam para conocer con exactitud las variables que contiene, y en su caso programarlas según se desee. MAPCam emitirá una respuesta *VariableList* donde se enumeran las distintas variables y sus características. Una vez conocida la existencia de una variable, bien *a priori*, bien después de recibir la lista de variables, MAPView puede conocer su valor mediante el servicio de lectura (*Read*) o bien modificarlo mediante el servicio de escritura (*Write*). MAPCam contesta cada petición *Read* o *Write* con un reconocimiento *Ack*, el cual transporta el valor de la variable leída en el caso de las indicaciones *Read*.

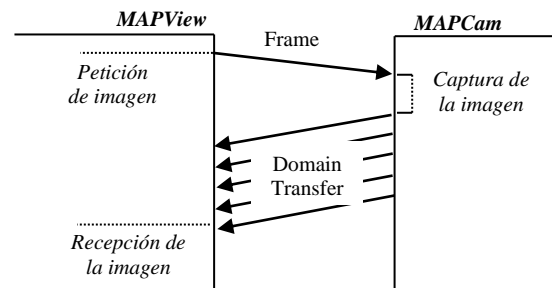


Fig. 3. Diagrama temporal de transmisión de imagen fija

3.3.3. Transmisión de una imagen fija (fotografía de proceso)

Esta operación permite que una entidad MAPView obtenga una imagen fija del estado actual del vídeo en la entidad MAPCam (ver Tabla 3). Este modo de funcionamiento es apropiado para necesidades de monitorización más sencillas, en las cuales una imagen cada cierto tiempo es suficiente, o bien para casos en los que la red presenta un cierto grado de saturación. MAPView solicita a MAPCam una imagen mediante la primitiva *Frame*, dicha imagen es capturada por el dispositivo de vídeo y almacenada en el dominio IMAGE de MAPCam. Finalmente dicho dominio es trasladado hacia MAPView haciendo uso de los servicios de transferencia de dominios de MMS. La Fig. 3 ilustra el proceso.

3.3.4. Transmisión de una secuencia de vídeo

En este modo de funcionamiento es donde se produce realmente la transmisión continuada de imágenes. El proceso es iniciado por MAPView, como en todos los casos, enviando esta vez una petición *VideoOn* a MAPCam. Una vez preparado el dispositivo de captura de vídeo asociado, empezará un bucle indefinido de captura y envío de fragmentos MPEG, con una cadencia T determinada por los parámetros de compresión establecidos. Los fragmentos MPEG llegarán a su destino gracias al servicio de transferencia de dominios MMS, del mismo modo que la transmisión de imágenes fijas, utilizando el dominio IMAGE como *buffer*. MAPView deberá ir reconstruyendo la secuencia MPEG a medida que recibe los distintos fragmentos.

Finalmente, uno de los dos extremos decidirá finalizar la transmisión (generalmente MAPView), y lo hará mediante el envío del mensaje *VideoOff*. En la Fig. 4 se presenta un diagrama temporal de este modo de funcionamiento.

3.3.5. Finalización

La fase de finalización es la contrapartida de la fase de establecimiento (ver Tabla 4). En este caso, cualquiera de los dos extremos puede iniciar la finalización enviando al otro una petición *AA-Release*. La entidad receptora deberá entonces emitir

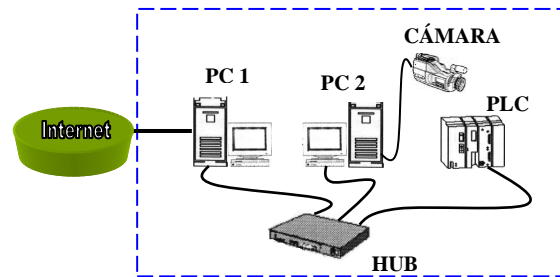


Fig. 5. Diagrama de la red MAP

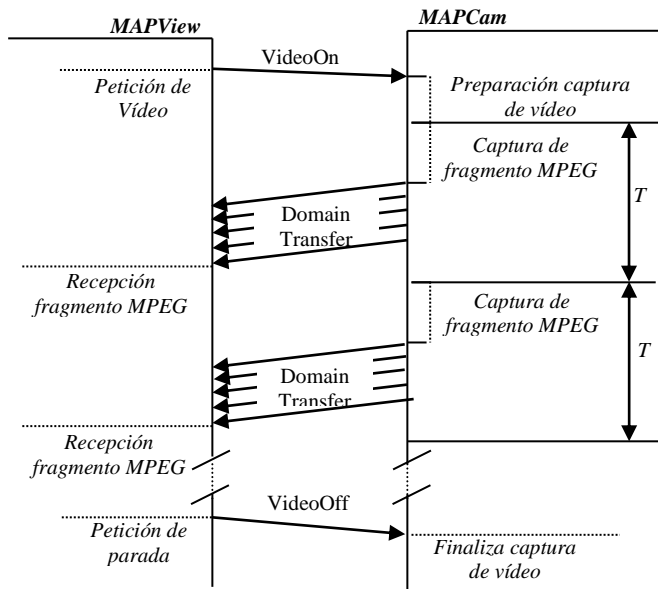


Fig. 4. Diagrama temporal de transmisión continua de vídeo

Tabla 4. Primitivas relativas a la fase de finalización	
Primitiva	Descripción
AA-Release	Solicitud de liberación de la conexión.
AA-Accept Release	Aceptación de la finalización.
AA-Abort	Indicación de finalización forzosa de la comunicación.

una respuesta *AA-Accept Release*, con lo cual la asociación quedará finalizada. Se permite que la entidad receptora emita algunas tramas pendientes antes de contestar definitivamente con *AA-Accept Release*. En circunstancias excepcionales que así lo requieran, la indicación *AA-Abort* puede utilizarse para indicar un cierre inmediato de la comunicación, sin dar opción al otro extremo a enviar ninguna trama más.

4. Prototipo práctico

Con la finalidad de probar el funcionamiento del diseño propuesto, los autores han desarrollado un pequeño prototipo práctico sobre una red MAP cuya estructura puede observarse en la Fig. 5.

La topología de la red se corresponde con un bus común, realizado mediante un Hub al que van conectadas todas las estaciones (dos PC's y un PLC). El acceso al medio es de tipo Ethernet

(CSMA/CD), trabajando a 10 Mbps. Los dos PC's van equipados con tarjetas de comunicaciones CP 1413 de Siemens que implementan los 7 niveles de la pila de protocolos MAP, mientras que el PLC (también Siemens, de la gama S5) dispone de un procesador de comunicaciones CP 1473 como interfaz con la red. En el PC 2 se ha instalado una tarjeta de adquisición de vídeo, que obtiene imágenes de una cámara PAL.

Las entidades MAPCam y MAPView han sido implementadas en los PC's como procesos funcionando sobre un entorno Windows. En el PC 2 se ha implementado una entidad MAPCam capaz de capturar vídeo de la cámara y ofrecerlo al resto de dispositivos de la red. En concreto, se ha dispuesto un proceso en el PC 1 que actúa como MAPView. Este proceso puede, a petición del usuario, obtener tanto imágenes fijas como secuencias de vídeo de la cámara conectada al PC 2, gracias a la interacción con el proceso MAPCam. Cabe notar que el PC 1 se encuentra conectado también a la red externa Internet. Esta característica permitirá en futuros desarrollos que el sistema facilite las imágenes capturadas a clientes remotos de Internet, siguiendo los esquemas de interconexión de redes y acceso remoto descritos en [1] y [2].

Una vez comprobado el funcionamiento de los procesos MAPCam y MAPView, se ha procedido a realizar una serie de mediciones temporales y de tráfico con el fin de intentar establecer una primera valoración del impacto de la transmisión de imágenes sobre el rendimiento de la red. Durante la realización de las pruebas se utilizaron los distintos modos de transmisión de imágenes entre el PC 2 y el PC 1. Las mediciones se llevaron a cabo mediante dos mecanismos distintos: por una parte, una serie de rutinas software incluidas en los programas desarrollados, y por otra, un analizador de protocolos conectado al Hub que implementa el bus común.

4.1. Carácter del tráfico de vídeo

La transmisión de vídeo se lleva a cabo fragmentando la información en segmentos de datos enviados a intervalos regulares de duración T . Estas transmisiones periódicas corresponden a las transiciones *DomainTransfer*. Visualizando en el analizador de protocolos la utilización de la red

durante la transmisión de vídeo se obtiene una imagen similar a la Fig. 6. En ella se observa cómo el tráfico se concentra en ráfagas DomainTransfer de una duración d separadas un tiempo T . La cota máxima de utilización de la red se sitúa alrededor de los 130 Kbps.

Las transiciones DomainTransfer suponen la transmisión de un determinado número de paquetes hasta completar el fragmento de vídeo correspondiente (Fig. 7). A menor tamaño de paquete menor es el *throughput* final que se conseguirá para la información de vídeo durante las ráfagas. Por tanto es conveniente utilizar un paquete tan grande como sea posible. Las tarjetas CP1413 limitan el tamaño máximo de la PDU (Protocol Data Unit) del nivel de aplicación a 1000 bytes. Descontando los bytes de control introducidos por el protocolo, queda una longitud máxima para el paquete de datos de 940 bytes.

La sobrecarga que introduce la pila MAP se manifiesta en dos facetas: la gran cantidad de bytes

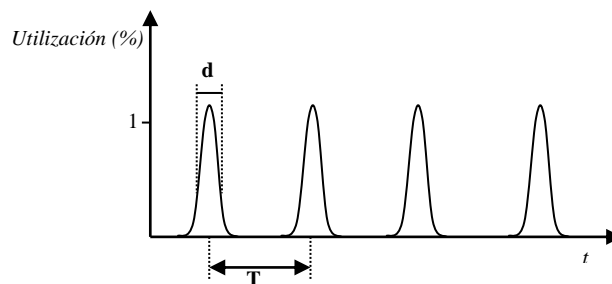


Fig. 6. Tráfico de vídeo en ráfagas DomainTransfer

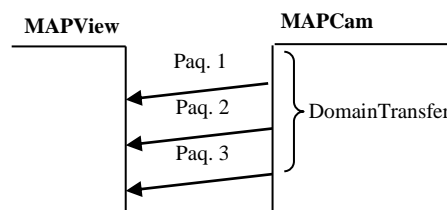


Fig. 7. Descomposición en paquetes de las ráfagas DomainTransfer

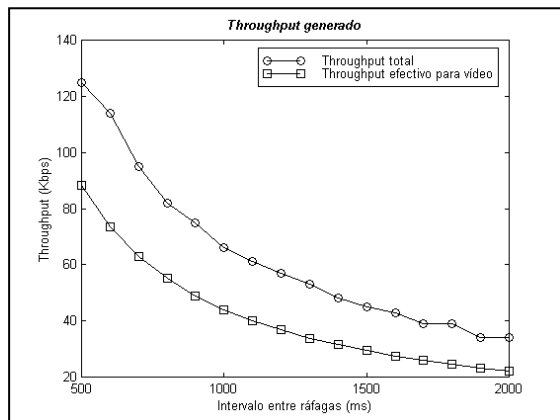


Fig. 8. Throughput total y efectivo dependiendo del intervalo entre ráfagas

adicionales que genera la pila MAP y el retardo de procesado. Para transmitir un paquete de 940 bytes de datos es necesario el intercambio de un total de 1289 bytes, lo que supone una eficiencia inferior al 73% (en general, se verifica que si n es la longitud del paquete, el número total de bytes intercambiados es $n+349$). En cuanto al tiempo de procesado, se ha podido comprobar que desde que se inicia la transmisión de un paquete hasta que empieza la del siguiente transcurren alrededor de 71 ms, con independencia de la longitud de los paquetes. Mientras dura el proceso de envío del paquete la mayor parte del tiempo no se está accediendo al medio sino procesando información, la capacidad del canal se desaprovecha. Quedan patentes las limitaciones de la pila MAP para soportar aplicaciones en tiempo real a causa de la sobrecarga que provoca.

En base a las cifras descritas se puede determinar aproximadamente el *throughput* máximo que se producirá en el canal durante las ráfagas

DomainTransfer. Resulta ser de 141.84 Kbps. Si sólo consideramos los bytes de datos, se tiene que el *throughput* efectivo máximo que queda para la información de vídeo durante las ráfagas es de 103.43 Kbps.

Conjugando la cantidad de información transmitida en cada ráfaga con el intervalo entre ráfagas T se determina el *throughput* final que se obtendrá para la transmisión de vídeo. Utilizando fragmentos de vídeo de 5640 bytes (6 paquetes de 940 bytes) para las ráfagas DomainTransfer, se observó la ocupación del canal en función de distintos valores de T . En la Fig. 8 se presenta el *throughput* total producido por la transmisión de vídeo, así como el *throughput* efectivo de la información de vídeo. Se observa como el *throughput* medio crece a medida que se juntan las ráfagas (menor T), acercándose progresivamente a las cotas calculadas de 141 Kbps para el tráfico total y 103 Kbps para el tráfico efectivo de vídeo.

El rendimiento del sistema se ve afectado también por la longitud de los fragmentos de vídeo. Recordemos que en cada ráfaga *DomainTransfer* se transmite uno de estos fragmentos. En la Fig. 9 se muestra el *throughput* máximo que alcanza el sistema de transmisión de vídeo en función de la longitud de los fragmentos. Los resultados mostrados corresponden a una longitud de paquete de 940 bytes. Dicha longitud, como ya se ha indicado, proporciona el mejor rendimiento. Se puede apreciar cómo la capacidad de transmisión es mayor para fragmentos de vídeo más grandes. Este hecho se debe a que cuanto mayor es el fragmento de vídeo menor número de ráfagas *DomainTransfer* son necesarias para transmitir la misma información. Esta disminución en el número de ráfagas conlleva una reducción en la sobrecarga producida al iniciar y finalizar cada transición *DomainTransfer*. Por ello

el rendimiento global del sistema aumenta al utilizar fragmentos de vídeo mayores. Notemos como con fragmentos de 10 Kbytes se pueden alcanzar velocidades efectivas de vídeo de unos 95 Kbps.

En cualquier caso, el sistema presenta el límite de 103 Kbps como techo para la transmisión de vídeo. Por una parte, este hecho supone que la calidad del vídeo transmitido está limitada, y será necesario sacrificar algo de resolución o reducir el número de cuadros por segundo. Por otra parte, la cota de 141 Kbps para el *throughput* total generado indica que no se producirán interferencias graves sobre el funcionamiento del resto de aplicaciones de la red. Estas características hacen apropiado el prototipo desarrollado para tareas de monitorización visual remota.

4.2. Impacto de la transmisión de vídeo sobre el rendimiento de otras aplicaciones

La descripción previa del carácter del tráfico de vídeo no ha tomado en consideración la carga producida por otras aplicaciones que compartan el medio de transmisión. El estudio de este escenario se llevó a cabo efectuando diversas medidas mientras el analizador de protocolos funcionaba como un generador de tráfico aleatorio. La carga introducida simuló la utilización de la red normalmente producida por otras aplicaciones industriales. El nivel de tráfico añadido varió desde 0 % hasta 90 % de utilización.

La Fig. 10 muestra la relación entre el *throughput* de vídeo requerido (determinado por el tamaño de los fragmentos de vídeo, 5640 bytes durante las pruebas, y el periodo de ráfaga T especificado) y el *throughput* de vídeo realmente conseguido por el sistema (medido por la aplicación MAPView en el PC1). El gráfico presenta ligeras desviaciones respecto a la relación 1:1 debido a una reproducción imprecisa del periodo T por parte de la aplicación MAPCam. Las desviaciones son relevantes únicamente para valores elevados de *throughput* requerido, esto es, para periodos T más cortos. La consecución de estos valores de *throughput* es posible si se utilizan fragmentos de vídeo de mayor tamaño en lugar de T's más cortos (ver Fig. 9). El hecho que cabe destacar es que la respuesta mostrada reflejó variaciones ínfimas cuando se añadió tráfico de fondo a la red con el analizador de protocolos (desde 0 % hasta 90 % de tráfico añadido). Este comportamiento indica que el sistema de transmisión de vídeo se ve mínimamente afectado por las condiciones de utilización de la red, al menos hasta un nivel de utilización del 90 %.

Sin embargo, otras aplicaciones industriales podrían verse influidas por la transmisión de vídeo. Durante las pruebas, el analizador de protocolos midió el incremento en el nivel de utilización de la red producido por el tráfico de vídeo. Este incremento varió según la cantidad de carga de fondo añadida a la red por el analizador de protocolos: el incremento

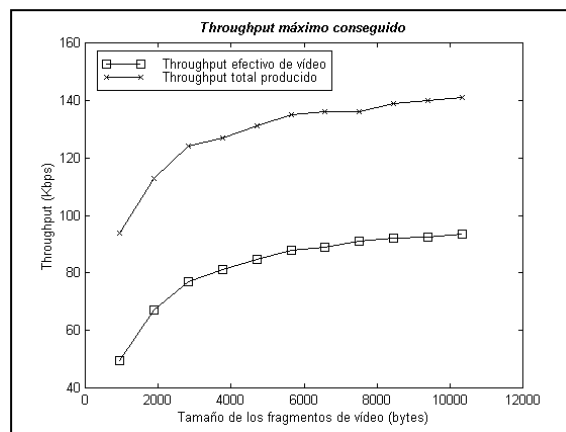


Fig. 9. Throughput máximo conseguido para la transmisión de vídeo en función del tamaño de los fragmentos de vídeo

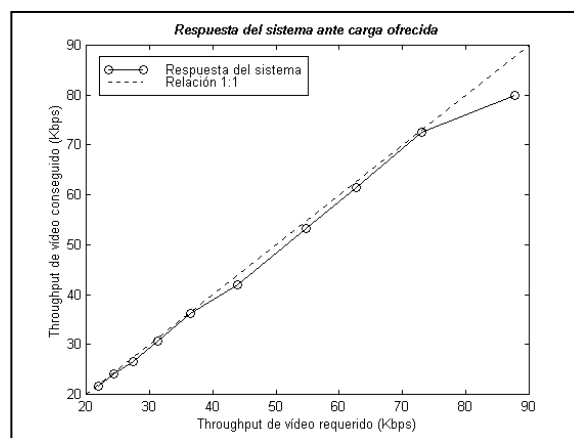


Fig. 10. Relación entre throughput de vídeo requerido y obtenido

fue menor para mayores niveles de carga añadida. Se ha expuesto previamente que la transmisión de vídeo consigue siempre el mismo *throughput*, independientemente del nivel de utilización de la red. Por tanto, los incrementos menores deben interpretarse como una disminución en el *throughput* del sistema de generación de tráfico del analizador de protocolos. Esto es, en condiciones de mayor utilización de la red, el funcionamiento del sistema de transmisión de vídeo redujo el *throughput* del generador de tráfico. Tras comparar los datos obtenidos para diferentes niveles de carga añadida con la situación de 0 % de tráfico de fondo generado, se ha calculado la disminución del *throughput* del analizador de protocolos en cada caso. El resultado aparece en la Fig. 11.

Esta indica que la interferencia producida por el tráfico de vídeo es mínima para niveles bajos de utilización de la red. A medida que el nivel de utilización crece, la interferencia se hace más importante, aunque sin llegar en ningún caso a los 100 Kbps. Estas cifras sugieren que la interferencia producida por la transmisión de vídeo no será un problema para gran cantidad de redes donde la carga habitual del medio de transmisión no es muy elevada.

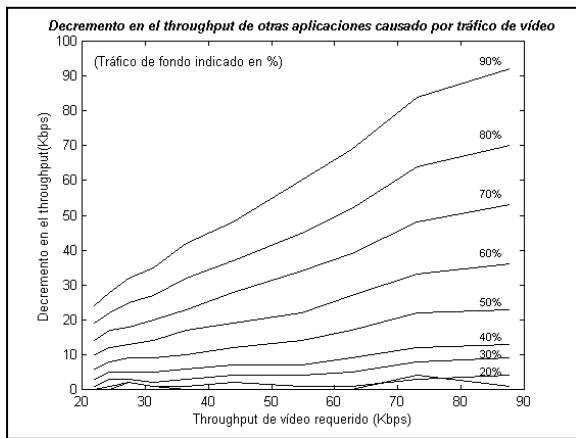


Fig. 11. Disminución del throughput de otras aplicaciones causado por la transmisión de vídeo

5. Conclusiones

En este artículo se ha presentado el diseño inicial de un sistema de transmisión de vídeo sobre redes MAP/MMS. El prototipo desarrollado demuestra la viabilidad de introducir tráfico multimedia sobre el protocolo MMS orientado al intercambio de mensajes, con una interferencia mínima sobre el resto de aplicaciones. De esta manera se integra perfectamente la transmisión de vídeo en las redes de control industrial. La aplicación inmediata de esta nueva funcionalidad es la monitorización visual de dispositivos.

La extensión VMS añade un conjunto de nuevos servicios al protocolo MMS para posibilitar el tráfico de imágenes/vídeo a través de la red, a la vez que se formaliza el control remoto de los dispositivos que capturan las imágenes. Los servicios más significativos son los directamente relacionados con la transmisión de las imágenes. El servicio *Frame* soluciona la obtención de imágenes fijas, mientras que *Video* hace lo propio con secuencias continuas. Basándose en la transferencia de dominios MMS, consiguen automatizar el trasvase de información multimedia entre estaciones de la red. VMS también se encarga de proveer de un método para controlar y programar remotamente los dispositivos de captura mediante los servicios de variables MMS.

Se ha puesto de manifiesto la eficacia limitada de la pila de protocolos MAP para aplicaciones multimedia. En el caso de la transmisión de vídeo, es necesario limitar la calidad de las imágenes (en tamaño o cuadros por segundo) para adecuar los requerimientos de capacidad de transmisión a las posibilidades de los protocolos MAP. Para monitorización, en muchos casos con una imagen enviada cada 2 o 3 segundos es suficiente para conseguir el fin propuesto. Existen alternativas para conseguir prestaciones de tiempo real sobre MAP [13]. Esta implementación introduce un esquema de prioridades y planificadores en la pila MAP, de forma que se asegura que las aplicaciones críticas

cumplen sus límites temporales.

Agradecimientos

Este trabajo ha sido desarrollado y financiado en el marco de la *Comisión Interministerial de Ciencia y Tecnología* (CICYT) bajo el proyecto TIC98-0495-C02-02

Referencias

- [1] Víctor Sempere Payá, Jorge Mataix Oltra, Estanislao Utrilla Ginés. "Remote Access to an Industrial Network MAP 3.0 through Internet". Looking Forward. IEEE Computer Society's Student Newsletter, Spring 98, Vol 6, n°1
- [2] Víctor Sempere Payá, Jorge Mataix Oltra, Pablo García Escalle, Estanislao Utrilla Ginés. "Modelo operativo multifuncional de acceso remoto a una Intranet Industrial genérica". Automática e Instrumentación - Noviembre 1998, n° 290, pp.80-85
- [3] "MAP 3.0 (Manufacturing Protocol Specification)". General Motors, 1988.
- [4] Salvatore Cavalieri, Antonella Di Stefano, Orazio Mirabella. "Impact of Fieldbus on Communication in Robotic Systems". IEEE Transactions On Robotics And Automation, Vol.13, No.1, Feb 1997, pp. 30-48
- [5] Xavier Alcober. "ISA Expo98 – El arte de encapsular estándares". Automática e Instrumentación - Diciembre 1998, n° 291, pp.55-80
- [6] A. Valenzano, C. Demartini, L. Ciminiera. "MAP and TOP communications. Standards and Applications". Addison-Wesley Publishing Company, 1992. ISBN 0-201-41665-4.
- [7] ISO, "Manufacturing Message Specification Standard". ISO/IEC 9506-1, Part 1: Service Definition, 1990. ISO/IEC 9506-2, Part 2: Protocol Specification, 1990.
- [8] Manufacturing Message Specification en Internet: <http://litpc13.epfl.ch/MMS/>
- [9] Herbert Falk, Jeffrey Robins. "An explanation of the architecture of the MMS standard". SISCO, Inc
- [10] _____, "Overview and Introduction to the Manufacturing Message Specification". SISCO, Inc. 1995. <http://www.sisconet.com/techinfo>
- [11] MPEG.ORG – MPEG Pointers and Resources, <http://www.mpeg.org/MPEG>
- [12] Chwan-Hwa Wu, J.David Irwin. "Multimedia and Multimedia Communication: A Tutorial". IEEE Transactions on Industrial Electronics, Vol.45, No.1, February 1995, pp. 4-14
- [13] K. Vijayananda, G. Berthet, P. Raja, J.D. Decotignie. "An Architecture of a MAP stack for real time applications". IFAC Real Time Programming. Lake Constance, Germany, 1994.
- [14] Kang G. Shin, Chih-Che Chou. "Design and Evaluation of Real-Time Communication for FieldBus-Based Manufacturing Systems". IEEE Transactions on Robotics and Automation, Vol.12, No.3, June 1996, pp. 357-367
- [15] Victor O.K. Li, Wanjiun Liao. "Distributed Multimedia Systems". Proceedings of the IEEE, Vol 85, n° 7, Jul 1997

Particionado Hardware/Software para Sistemas Multimedia

Juan P. Castellano, David Sánchez, Onassis Cazorla, Álvaro Suárez
GAC (Grupo de Arquitectura y Concurrencia)
Dpto. de Ingeniería Telemática - Universidad de Las Palmas de G.C.
Campus Universitario de Tafira s/n - 35017- Las Palmas de G.C.
e-mail: {francis, alvaro}@cic.teleco.ulpgc.es

Abstract

In the last years multimedia systems are present in an ever increasing number of applications. A software implementation often can not satisfy system timing constraints. This problem can be solved by adding specific hardware to the system. Our research group have developed a hardware/software codesign environment named GACSYS (GAC's Codesign System) for designing this kind of systems. In this article we present our Hw/Sw partitioning tool. It supports process-level pipelining and takes into account system power consumption. System designer can explore the design space to make latency, area and power trade-offs.

Keywords: *Multimedia System Design, Hardware/Software Codesign and Partitioning.*

1. Introducción

En los últimos años, los sistemas multimedia están cada vez más extendidos en los sectores de la informática y las comunicaciones. Este tipo de sistemas procesa distintos tipos de información, tales como voz, imágenes y datos. La mayoría de estos sistemas se caracterizan por un procesamiento dominado por los datos (*data dominated*), ya que se realiza una gran cantidad de operaciones de cálculo. Además, muchos de estos sistemas necesitan satisfacer algunas restricciones temporales.

En general, una implementación software no es suficiente para satisfacer los requerimientos de estos sistemas. Una solución a este problema consiste en añadir hardware específico para acelerar la ejecución de los mismos. En los últimos años se han desarrollado diversas metodologías para diseñar sistemas hardware/software (codiseño hardware/software) [1][2][3].

Actualmente, nuestro grupo de investigación está desarrollando un entorno de codiseño denominado GACSYS (*GAC's Codesign System*) [4][5][6] (Fig. 1). GACSYS es un entorno orientado hacia aplicaciones dominadas por los datos. Las principales tareas que hemos desarrollado hasta el momento son las siguientes: a) diseño de un lenguaje de especificación denominado VSS (*VHDL-based System Specification*) que permite una codificación sencilla y soporta primitivas con un nivel de abstracción adecuado para especificar sistemas. Este lenguaje contribuye a disminuir el

tiempo de diseño. Además, hemos desarrollado un compilador que genera código VHDL a partir de VSS. b) Diseño de una representación intermedia, basada en un grafo de flujo de datos denominado ASCIS, que soporta especificaciones VHDL. También hemos desarrollado un compilador que genera dicha representación a partir del código VHDL. c) Desarrollo de una herramienta automática de particionado Hw/Sw que permite una mejor exploración del espacio de diseño.

En este artículo presentamos la herramienta de particionado de nuestro entorno. Sus aspectos más destacados son los siguientes: a) *soporta la segmentación a nivel de procesos*, no contemplada en la mayoría de los trabajos publicados. Esta característica permite la realización de un mayor número de compromisos entre el coste y la latencia de un diseño. b) *Tiene en cuenta la potencia consumida por el diseño*, no considerada en muchas herramientas de particionado, aunque es un parámetro importante para el diseño de los sistemas actuales. Además, esta característica permite que el diseñador utilice unidades funcionales con diversos voltajes para realizar compromisos entre el área y la potencia del diseño.

Este artículo está estructurado de la siguiente forma: en el apartado 2, se comentan los trabajos relacionados. En el apartado 3, se describe la herramienta de particionado. En el apartado 4, se presentan los resultados más interesantes mediante un ejemplo. Finalmente, se presentan las conclusiones y las referencias.

2. Trabajos relacionados

Los principales trabajos que se han publicado sobre particionado Hw/Sw pueden clasificarse atendiendo a múltiples aspectos. En este apartado se han

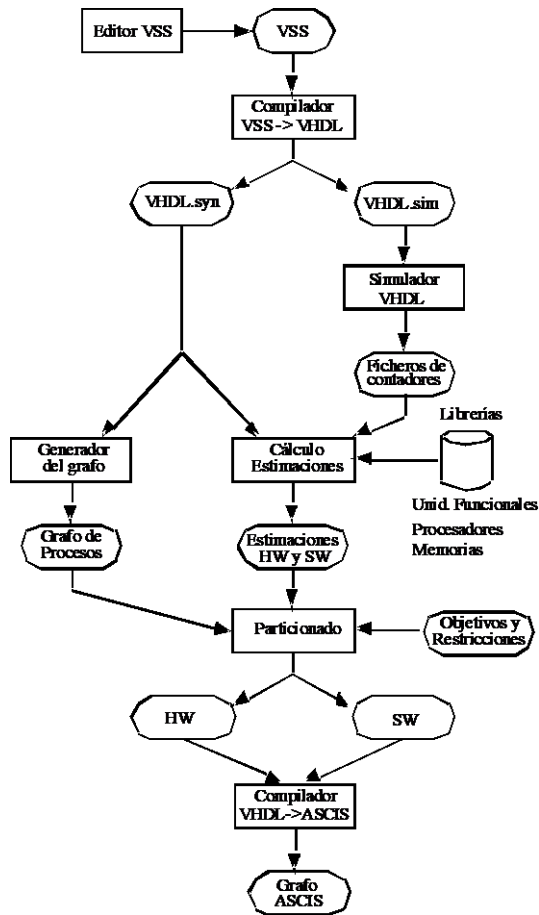


Figura 1. Entorno de Codiseño GACSYS.

realizado diversas clasificaciones atendiendo a los siguientes puntos: tipo de aplicaciones, modelo de representación, arquitectura, segmentación, granularidad, estrategia, algoritmos, y parámetros considerados.

En primer lugar, los trabajos pueden clasificarse en dos grupos atendiendo al tipo de aplicaciones: a) orientados hacia aplicaciones dominadas por el control (*control dominated*) [1][2], y b) orientados hacia aplicaciones dominadas por los datos (*data dominated*) [8][9].

En segundo lugar, los trabajos pueden clasificarse en tres grupos según el modelo de representación: a) trabajos que utilizan un grafo DAG (*Directed Acyclic Graph*) de procesos [8][10], b) trabajos que utilizan un grafo CDFG (*Control and Data Flow Graph*) [9], y c) trabajos que utilizan algún otro tipo de grafo [11][12].

En tercer lugar, los trabajos pueden clasificarse en dos grupos atendiendo al modelo de arquitectura: a) trabajos que utilizan un único procesador programable, y b) aquellos que utilizan varios procesadores programables [8]. A su vez, los

primeros pueden clasificarse en dos grupos: aquellos que utilizan el hardware específico como coprocesador/es [11], y aquellos otros que lo utilizan como procesador/es paralelo/s [1][7][10].

En cuarto lugar, el único trabajo que utiliza la segmentación a nivel de procesos es [8], aunque en [7] se menciona como trabajo futuro.

En quinto lugar, los trabajos pueden clasificarse en cinco grupos según la granularidad que utilizan: a) nivel de proceso [7][8][10], b) nivel de subprograma [13], c) nivel de bloque básico [1][9], d) nivel de sentencia u operación [14], y e) granularidad mixta, es decir, se soportan varios niveles [11][15].

En sexto lugar, los trabajos pueden ser clasificados en dos grupos según la estrategia de particionado que utilizan: a) particionado manual/ interactivo [13][14][15], y b) particionado automático. Estos últimos pueden ser clasificados en tres grupos según la orientación del particionado: a) métodos de orientación software, que inicialmente suponen que todo el sistema se implementa en software, y posteriormente realizan una migración hacia el hardware de algunas partes del sistema mientras no se verifiquen los requerimientos del diseñador [8]. b) Métodos de orientación hardware, que inicialmente suponen que todo el sistema se implementa en hardware, y posteriormente realizan una migración hacia el software de algunas partes del sistema mientras se verifiquen los requerimientos del diseñador [1]. c) Métodos mixtos, en los que no se presupone una distribución orientada hacia el software ni hacia el hardware [7][11][12].

En séptimo lugar, los trabajos pueden clasificarse en dos grupos atendiendo a los algoritmos de optimización que utilizan: a) algoritmos que proporcionan soluciones óptimas, tales como ILP (*Integer Linear Programming*) [8] y programación dinámica [9]. b) Algoritmos heurísticos basados en técnicas tales como *clustering* [12][14], migración de grupos [7][12], *simulated annealing* [11][12], o bien heurísticas *ad-hoc* [10].

Finalmente, los trabajos pueden clasificarse en dos grupos según los parámetros considerados para optimizar el particionado: a) trabajos que sólo consideran el tiempo de ejecución del diseño [8][11][13]. b) Trabajos que tienen en cuenta el tiempo de ejecución y el área del diseño [1][7][12]. En esta clasificación debe señalarse que ninguno de los trabajos estudiados considera la potencia consumida por el diseño. La mayoría de los trabajos anteriores consideran las comunicaciones.

Las características de la herramienta que se presenta en este artículo son las siguientes: a) *tipo de aplicaciones*: dominadas por los datos. b) *Modelo de representación*: grafo acíclico de procesos. c) *Arquitectura*: un procesador programable y

hardware específico utilizado de forma paralela (un procesador específico y una memoria local para cada proceso). *d) Segmentación:* utilizada para la ejecución del grafo de procesos. *e) Granularidad:* a nivel de proceso. *f) Estrategia de particionado:* automática. *g) Algoritmos de optimización:* heurísticos y basados en *simulated annealing*. *h) Parámetros considerados:* tiempo de ejecución, área y potencia del diseño. Las comunicaciones se tienen en cuenta durante la fase previa de estimaciones (Fig. 1).

3. Particionado hardware/software

Dado el grafo de procesos de la especificación del sistema (los nodos representan a los procesos, y los flancos indican el flujo de datos entre los mismos), dadas las estimaciones hardware y software de diferentes parámetros de los procesos (tiempo de ejecución, área y potencia), y dados los objetivos y restricciones requeridos por el diseñador, la herramienta de particionado distribuye los procesos en dos clases, hardware y software, de tal forma que se optimicen los objetivos y se satisfagan las restricciones (Fig. 1).

Las estimaciones de los parámetros se obtienen mediante una herramienta que hemos desarrollado recientemente [16]. Esta herramienta proporciona como salida un fichero que básicamente especifica la siguiente información para cada proceso de la especificación (Fig. 2):

- Nombre del proceso al que pertenecen las estimaciones.
- Estimaciones software: tiempo de ejecución (T), área (A) y energía (E) que se han estimado para la ejecución del proceso en el procesador seleccionado por el diseñador.
- Memorias que se han utilizado durante el proceso de evaluación de las estimaciones software y hardware del proceso.
- Número de operadores diferentes que tiene la descripción VHDL del proceso.
- Configuraciones de unidades funcionales que pueden utilizarse para una implementación hardware del proceso. La información especificada para cada configuración es la siguiente: a) número máximo de unidades funcionales para cada operador, b) código de las unidades funcionales, c) tiempo de ejecución (T) estimado para el proceso, d) área (A) estimada que ocupa el proceso, e) energía (E) media estimada que consume el proceso.

En relación a los objetivos y restricciones, el diseñador especifica los siguientes parámetros: a) tiempo máximo para la ejecución del grafo de procesos (T_E), b) número de fases del *pipeline* de procesos, y c) cota superior de área o potencia,

```

proceso_1

Estimaciones software      /* Pentium */
T (seg)      A (mm2)      E (mj)
6.82e-7      2.96e+2      1.09e-2

Memoria : RAM_3

Estimaciones hardware
Memoria : RAM_1

3          /* Número de operadores */

Configuraciones de Unidades Funcionales

Nº      Cod.      T(seg)  A (mm2)  E(mj)
* + >      * + + <
1 1 1      3 1 5      1.5e-6  7.1e-1  1.9e-5
1 1 1      3 2 5      1.6e-6  7.4e-1  1.3e-5
1 2 1      3 1 1 5      1.9e-5  8.6e-1  1.9e-5
1 2 1      3 1 2 5      2.0e-5  8.6e-1  1.6e-5
1 2 1      3 2 1 5      2.0e-5  8.6e-1  1.6e-5
1 2 1      3 2 2 5      2.1e-5  8.6e-1  1.3e-5

```

Figura 2. Ejemplo de fichero generado durante la fase de estimaciones.

según el parámetro a optimizar. Dados estos parámetros, los objetivos y restricciones son los siguientes:

- *Objetivos:* en primer lugar, el tiempo de ejecución de cada fase del *pipeline* debe aproximarse al tiempo máximo permitido, ya que ejecutarlas en un tiempo inferior no es necesario y genera diseños con un coste superior. Este objetivo es típico en la mayoría de sistemas de procesamiento digital de señales. En segundo lugar, debe optimizarse el área o la potencia, según decida el diseñador.
- *Restricciones:* en primer lugar, el tiempo de ejecución de cada fase del *pipeline* no debe superar el tiempo máximo permitido. En segundo lugar, el área o la potencia, según sea el objetivo a optimizar, no debe superar la cota máxima especificada por el diseñador.

Los aspectos más destacados de nuestra herramienta de particionado son los siguientes:

- *Soporta la segmentación a nivel de procesos* (Fig. 3): como se ha mencionado en los trabajos relacionados, la mayoría de los mismos no considera esta posibilidad. Esta característica permite al diseñador la realización de un mayor número de

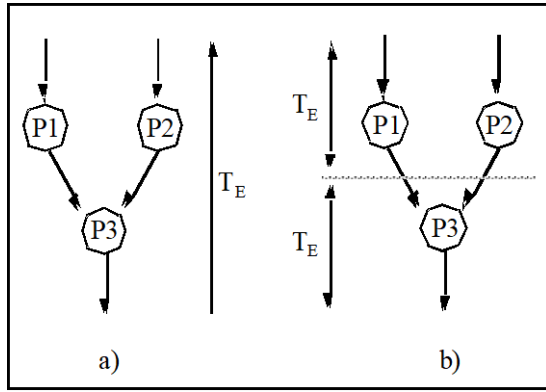


Figura 3. (a) Grafo de procesos no segmentado y (b) posible segmentación del grafo en dos fases.

compromisos entre el coste (área y/o potencia) de las soluciones y la latencia para ejecutar el grafo de procesos.

- *Considera la potencia consumida por el diseño:* este parámetro tampoco es considerado en la mayoría de las herramientas de particionado. Sin embargo, es un parámetro importante para el diseño de los sistemas actuales. En cuanto a las estimaciones de potencia, debemos puntualizar dos aspectos: a) hemos supuesto una implementación del sistema con tecnología CMOS, en la que el principal consumo de potencia se produce durante el régimen dinámico de los transistores (*switching power*) [17][18]. b) También hemos supuesto la utilización de técnicas de *power-down*, es decir, la deshabilitación de la señal de reloj de algunos *latches* para que las unidades funcionales que no son utilizadas en un determinado ciclo no consuman potencia [18][19].

3.1. Función de coste

La función de coste tiene en cuenta aquellos parámetros que son objeto de la optimización. Por un lado, el área (A) y la potencia (P) del diseño se obtienen sumando los valores de cada proceso, que vienen dados en el fichero de estimaciones. Por otro lado, la función de coste considera el tiempo de ejecución (T) no utilizado en cada fase del *pipeline* (Expr. 1), es decir, la diferencia entre el tiempo máximo permitido (T_E) y el tiempo de ejecución utilizado en cada fase (T_{Fi}).

$$T = \sum_i^{\text{Fases}} (T_E - T_{Fi})$$

Expresión 1. Tiempo de ejecución no utilizado en las fases del *pipeline*.

Dada una determinada planificación del *pipeline* de procesos, y dada una configuración hardware o software para cada proceso, el tiempo de ejecución de una fase depende de la planificación de los procesos en dicha fase. Esta planificación interna a cada fase la hemos denominado planificación virtual (Fig. 4). Dada una planificación virtual, el tiempo de ejecución de la fase i -ésima (T_{Fi}) se obtiene mediante la Expresión 2 (P_i representa un paso de la planificación virtual).

$$T_{Fi} = \sum_{P_i} \max\{\max\{T_{Hw}, \sum T_{Sw}\}$$

Expresión 2. Tiempo de ejecución de la fase i -ésima del *pipeline*.

En general, en cada paso de una planificación virtual pueden haber procesos implementados tanto en hardware como en software. Por tanto, el tiempo de ejecución de un paso virtual viene dado por el máximo valor de entre los dos siguientes:

- Por un lado, el tiempo de ejecución del proceso más lento de entre los implementados en hardware ($\max\{T_{Hw}\}$).
- Por otro lado, el tiempo de ejecución de los procesos implementados en software. Como se utiliza un único procesador, dicho tiempo se obtiene sumando el tiempo de cada proceso ($\sum T_{Sw}$).

La función de coste que hemos utilizado en el algoritmo de particionado de la herramienta se muestra en la Expresión 3. P_{\min} y A_{\min} representan la potencia y el área mínima del diseño, respectivamente. Los rangos que aparecen en la expresión se utilizan para normalizar los parámetros. Mediante el parámetro α , el diseñador puede dar mayor prioridad a la optimización de uno u otro parámetro.

$$\text{COSTE}_A = \alpha \frac{A - A_{\min}}{\text{Rango}(A)} + (1 - \alpha) \frac{T}{\text{Rango}(T)}$$

$$\text{COSTE}_P = \alpha \frac{P - P_{\min}}{\text{Rango}(P)} + (1 - \alpha) \frac{T}{\text{Rango}(T)}$$

Expresión 3. Función de coste según el objetivo sea la potencia o el área.

3.2. Algoritmo de particionado

El particionado Hw/Sw es un problema de optimización combinatoria NP-completo [20]. En general, no se pueden utilizar métodos que obtienen soluciones óptimas debido al excesivo tiempo de computación. La alternativa para solucionar el problema es la utilización de algoritmos heurísticos. Estos algoritmos permiten realizar compromisos

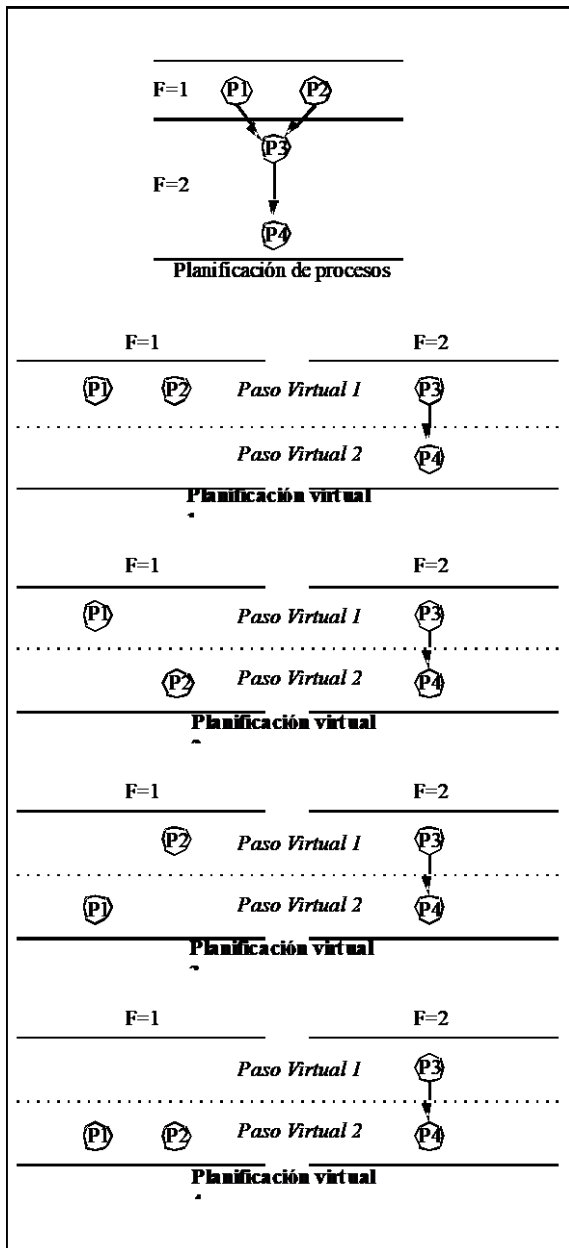


Figura 4. Planificaciones virtuales para una determinada planificación del pipeline de procesos.

entre la calidad de las soluciones y el tiempo de cómputo para obtenerlas. En definitiva, se trata de obtener soluciones *satisfactorias* utilizando tiempos de ejecución *razonables*.

El algoritmo de particionado que hemos desarrollado se basa en el *simulated annealing* [21]. La selección de este algoritmo se basa en los estudios experimentales realizados en [12], que demuestran que el *simulated annealing* obtiene mejores resultados que otros métodos heurísticos tales como los basados en *clustering* o en migración de grupos.

En primer lugar, el algoritmo comienza con la obtención de una solución inicial formada por dos componentes: por un lado, una planificación inicial

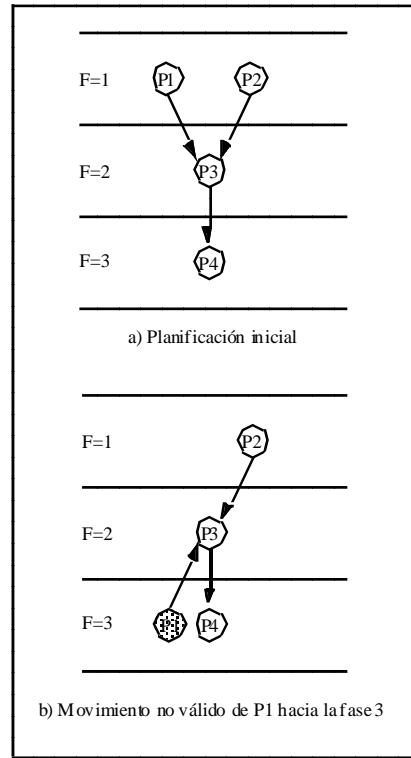


Figura 5. Ejemplo de movimiento de fase no válido.

del *pipeline* de procesos, y por otro lado, una configuración hardware o software de los mismos. La planificación inicial del *pipeline* de procesos se obtiene mediante un algoritmo basado en la planificación ASAP (*As Soon As Possible*) [22]. La obtención de la configuración Hw/Sw inicial se realiza de forma aleatoria.

Una vez obtenidas una planificación y una configuración Hw/Sw iniciales, comienza el proceso de *annealing*. Para obtener una solución vecina se selecciona un proceso de la planificación actual de forma aleatoria. Una vez seleccionado, el proceso es cambiado de fase de *pipeline*, pasando de forma aleatoria a otra fase. El movimiento de un proceso de una fase a otra del *pipeline* debe respetar las dependencias (Figuras 5 y 6).

Este movimiento de fase de un proceso no se realiza en todas las iteraciones del proceso de *annealing*. De esta forma se consigue que cada planificación analizada por el algoritmo permanezca estable durante una serie consecutiva de iteraciones. Esto permite que el resto de procesos se adapten al cambio brusco que se produce cuando un proceso es cambiado de fase.

Finalmente, si el coste de la solución vecina es menor que el coste de la solución actual, la primera es aceptada como nueva solución actual. Si no es así, aún existe una probabilidad de aceptarla que viene dada por una función dependiente de la diferencia de coste entre las dos soluciones y de la

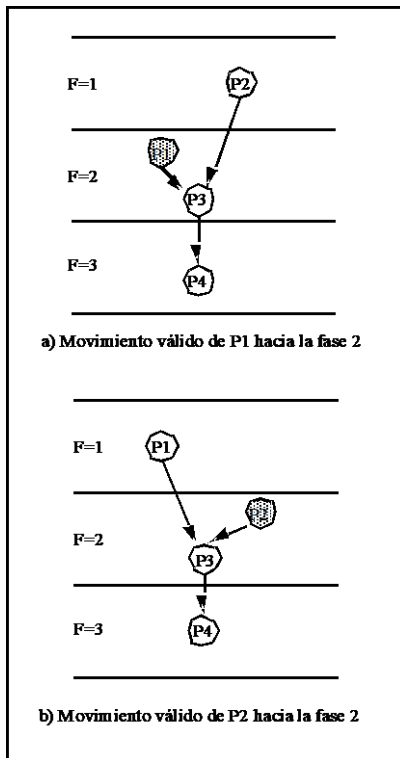


Figura 6. Ejemplo de movimientos de fase válidos.

temperatura del proceso de *annealing* [21].

Para evaluar la calidad de los resultados del algoritmo utilizado en la herramienta, inicialmente se desarrolló un algoritmo exhaustivo para obtener soluciones óptimas y comparar los resultados. Sin embargo, el tiempo de ejecución era tan elevado que no nos permitía obtener resultados para los sistemas que hemos utilizado para *testear* la herramienta. De hecho, varios experimentos fueron abortados después de varias semanas de ejecución. Para resolver el problema se ha optado por comparar con un algoritmo semi-exhaustivo. Este algoritmo solamente se diferencia en un aspecto del utilizado en la herramienta: analiza todas las planificaciones posibles del *pipeline* de procesos.

4. Ejemplo

En este apartado se presentan algunos resultados de los experimentos que se han realizado con la herramienta de particionado. El sistema que se ha utilizado como ejemplo forma parte de un sistema de reconocimiento de voz. El objetivo del sistema es la obtención de un conjunto de coeficientes a partir de una señal de voz muestreada a 8 KHz y dividida en tramas de 128 muestras. Por tanto, el tiempo de ejecución requerido para el sistema es $1.6E-2$ segundos ($128 \cdot 8000^{-1}$). El sistema está formado por ocho procesos concurrentes (el grafo de procesos se ilustra en la Figura 7). La especificación VHDL del sistema tiene 2670 líneas de código.

A continuación se muestran las posibilidades que tiene el diseñador para explorar el espacio de diseño utilizando la técnica de segmentación a nivel

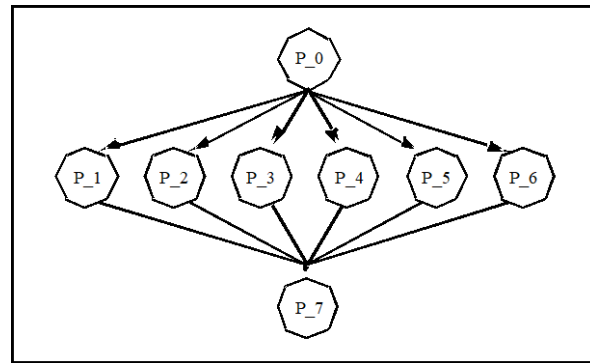


Figura 7. Grafo de procesos.

de procesos. Posteriormente, se analiza la calidad del algoritmo de particionado.

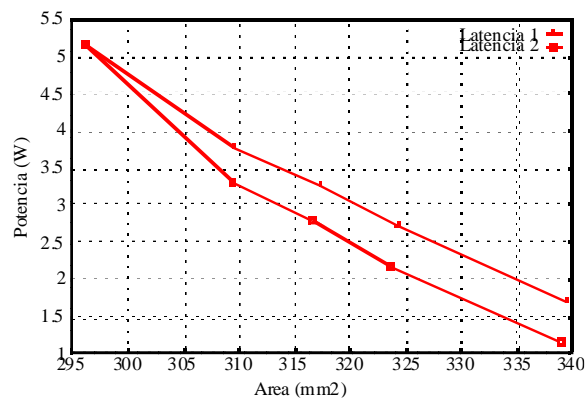


Figura 8. Segmentación a nivel de procesos.

Por lo que a la exploración del espacio de diseño respecta, la herramienta de particionado permite que el diseñador explore ampliamente el espacio de diseño. Así, el diseñador puede optar por diferentes particiones Hw/Sw en función del área y/o la potencia. En la Figura 8 se muestran diferentes soluciones del espacio de diseño en función del área y la potencia. La figura muestra claramente que la potencia consumida por el diseño disminuye conforme aumenta el área permitida para el mismo. Esta tendencia es debida a dos causas: por una parte, al aumentar el área permitida pueden utilizarse unidades funcionales con mayor área y menor voltage. Por otro lado, al permitirse un área mayor pueden implementarse un mayor número de procesos en hardware (Fig. 9), disminuyendo de esta forma la potencia consumida por el microprocesador.

Además, en la Figura 8 se ilustra la disminución de la potencia consumida por el diseño si se utiliza un *pipeline* de dos fases (latencia 2) en lugar de utilizar una única fase (latencia 1). En dicha figura se observa claramente que las particiones que se obtienen utilizando el *pipeline* de dos fases consumen aproximadamente 0.5 vatios menos que las realizadas con una fase. No obstante,

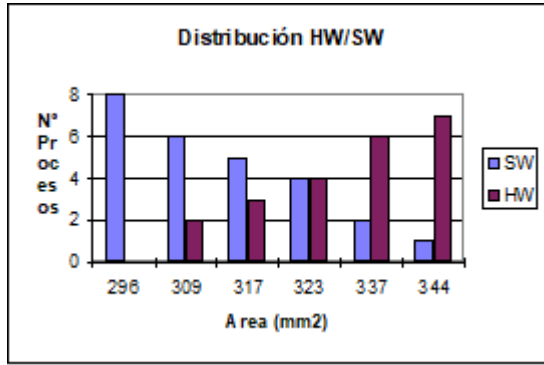


Figura 9. Distribución de procesos Hw/Sw.

no siempre es beneficioso aumentar el número de fases del *pipeline*. En la Figura 10 se muestra que la potencia consumida por las particiones obtenidas para un *pipeline* de tres fases es la misma que la consumida para un *pipeline* de dos fases.

En definitiva, el diseñador puede seleccionar, según las necesidades del diseño, entre particiones con diferentes compromisos entre latencia, área y potencia. Así por ejemplo, si el grafo de procesos representa a un diseño que va a ser implantado en un sistema portátil, el diseñador puede optar por una partición que minimice el consumo de potencia.

Por lo que a la calidad del algoritmo heurístico se refiere, ha sido medida respecto del algoritmo semi-exhaustivo comentado en el apartado anterior. En la Figura 11, se observa que la optimización de la función de coste realizada por el algoritmo utilizado en la herramienta se aproxima bastante a la realizada por el algoritmo semi-exhaustivo. En concreto, la diferencia entre los resultados está aproximadamente entre un 5% y un 10%.

5. Conclusiones

Actualmente, nuestro grupo de investigación está desarrollando un entorno de codiseño denominado GACSYS que está orientado hacia aplicaciones

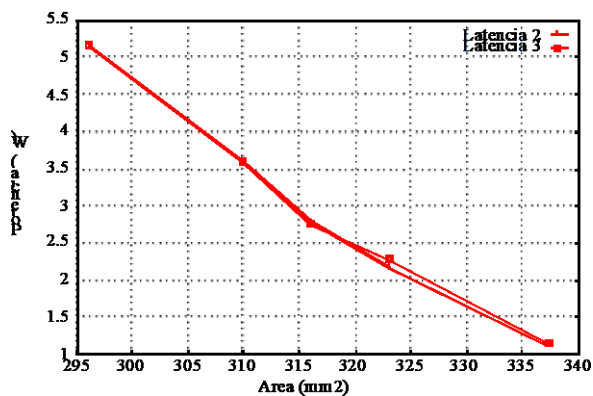


Figura 10. Segmentación a nivel de procesos.

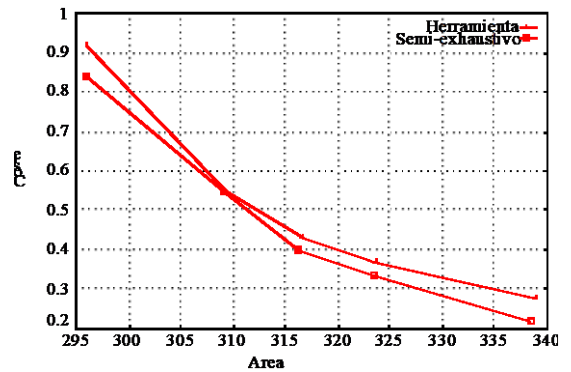


Figura 11. Calidad de los resultados.

dominadas por los datos. En este artículo hemos presentado la herramienta de particionado de dicho entorno. Los aspectos más destacados de la misma son los siguientes: a) *soporta la segmentación a nivel de procesos*, no soportada en la mayoría de los trabajos relacionados. Esta característica permite al diseñador la realización de un mayor número de compromisos entre el coste (área y/o potencia) y la latencia de las soluciones. b) *Considera la potencia consumida por el diseño*, parámetro que tampoco es considerado en la mayoría de las herramientas de particionado.

Por lo que al trabajo futuro respecta, nuestro grupo de investigación ha obtenido recientemente financiación para un proyecto de *codiseño de protocolos MAC y LLC para redes inalámbricas ATM*. Sin embargo, nuestro entorno de codiseño está orientado hacia aplicaciones dominadas por los datos. Por tanto, en la actualidad estamos trabajando en los nuevos aspectos que deberían considerarse en nuestro entorno de codiseño para soportar los protocolos bajo estudio.

En primer lugar, la herramienta que actualmente obtiene las estimaciones debería considerar los siguientes aspectos: a) un modelo de arquitectura basado en memoria compartida parece más adecuado para implementar protocolos, en lugar del actual modelo de memoria distribuida. Además, este cambio de modelo conllevaría la necesidad de estimar con mayor detalle las interferencias que se producen al acceder a la memoria compartida. b) El modelo actual no considera los tiempos de espera que se producen hasta que dos procesos están preparados para comunicarse, ya que la herramienta está orientada hacia aplicaciones en las que los cálculos y las comunicaciones están desacoplados. Sin embargo, estos tiempos de espera deben ser considerados en el codiseño de protocolos.

En segundo lugar, el flujo de control de las aplicaciones dominadas por datos es bastante similar para diferentes datos de entrada. Por tanto, se puede realizar un particionado Hw/Sw totalmente automático ya que se pueden obtener buenas

estimaciones. Sin embargo, no ocurre lo mismo con el flujo de control de un protocolo, por lo que sería adecuado que la herramienta de particionado pueda recibir asistencia por parte del diseñador.

Agradecimientos

Este trabajo ha sido parcialmente financiado por el proyecto CICYT TIC98-1115-C02-02.

Referencias

- [1] R. K. Gupta. *Co-Synthesis of Hardware and Software for Digital Embedded Systems*. Kluwer Academic Publishers, 1995.
- [2] T.B. Ismail, A.A. Jerraya. *Synthesis Steps and Design Models for Codesign*. IEEE Computer Magazine, February 1995, pp. 44-52.
- [3] F. Balarin et al. *Hardware-Software Codesign of Embedded Systems: The POLIS Approach*. Kluwer Academic Publishers, 1997.
- [4] J.P. Castellano, A. Suárez, J.C. Bordón. *Intermediate Representations of Concurrent VHDL-based Specifications*. EUROMICRO Workshop on Parallel and Distributed Processing, 1997, pp. 66-71.
- [5] J.P. Castellano, O. Cazorla, J.C. Bordón, A. Suárez. *A VHDL-based Hardware/Software Codesign Approach*. VHDL User's Forum in Europe, 1997, pp. 159-170.
- [6] J.P. Castellano, D. Sánchez, O. Cazorla, J. Bordón, A. Suárez. *GACSYS: a VHDL-based Hw/Sw Codesign Tool*. 2nd International Workshop on Design and Diagnostics of Electronic Circuits and Systems (DDECS'98), 1998, pp. 293-299.
- [7] C. Carreras et al. *A Codesign Methodology Based on Formal Specification and High-level Estimation*. Intl. Work. on Hw/Sw Codesign, 1996, pp. 28-35.
- [8] A. Bender. *MILP Based Task Mapping for Heterogeneous Multiprocessor Systems*. EURODAC, 1996, pp. 190-197.
- [9] P. V. Knudsen, J. Madsen. *PACE: A Dynamic Programming Algorithm for Hardware/Software Partitioning*. International Workshop on Hw/Sw Codesign, 1996, pp. 85-92.
- [10] A. Kalavade, E. Lee. *A Global Criticality/Local Phase Driven Algorithm for the Constrained Hardware/Software Partitioning Problem*. Intl. Workshop on Hw/Sw Codesign, 1994, pp. 42-48.
- [11] P. Eles, Z. Peng, A. Doboli. *VHDL System-Level Specification and Partitioning in a Hardware/Software Co-Synthesis Environment*. International Workshop on Hw/Sw Codesign, 1994, pp. 49-55.
- [12] F. Vahid, T. dm Le. *Towards a model for hardware and software functional partitioning*. International Workshop on Hw/Sw Co-Design, 1996, pp. 116-123.
- [13] M. Theibinger, P. Stravers, H. Veit. *Castle: An Interactive Environment for HW/SW Co-Design*. Intl. Work. on Hw/Sw Codesign, 1994, pp. 203-209.
- [14] E. Lagnese, D. Thomas. *Architectural Partitioning for System Level Synthesis of Integrated Circuits*. IEEE Trans. on Computer-Aided Design, July 1991.
- [15] T. B. Ismail, K. O'Brien, A. Jerraya. *Interactive System-level Partitioning with PARTIF*. EDAC'94, 1994, pp. 464-468.
- [16] D. Sánchez, J. P. Castellano. *Herramienta de Particionado Hardware/Software para Codiseño de Sistemas*. Proyecto Fin de Carrera, ETSIT, ULPGC, Enero 1999.
- [17] A. Raghunathan, N. Jha. *Behavioral Synthesis for Low Power*. Intl. Conf. on Computer Design: VLSI in Computers and Processors, 1994, pp. 318-322.
- [18] R. S. Martin, J. P. Knight. *Power-Profiler: Optimizing ASICs Power Consumption at the Behavioral Level*. Proc. of the DAC, 1995, pp.42-47.
- [19] E. Macii, M. Pedram, F. Somenzi. *High-Level Power Modeling, Estimation and Optimization*. In Proc. of the 34th DAC, June 1997, pp. 504-511.
- [20] D. S. Hochbaum. *Approximation Algorithms for NP-hard Problems*. PWS Publishing Company, 1997.
- [21] K. A. Dowsland. *Modern Heuristic Techniques for Combinatorial Problems*. McGraw-Hill, 1995.
- [22] D. Gajski, N. Dutt, A. Wu, S. Lin. *High-Level Synthesis: Introduction to Chip and System Design*. Kluwer Academic Publishers, 1992.

Sistema Multicanal de Audio Digital sobre protocolo IEEE-1394.¹

José F. Soler Lucas <jsoler@sunul-1.ita.es>

Francisco J. Cortés Santaolalla <fcortes@sunsp-3.ita.es>

M. Ángel Pérez Costero <maperez@ita.es>

*Departamento de Tecnologías Avanzadas de las Telecomunicaciones
Área de Electrónica y Nuevas Tecnologías del Instituto Tecnológico de Aragón*

C/María de Luna 8-50015-Zaragoza

Tel: 976 738338, Fax: 976 525965

E-mail: telematica@sunul-1.ita.es

Abstract

The aim of this paper is to describe how a prototype for the multichannel transmission of digital audio over the high speed serial bus protocol IEEE-1394 can be built up. To do that, commercial development kits from Texas Instruments have been used due to its ideal characteristics for the proposed aim. A description of the two transfer modes of the protocol is provided so that our "channel" model can be understood. Later the prototype is described in each of its modules, the transmitter based on a PCI-1394 card over a PC platform and the receptor, based on a DSP which performs the selection of the audio channel desired by its user, the processing and play.

1. Introducción

Al llegar la revolución multimedia a los hogares, los ordenadores incorporan capacidades de audio/vídeo avanzadas como la síntesis, grabación y reproducción de audio o el procesado de estas señales. Nos son ya cotidianos sistemas PC multimedia que incluyen DolbyProLogic AC-3, MPEG-III, síntesis de tablas wav o mezclado. De hecho, la diferencia entre productos A/V y productos multimedia para PC se desvanece. La calidad, suele ser el factor diferenciador y pronto, incluso eso cambiará.

Mientras la industria de los ordenadores se esfuerza por mejorar la calidad de la reproducción, la industria audiovisual trata de mejorar los niveles de procesado de señal. Innovaciones recientes en los sistemas de audio digital (sistemas controlados por ordenador de Peavey, IED, TOA, QSC, Crown y otros)[1] distorsionan las líneas diferenciadoras entre procesadores de señal de audio y ordenadores. Ante esta rápida convergencia industrial se hace evidente la necesidad de un estándar digital de interconexión. Los estándares SPDIF y AES/EBU han sido la única solución desde que los equipos de audio digital se generalizaron a partir de los 80. Estos dos estándares están asociados al formato CD-Audio original, transmitiendo un solo flujo de audio estéreo en una conexión punto a punto entre dos dispositivos. La información de control se pasa a través de cables separados o vía IR. Canales adicionales de audio deben ser transferidos sobre múltiples cables SPDIF o AES/ EBU añadiendo costes y complejidad a los sistemas.

La naturaleza asíncrona de SPDIF y AES/EDU complica además la tarea de sincronización de las

conexiones múltiples. Ninguno de estos estándares tiene la capacidad ni la flexibilidad para tratar flujos de datos DVD o DBS. Más aún, el número de dispositivos en un equipo de audio digital típico ha crecido de un simple reproductor CD a varios dispositivos ópticos (DVD, laserdisk, CD) DBS, DSS, DMX, un procesador de sonido envolvente, etc. Está apareciendo una nueva arquitectura distribuida de ordenador y dispositivos A/V que ha de posibilitar la circulación de multitud de flujos de audio, vídeo y datos de control sobre un bus o red y las inherentes limitaciones de SPDIF y AES/EBU los hace obsoletos para esta arquitectura.

El presente artículo describe la construcción del prototipo de un sistema que intenta ofrecer solución a las demandas de alta capacidad y alta calidad de las aplicaciones, los sistemas y usuarios actuales.

2. El Estándar IEEE 1394

2.1. Origen.

El Comité para Estándares de Microcomputador del Instituto de Ingenieros Eléctricos y Electrónicos (IEEE) comenzó en 1986 un proceso de unificación para varias implementaciones sobre bus serie de los estándares VME, Multibus II y Future Bus. Este esfuerzo del IEEE dio lugar al desarrollo original que se convirtió, en otoño de 1995, en el estándar IEEE 1394-1995.

El bus serie 1394 está basado en el bus original FireWire de Apple Computer, que fue diseñado como un sustituto de bajo coste del bus SCSI. Ofrece una conexión escalable y que permite la inserción de conectores "en caliente"; actualmente opera a velocidades de hasta 400 Mbps y tiene

¹ Esta investigación ha sido parcialmente financiada por el Proyecto Europeo ACTS-101 IBCoBN.

previsto llegar hasta los Gbps. Soporta hasta 64 nodos en una red peer-to-peer y no requiere un nodo específico de gestión. Al soportar transferencias de datos asíncronas y síncronas, es idóneo para transmisión de datos en tiempo real (audio, vídeo) y también para otras aplicaciones no específicas (impresoras, escanners) o control.

Pero este protocolo proporciona además un sistema de alimentación de dispositivos lo que combinado con su alta capacidad de transmisión le convierten en el protocolo idóneo para la conexión de sistemas de módulos distribuidos. Con este bus serie de alta velocidad el usuario será capaz de utilizar su PC para controlar no solo los periféricos típicos sino cualquier electrodoméstico, editar contenidos audiovisuales, conectar periféricos a Internet y mucho más. El protocolo 1394 llevará el mundo multimedia al hogar, al permitir experiencias imposibles hoy en día, será la ventana en nuestra casa a las autopistas de la información, tal y como se aprecia en la Fig.1. [2]

IEEE1394-1995 se está implementando hoy en día en equipos electrónicos comerciales. Una revisión del estándar, 1394a, ya está siendo incorporada por la mayor parte de los fabricantes que incluyen 1394 en sus diseños [3]. El siguiente paso, la revisión 1394b que lleva al protocolo a velocidades de Gbps es ya un hecho irrefutable.

2.2. Características.

Las principales características del Bus Serie 1394 son las siguientes[4]:

- Asignación automática de la dirección de los nodos.
- La capa física soporta tanto el medio cable

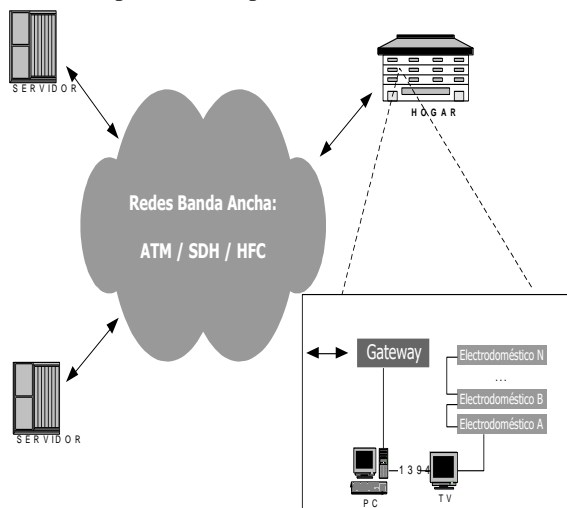


Figura 1: 1394 permitirá el acceso a las superautopistas de la información

como buses tipo backplane.

- Velocidad de transmisión de datos variable, basada en tasas compatibles con RDSI y que

cubren el rango 24.576 Mbps a 49.152 Mbps para backplane y 98.304, 196.608 y 393.216 Mbps para el medio cable.

- El medio cable permite hasta 16 conexiones físicas consecutivas, cada una de 4.5 m, por lo que la máxima distancia física entre dos dispositivos es de 72 m.
- Las transacciones en el bus pueden ser tanto lecturas/escrituras de quadlets (32 bits) aislados como de bloques de éstos. Existe un modo de transmisión síncrono que permite un servicio de ancho de banda garantizado.
- El mecanismo de acceso al bus garantiza la equidad de todos los nodos. El entorno backplane permite un mecanismo de prioridad.
- Este estándar es consistente con la Arquitectura de Registros de Control y Estado (CSR) para Buses de Microordenador ISO/IEC 13213, ANSI/IEEE Std. 1212-1991.

La revisión P1394a (1997) añade algunas mejoras al protocolo como aceleración de reconocimientos y optimización de acceso al bus. [5]

La revisión P1394b está todavía en elaboración y contempla tasas de transmisión del orden de los Gbps y longitudes del enlace de centenares de metros aunque introduce variaciones respecto al medio físico (POF, CAT-5). [5]

3. Modelo del Prototipo

3.1. Servicios de Transferencia.

Como ya se ha mencionado, el estándar soporta dos servicios básicos de transferencia de datos [3]:

- Transferencia Asíncrona: Proporciona un protocolo de distribución de paquetes de longitud variable a una dirección específica que comunica un reconocimiento.
- Transferencia Síncrona: Proporciona un protocolo de distribución de paquetes de longitud variable, en broadcast y a intervalos regulares.

En la Fig.2 se muestran de forma gráfica la concatenación de transferencias síncronas y asíncronas durante un ciclo nominal del protocolo.

Como se aprecia en la figura, cada 125 μ s el bus está dedicado a las transferencias. El 80% de este tiempo de ciclo se dedica a transferencias síncronas, permitiendo la definición de diferentes flujos de información de este tipo, lo que el estándar denomina canales síncronos, asegurando un porcentaje constante de estos 100 μ s para cada uno de ellos en todos los ciclos nominales. Al finalizar la transmisión de todos los canales síncronos, se producen las transferencias asíncronas.

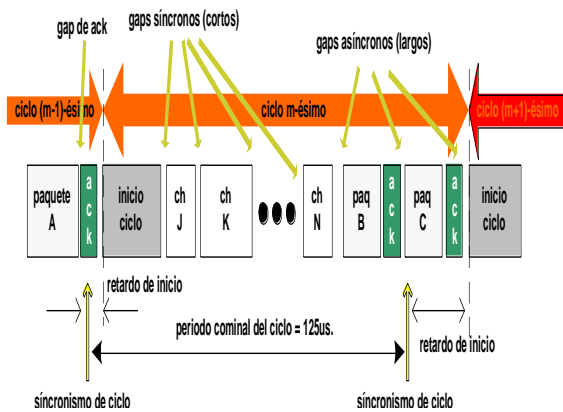


Figura 2: Concatenación de Transferencias

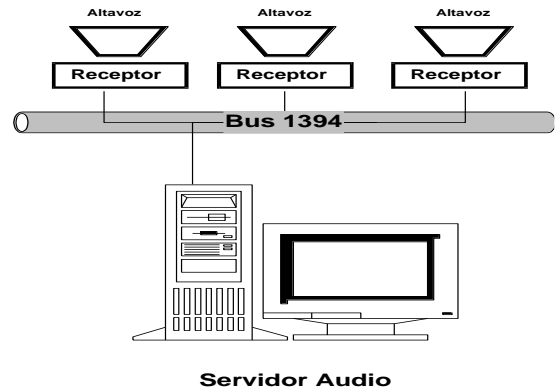


Figura 3: Esquema del Prototipo

El prototipo se basa en este sistema de transferencias, transmitiendo la información de audio digital durante los ciclos síncronos.

Ahora bien, planteamos una posible alternativa durante la transferencia síncrona como se comentará posteriormente.

3.2. El prototipo.

El esquema del prototipo propuesto es el representado en la Fig.3.

Un PC, se encargará de enviar a través del bus IEEE1394 los distintos canales de audio digital y un receptor autónomo, opcionalmente alimentado a través del propio cable 1394, se encargará de seleccionar en recepción el canal que su usuario determine, se encargará del procesado de la información de audio de dicho canal y de su reproducción.

4. Realización del Prototipo

Para la realización del prototipo se recurrió a soluciones comerciales de Texas Instruments, ya que en el momento de desarrollo del prototipo, dicha compañía ofrecía distintos kits para aplicaciones sobre el protocolo IEEE1394 que, en primera aproximación, permitían la realización de aplicaciones de forma rápida y eficaz.

De las distintas posibilidades se eligieron los kits de desarrollo TSBKPCI y TSBKPHRL que se describen a continuación.

4.1. TSBKPCI.

El TSBKPCI se basa en el integrado TSB12LV21A (PCILynx) y en el controlador de la capa de enlace TSB21LV03 para crear una plataforma sencilla de adaptación del bus PCI y el bus 1394. Incorpora una EEPROM que permite guardar la configuración de los registros del TSB12LV21A y cargarlos de forma automática. TSBKPCI proporciona un entorno de programación de

aplicaciones basado en una librería de lenguaje C (LynxsoftAPI) que define las estructuras de datos, secuencia de llamadas y mecanismos de control para realizar [6] la transmisión.

El integrado TSB12LV21A (PCILynx) proporciona una interconexión IEEE 1394-1995 de altas prestaciones, con la capacidad de transferir datos desde la interfaz de la capa de enlace a la física del bus 1394, la interfaz del PCI, y otros dispositivos conectados a la interfaz del bus local de un PC.

La conexión al dispositivo de la capa física 1394, se apoya en el controlador de la capa de enlace (LLC) integrado. Este LLC proporciona el control para transmitir y recibir paquetes de datos 1394 entre una FIFO y la interfaz de la capa física a velocidades de 100, 200 y 400 Mbps. La capa de enlace también permite la recepción de información de estado de los dispositivos de la capa física y da acceso a los registros de control y estado de la capa física. [7]

Una memoria interna de 1 Kbyte puede ser configurada como múltiples FIFOs de tamaño variable, lo que elimina la necesidad de añadir FIFOs externas. Estas FIFOs pueden configurarse para soportar las distintas operaciones del bus 1394, recepción y transmisión asíncrona o síncrona.

La interfaz PCI permite ráfagas de transferencia de hasta 32 bits a 33Mhz y es capaz de operar como maestro (transmite al bus PCI) y como esclavo (recibe del bus PCI). El controlador de DMA usa estructuras de datos tipo lista de control de paquetes (PCL) para controlar las transferencias de datos y permitir que la DMA se realice sin intervención de la CPU. Estas PCLs pueden residir en memoria PCI o bien en memoria conectada al bus local. Las PCLs aportan un conjunto de instrucciones que permite el "lineado", el salto condicional, el control de la transferencia de datos, órdenes auxiliares e informes de estado.

El integrado TSB21LV03 proporciona las funciones de la capa física, codificación, sincronismo y arbitrio, necesarias para la implementación de un

nodo de tres puertos cable, en una red IEEE 1394-1995, a velocidades de hasta 200 Mbps. Cada puerto cable incorpora dos transceptores diferenciales de línea. Estos transceptores incluyen circuitería para monitorizar las condiciones de la línea cuando se determine el estado de la conexión en la inicialización, el arbitrio, la recepción y la transmisión de paquetes. [8] La estructura formada por la tarjeta PCILynx integrada como periférico en el bus PCI del ordenador personal, se muestra en la Fig.4.

4.2. TSBKPRPHRL.

La placa TSBKPRPHRL, va a constituir nuestro receptor autónomo, tal y como se explicó en el modelo del prototipo. Se basa en un DSP de la serie C5x de Texas Instruments, el TMS320C52 y los integrados TSB11C01 y TSB12C01A para dar soporte al desarrollo de aplicaciones sobre bus 1394.

Una PLD compleja, el integrado MACH445, proporciona el control necesario al diseño, implementando un controlador de canal simple DMA que transfiere datos entre el DSP y las memorias SRAM y Flash incorporadas en la placa. Un convertor digital-analógico, el TLC320C46, controlado desde el DSP permitirá la conversión y la reproducción del canal de audio seleccionado por el usuario.

El TSB11C01 proporciona las funciones de transceptor analógico necesarias para la implementación de un nodo de tres puertos en una red IEEE1394-1995 de entorno cable a velocidad de hasta 100 Mbps. Cada puerto incorpora dos transceptores diferenciales que proporcionan la

lógica necesaria para determinar el estado de la conexión, la inicialización, el arbitrio, y para la recepción y transmisión de paquetes. [9]

El TSB12C01A permite una integración sencilla en subsistemas de entrada/salida a través de su interconexión a bus de controlador de 32 bits. Este integrado transmite y recibe de forma correcta paquetes 1394 y genera y chequea el código cíclico de redundancia (CRC). El integrado TSB12C01A tiene tres FIFOs de tamaño ajustable, para facilitar la transmisión asíncrona (ATF), la transmisión síncrona (ITF) y la recepción (GRF) [10]. Un diagrama funcional del TSBKPRPHRL se muestra en la Fig. 5.

Así pues el módulo transmisor del prototipo va a estar constituido por un PC, a partir de una aplicación realizada sobre la placa PCILynx (TSBKPCI), mientras que el receptor va a estar constituido por la placa TSBKPRPHRL que, apoyándose en el DSP, se ocupará de desentramar la información de acuerdo a lo especificado en el siguiente apartado y a continuación se encargará de direccionar la información de audio seleccionada por el usuario hacia el convertor DA integrado, que alimentará a los correspondientes altavoces.

5. Sistema Multicanal

5.1. El concepto de canal.

Bajo la definición del estándar, cada uno de los distintos canales audio debería integrarse en un “canal” síncrono diferente, lo que supondría un incremento de la información no útil (cabeceras, CRCs), generando una considerable ineficiencia.

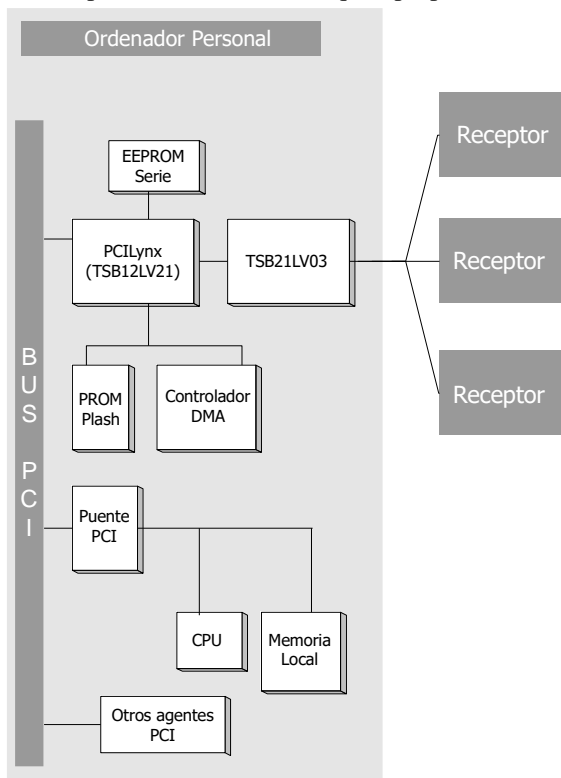


Figura 4 : Estructura PCILynx-PC-Bus 1394

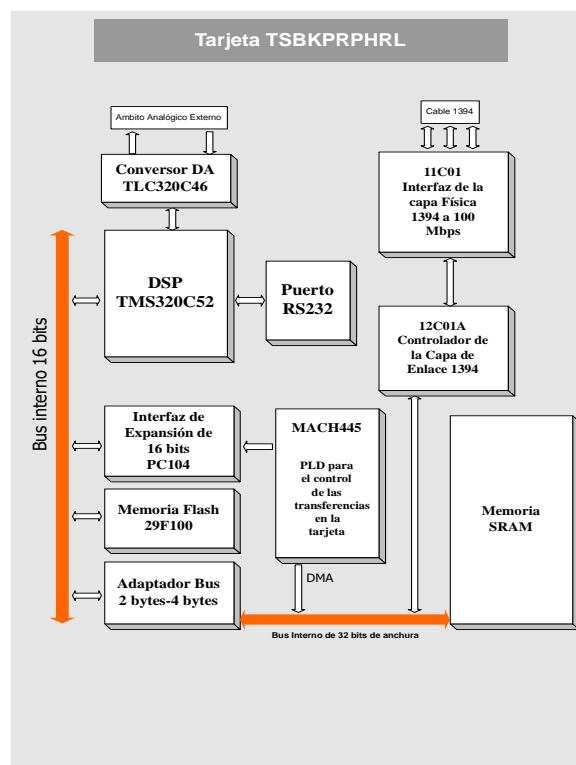


Figura 5 : Estructura Lógica del TSBKPRPHRL

Además, no es posible conocer a priori el número de canales audio que se van a transmitir por el bus y si se limita este número se empobrece la capacidad del sistema.

Esto lleva a un problema: la modificación de los parámetros de ancho de banda reservado por “canal” síncrono no se puede variar de forma dinámica una vez determinada dicha reserva, sin realizar una ruptura en la transmisión de todos los canales. [6] Es decir, si reproduciendo una serie de canales audio, que bajo esta concepción se asocian a “canales” síncronos distintos, se quiere introducir un nuevo canal, la reserva de ancho de banda para este nuevo flujo de información, podría requerir la interrupción momentánea de los restantes flujos de información y la modificación de anchos de banda utilizados por éstos, lo que resulta inviable.

La propuesta es la siguiente. Dentro del ciclo nominal de 125 µs, se va a establecer un único canal síncrono, dentro del cual se va a estructurar la información de todos los canales de audio, en función de su número. De esta forma, la información redundante, debida a cabeceras y CRCs va a ser la correspondiente a un único canal, 3 quadlets, por ciclo nominal, puesto que solo se envía una trama, la correspondiente al único “canal” síncrono definido. El resto del espacio puede ser aprovechado para la información de los distintos canales de audio digital. En la Fig.6 se describe la trama síncrona [3].

Esto puede ser realizado de forma dinámica en función del número de canales de audio digital y sin interrupción de la transmisión, ya que el ancho de banda reservado es único y constante durante toda la

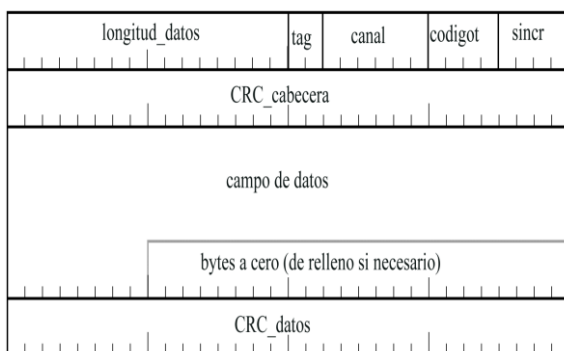


Figura 6: Trama Síncrona

transmisión: el máximo. Así, para la mínima capacidad de transmisión del bus 1394, 98.304 Mbps y durante los 100 µs. de duración del ciclo nominal síncrono, se transmite un flujo de 9830 bits o 307 quadlets (32 bits), lo que deja una capacidad de 304 quadlets de información útil cada ciclo nominal.

Se puede utilizar uno de estos quadlets como método de información para comunicar a los receptores el número de canales que se están transmitiendo, para

que de esta forma el receptor pueda conocer la distribución exacta de la información de los distintos canales de audio digital dentro de la trama 1394, de acuerdo a la política de distribución adoptada. Puede haber distintas políticas, tales como relleno con ceros, información simétrica o asimétrica para los distintos canales, etc. lo que también puede indicarse en el citado quadlet.

5.2. Aplicación en Transmisor.

La aplicación realizada en transmisor se basa en un interfaz gráfico de usuario, en el que las posibilidades son añadir canales audio y asignar los ficheros a reproducir en cada uno de ellos.

Tal y como se ha comentado previamente, un canal audio no afecta a la reproducción del resto de canales que se están ya transmitiendo, gracias a una política de adecuación de los 304 quadlets por trama síncrona. De éstos, 2 se utilizarán como quadlets de control.

Bajo velocidades de reloj de 33 Mhz y con palabras de 32 bits, la capacidad de transmisión del bus PCI es de 133 MBps. (266 MBps bajo reloj a 66 Mhz). Estos valores nos van a marcar el número máximo de canales a transmitir. Suponiendo cada uno de los canales audio como un flujo estéreo de señal muestreada a 44.1 Khz y 16 bits por muestra, tenemos unos requisitos por canal de 1411200 bps. Esto nos lleva a un máximo de 753 canales diferentes sobre el bus PCI (1507 bajo reloj a 66 Mhz.). Ahora bien, conviene no agotar toda la capacidad del bus PCI, para permitir realizar otras operaciones en el PC que servirá como transmisor, sin latencias considerables. Así no se utilizará la capacidad máxima para transferencias de canales de audio, por lo que el máximo impuesto a nuestro prototipo será de 75 canales estéreo de audio diferentes (150 si el reloj es de 66Mhz).

Con estos límites, se ha establecido una política de reparto de los 302 quadlets de la trama síncrona 1394 que el prototipo utilizará y que deberá implementarse en receptor. Esta política se basa en el principio de equidad de todos los flujos de audio y en aprovechar en su totalidad los 302 quadlets mencionados. Así el reparto de este espacio va a ser el siguiente:

Los P primeros canales ocuparán Pn quadlets, mientras que el resto ocuparán Pr, donde

$$\max = 302$$

$$P = (\max - (\text{Int}[\max / n]) \times n)$$

$$Pn = 1 + \text{Int}[\max / n]$$

$$Pr = \text{Int}[\max / n]$$

Por ejemplo, para n = 25 canales diferentes de audio, los primeros 2 (P) canales ocuparán 13 (Pn) quadlets

cada uno y los restantes 23 canales ocuparán 12 (Pr) quadlets cada uno, dentro de la trama.

Durante la fase de implementación, esta primera aproximación teórica hubo de ser modificada debido a las restricciones impuestas por el hardware utilizado en recepción, tal y como se describe en apartados posteriores.

5.3. Aplicación en Receptor.

En receptor, se ha realizado una aplicación que se encarga de solicitar al usuario el número de canal que desea escuchar, presentándole una lista numérica de los canales que hay en el bus, de forma que el DSP, a partir de este número, selecciona la información correspondiente a ese flujo de audio dentro de la trama síncrona. El número de canales viene codificado en uno de los quadlets de control que hemos utilizado.

Se ha hecho necesaria la modificación de las rutinas de interrupción de la tarjeta TSBKPRPHL para permitir la reproducción continua, dado que cuando la interrupción correspondiente a la conversión digital/analógico se producía mientras se realizaba una transferencia DMA de la FIFO de recepción a memoria, se producía un conflicto en la utilización del bus de datos. Esto se solucionó habilitando un buffer de reproducción en memoria interna del DSP (ver Figura 5) de forma que la interrupción de conversión tenga como fuente de datos esta zona de memoria de la tarjeta.

5.4. Control de Congestión.

Cuando el número de canales es alto, la cantidad de información que se transmite y el bajo tiempo de latencia entre la transmisión de dos tramas consecutivas, puede hacer que el procesado de la información de una trama (la selección de la información útil) conlleve un tiempo mayor en el receptor que el correspondiente a la llegada de la siguiente trama, con lo que se perdería información.

Para solventar este problema, la solución óptima sería la transmisión asíncrona desde cada uno de los receptores hacia el transmisor. Cuando un receptor prevé que se saturará en la próxima trama, habilitaría un mensaje hacia el transmisor, en el que le comunica su estado. El transmisor, en las sucesivas tramas y hasta que no reciba un nuevo mensaje desde el mismo receptor, en el espacio correspondiente en la trama al canal que escucha aquél, introduciría el valor de la última información aceptada, de forma que no se perdiese información. (Se asume que todos los receptores asociados a un mismo canal poseen las mismas características). El formato de la trama asíncrona aparece en la Fig.7 [3].

Pero nuestra sorpresa fue el encontrar que los drivers correspondientes a la tarjeta TSBKPCI no permitían simultáneamente la realización de transferencias

síncronas y asíncronas. Ante este contratiempo la dificultad de implementación del prototipo se incrementó.

Como solución se llegó a una de compromiso: si bien no podíamos establecer una comunicación de control receptor-transmisor, decidimos que éste fuera “autocontrolado”. Así la política de control recayó en la fase de construcción del buffer de transmisión en la aplicación de transmisor: Si en la aproximación inicial construimos un buffer con toda la información a transmitir y del cual leía el protocolo 1394 en cada ciclo nominal, la solución de compromiso fue realizar un proceso de creación de buffer, inicio de transmisión y fin de transmisión, con tamaño de buffer divisor del total y repetir el proceso hasta completar el tamaño total de la información, añadiendo un retardo antes de comenzar el siguiente bucle del proceso. Este retardo constituye la clave del sistema, ya que es el que evita sobrescribir datos en la FIFO del receptor.

Este retardo sin embargo, presenta un problema añadido: dependiendo del número de canales, la cantidad de información útil para el receptor varía, tal como se reflejó en el apartado 5.2, por lo que el procesado de la misma sufre también una variación y por lo tanto también la cuantía de este retardo a realizar en transmisor.

6. Futuras Líneas de Trabajo

Con el prototipo actual se ha demostrado la viabilidad de transmisión de audio raw. Las futuras ampliaciones del sistema podrían incluir el soporte de codificación MPEG en el receptor, lo que

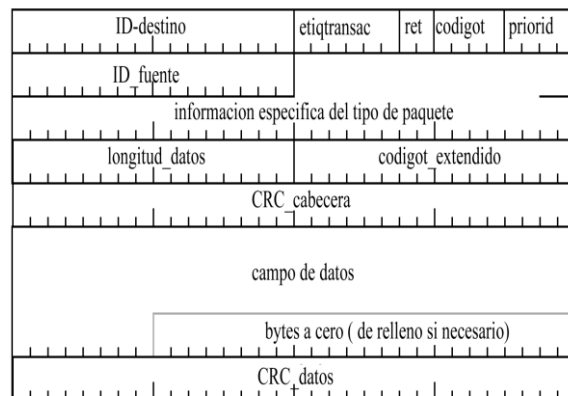


Figura 7: Trama Asíncrona

posibilitaría ampliar considerablemente las capacidades del sistema. Podría estudiarse la incorporación de canales de vídeo, lo que haría del sistema un medio idóneo para su utilización en presentaciones de todo tipo. Otras modificaciones a introducir podrían ser la variación de la política de distribución de los quadlets en la trama, añadir algún

mecanismo de prioridad. En transmisor cabría incorporar la opción de eliminación de uno de los canales. Para ello el receptor debería almacenar el número de canales audio que se estaban transmitiendo en el bus cuando se conectó a él, ya que al eliminar alguno de ellos, en el módulo transmisor, se puede modificar la situación de la información correspondiente a dicho canal dentro de la trama y en función del número de canales. Es por esto, que en cada trama que recibe, debería verificar el número de canales existente y compararlo con el existente en el momento inicial de conexión. La construcción de una placa receptora específica sería el objetivo último de estos desarrollos, de forma que se adecue a las características necesarias, optimizando los puntos que no quedan cubiertos por los kits de desarrollo utilizados en la actualidad (drivers, memoria, incremento capacidad DSPs, etc).

La adopción del Protocolo para la Transmisión de Datos Audio y Música [11] así como la integración del sistema dentro del entorno HAVi [12] deberían incorporarse en función del grado de adopción que dicho protocolo y la mencionada filosofía de interconexión encuentren en la industria audiovisual.

7. Conclusiones.

Se ha construido un prototipo para la transmisión multicanal de audio digital sobre el protocolo IEEE1394, lo que abre el camino para futuras aplicaciones relacionadas con el ámbito de la sonorización.

Las aplicaciones comerciales del presente diseño y prototipo son amplias. Por un lado puede ser utilizado como mecanismo de sonorización de recintos feriales o cualquier conjunto de espacios acotados con información sonora restringida al ámbito del acotamiento (museos, locales y centros comerciales, etc.).

Dentro del ámbito doméstico, puede integrarse como hilo musical de calidad digital, o bien como elemento de sonorización "home theater". De la misma manera, la incorporación de estándares de codificación tipo Dolby AC3, podría llevar a su utilización como elemento de sonorización de cines o teatros.

Apéndice I. Glosario.

AC-3: Técnica de codificación perceptual para audio desarrollada por Dolby Digital, ampliamente utilizada para sonorización de salas de cine por su capacidad multicanal.

AES/EBU: Estándar de interconexión de equipos digitales de audio, desarrollado por la Audio Engineering Society y ratificado por la European Broadcasting Union.

API: Application Program Interface. Conjunto de bibliotecas de desarrollo que facilitan la implementación de aplicaciones.

ATF: Asynchronous Transmission FIFO. Buffer de transmisión para paquetes asíncronos en un dispositivo IEEE1394.

A/V: Audio/Video.

DBS: Direct Broadcast Satellite. Alternativa para la transmisión multicanal de televisión frente a las redes CATV.

DMA: Direct Memory Access.

DMX: Digital Music Express. Servicio multicanal de audio digital a la carta.

DSP: Digital Signal Processor.

DSS: Digital Satellite System. Su desarrollo permitirá la retransmisión (broadcast) de radio y Tv con calidad digital.

DVD: Digital Video Disk. Última generación en sistemas de almacenamiento óptico de la información.

GRF: General Reception FIFO. Buffer de recepción en un dispositivo IEEE1394.

ITF: Isochronous Transmission FIFO. Buffer de transmisión para paquetes síncronos en un dispositivo IEEE1394.

LLC: Link Layer Controller. Se encarga de la implementación de las tareas específicas de la capa de enlace de un protocolo.

PLD: Programmable Logic Device.

POF: Plastic Optical Fiber. Fibra óptica de material plástico de bajo coste aunque con parámetros de dispersión elevados.

SCSI: Small Computer System Interface. Estándar de interconexión paralelo para interconexión de periféricos de ordenador.

SPDIF: Sony Philips Digital InterFace. Estándar de interconexión de equipos digitales de audio, de características similares al estándar AES/EBU.

VME: VersaModule Eurocard bus. Bus de transmisión de 32-64 bits ampliamente utilizado en el sector industrial.

Referencias

- [1] B.Moses. *"Applications of IEEE1394 in Audio/Video Entertainment Systems."*. PAVO 1996.
- [2] M.A. Pérez, F.J.Cortés. *"Arquitectura para la futura red del hogar de banda ancha"*. Artículo presentado en las VII Jornadas de I+D en Telecomunicaciones. Telecom I+D'97.Madrid, 28-29 Octubre 1997. Actas del Congreso.
- [3] *"P1394 Standard for a High Performance Serial Bus"*. The Institute of Electrical and Electronics Engineers (IEEE), 1995. ISBN: 1559375833.
- [4] D.Anderson.*"Firewire System Architecture: IEEE1394"*. PC System Architecture Series. Addison-WesleyPubCo, Marzo 1998. ISBN: 0201694700.
- [5] D.Anderson.*"Firewire System Architecture: IEEE1394a"*. PC System Architecture Series. Addison-Wesley Pub Co, 1999. ISBN:0201485354.
- [6] *"LynxSoft 1394 Software Application Programmer, User's Guide"*, Texas Instruments 1996. SLLU003.
- [7] *"TSB12LV21A (PCILynx) IEEE1394-1995 Bus to PCI Bus Interface"*. Texas Instruments 1997. SLLS273.
- [8] *"TSB21LV03 IEEE 1394-1995 Triple-Cable Transceiver/Arbiter"*. Texas Instruments. 1996. SLLS230A.
- [9] *"TSB11C01 IEEE1394 -1995 Triple Cable Transceiver/Arbiter"*. Texas Instruments. 1996. SLLS167A.
- [10] *"TSB12C01A 1394 High-Speed Serial Bus Link Layer Controller, Data Manual"*, Texas Instruments. 1996. SLLS219.
- [11] Audio/Video Working Group of the 1394 Trade Association. *"Audio an Music Data Transmission Protocol."* Versión 1.0. 1394 Trade Association. Mayo 1997.
- [12] *"Specification of the Home Audio/Video Interoperability (HAVi) Architecture."*V10B. Noviembre 1998. Grundig, Hitachi, Matsushita, Philips, Sharp, Sony, Thomson Toshiba.

Desarrollo de un Servidor de Vídeo MPEG sobre ATM nativo y JAVA

J.M. Fornés y J.A. Gómez Argudo
Area de Ingeniería Telemática Universidad de Sevilla
C. de los Descubrimientos s/n. 41092 Sevilla (SEVILLA)
Telf: 95-4487384, Fax: 95-4487385
E-mail: fornes,ango@trajano.us.es

Abstract

In last years there has been an increase in number of multimedia and real-time applications in the network. Besides, powered by the explosive growth of Internet, we have the appropriated language: Java. This paper describes the implementation of an MPEG Video Aggregation System over ATM developed in Java. Java is a new programming language suited for Internet applications. The performance of this language are slow compared with other programming language like C or C++. But the advantages of this language are: object oriented, to compile once and execute on different processors and operating systems, security, simplicity and fast learning.

1. Introducción

Este artículo describe la realización de un sistema de Agregación de Vídeo[12] MPEG sobre ATM (*Asynchronous Transfer Mode*) mediante el lenguaje de programación Java. Java es un lenguaje especialmente adecuado para aplicaciones Internet.

En cambio, su rendimiento es bajo comparado con lenguajes como C o C++. A pesar de ello, las ventajas de este lenguaje (facilidad de programación, seguridad, orientación a objetos) y la posibilidad de compilarse una sola vez y poder ejecutarse en distintas plataformas y sistemas operativos lo hacen especialmente adecuado para este tipo de aplicaciones, o para versiones de desarrollo previo. El sistema se ha desarrollado sobre el S.O. Linux, pero está trabajando sobre Solaris con tarjetas ATM.

Incluso con rendimientos esperados pobres en teoría, Java permite este tipo de aplicaciones en tiempo real. Nuevas aplicaciones están siendo desarrolladas usando las APIs de ATM nativas. Para nuestro desarrollo se ha utilizado la especificación XTI (X/Open Transport Interface) para ATM nativo [1][2][3].

2. Objetivo

Ante la gran demanda de aplicaciones multimedia, y entre ellas la transmisión de vídeo en tiempo real, se hace necesario optimizar las condiciones de transmisión en redes de banda ancha.

Entre los parámetros de calidad de servicio hay que considerar la calidad de la imagen en las secuencias de vídeo como un parámetro fundamental a vigilar.

Para ello se ha desarrollado previamente un sistema de agregación de vídeo MPEG[4], que consigue una calidad de imagen homogénea entre las fuentes de

vídeo agregadas, que se prueba y contrasta con esta aplicación.

Por otra parte, se intenta evaluar el rendimiento de aplicaciones desarrolladas en Java, un lenguaje que ha despertado grandes expectativas, en situaciones donde teóricamente se ve superado por lenguajes como C y C++.

3. ATM

ATM es la arquitectura de transporte para las redes de banda ancha que están siendo estandarizada tanto por la UIT-T como por el ATM Forum. En las redes ATM, los datos se transmiten en celdas de longitud fija: 48 octetos de datos de usuario y 5 octetos de cabecera. Este tamaño de celda reduce la varianza del retardo, haciendo ATM adecuado para la integración de tráfico consistente en voz, vídeo y datos.

Las redes ATM son orientadas a conexión: es necesario configurar una conexión de canal virtual (VCC) antes de transmitir la información.

Para permitir integrar distintos tipos de tráfico, de distintas características y requerimientos, se hace necesario definir y determinar la calidad de Servicio (QoS, *Quality of Service*), los parámetros de tráfico y las categorías de servicio.

3.1. Calidad de Servicio

Los parámetros de Calidad de Servicio (QoS) son:

- Retardo de transferencia de celda (CTD, *Cell Transfer Delay*): retardo entre el primer bit transmitido por el origen y el último recibido por el destino. Se usan el máximo y el valor medio.
- Variación del retardo de celda pico a pico (CDV, *Cell Delay Variation*): diferencia entre el máximo y el mínimo CTD.

- Tasa de pérdida de celdas (CLR, *Cell Lost Ratio*): porcentaje de celdas que se pierden en la red debidas a errores o congestión y no se reciben en destino.

3.2. Parámetros de tráfico

El tráfico se define por un conjunto de parámetros durante la negociación de la conexión. La red garantiza la QoS para las celdas que no violan estos parámetros. Algunos de ellos son:

- Tasa de pico (PCR, *Peak Cell Rate*): tasa máxima a la que el usuario puede transmitir.
- Tasa permitida (ACR, *Allowed Cell Rate*): tasa permitida concedida por la red.
- Tasa mínima (MCR, *Minimum Cell Rate*): tasa de celda mínima que el usuario solicita.

3.3. Categorías de servicio

El ATM Forum define sobre esta red una serie de categorías de servicios, que son las que se presentan en la tabla 1.

Tabla 1. Categorías de servicio

CBR	Constant Bit Rate
Rt-VBR	Real-Time Variable Bit Rate
Nrt-VBR	Non-Real-Time Variable Bit Rate
UBR	Unspecified Bit Rate
ABR	Available Bit Rate

De ellas, para la realización de la aplicación se ha trabajado con las siguientes: CBR y ABR.

3.4. Tasa de Bit Constante, CBR

La categoría de servicio CBR, *Constant Bit Rate*, esta diseñada para soportar comunicaciones en tiempo real que requieren un ancho de banda fijo y una variación en el retardo de transferencia de celda muy pequeño. Para determinarlo son necesarios dos parámetros: PCR y CDVT.

La tasa de pico de celda, PCR, fija el número máximo de celdas que pueden ser enviadas por el cliente por segundo. La tolerancia a la variación del retardo de celda, CDVT, es un parámetro obligatorio, y debe ser explícitamente especificado por el cliente o implícitamente por la red.

3.5. Tasa de Bit Disponible, ABR

Se puede utilizar para aplicaciones que esperen una cierta garantía en la pérdida de celdas, CLR, pero que puedan controlar el envío de datos de forma dinámica cuando la red se lo demande.

En ABR existe un método de realimentación para controlar la tasa de envío en respuesta a las características de cambio del nivel ATM. Este mecanismo utiliza la capacidad de la red cuando está disponible, y controla la tasa de envío mediante

realimentación para minimizar el CTD, CDV y CLR.

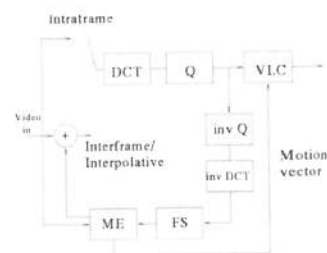
La fuente ABR envía datos a la tasa de celda permitida, ACR, la cual es menor que la tasa PCR y mayor que MCR, que es el único parámetro que se asegura.

4. Tráfico MPEG

En la Fig. 1 se muestra el esquema de un codificador MPEG. En el estándar de codificación MPEG [5] la información espacial de una imagen se particiona en 4 capas: cuadro (frame), rebanada (slice), macrobloque y bloque. Un macrobloque (MB) compuesto por una matriz de 16x16 valores es la unidad básica para la compensación de movimiento. A cada bloque de 8x8 se le aplica la transformada del coseno discreta (DCT). Para el vídeo en color, un MB está compuesto de 4 bloques de 8x8 elementos de luminancia y dos bloques de 8x8 de crominancia. Para cada cuadro, existen tres opciones de algoritmos de codificación: Intracuardo, Intercuardo y codificación interpolativa. La codificación Intracuardo es independiente del resto de cuadros y constituyen las imágenes I; la codificación Intercuardo es predictiva y da lugar a las imágenes P mientras que la codificación Interpolativa determina las imágenes B. Una vez codificadas, los coeficientes de la DCT se cuantifican.

La cuantificación es una técnica de compresión con pérdida de información. Los coeficientes de la DCT se dividen en primer lugar por la matriz de cuantificación y posteriormente por el parámetro de cuantificación. Es posible determinar un parámetro de cuantificación para cada macrobloque. Variando este parámetro se puede conseguir un aumento o disminución del ancho de banda necesario para este flujo MPEG.

Figura 1. Codificador MPEG



El primer coeficiente de la DCT de cada bloque individual, que representa la componente continua (DC), se codifica de forma distinta al resto de coeficientes, los cuales se agrupan primero para

cada coeficiente distinto de cero determinando cuantos ceros le preceden y posteriormente se le asigna a esta pareja de valores un código Huffman preestablecido resultando un código de longitud variable. Para la codificación Intercuadro (P), primero se reduce la redundancia temporal mediante una compensación de movimiento basada en macrobloques con respecto a las imágenes I o P precedentes y como resultado se codifican los macrobloques mediante vectores de movimiento (MV) con código de longitud variable y diferencias respecto a la predicción o simplemente una codificación aislada si se necesitan menos bits. Las imágenes interpolativas (B) se codifican de forma similar pero con una compensación de movimiento bidireccional.

4.1. Calidad de imagen

En las aplicaciones multimedia se considera muy importante garantizar una calidad de imagen constante para la transmisión de vídeo en redes de banda ancha.

El tráfico MPEG VBR implica un uso inferior a los recursos reservados de un contrato ATM CBR. Si se usa un contrato ATM rt-VBR, la red puede multiplexar varias fuentes VBR e incrementar el número de conexiones aceptadas mediante una ganancia estadística.

La pérdida de células puede dañar la calidad de una imagen y a veces de una secuencia de imágenes completa. En la literatura se han propuesto muchas medidas de la calidad de imagen [6] aunque se sigue usando la relación de Señal pico respecto a ruido (PSNR). Tal y como muestran S. Gara y G. Pujolle[7], la calidad de imagen de una codificación MPEG VBR es casi constante. Mientras más pequeño sea el parámetro de cuantificación, mejor será la calidad de imagen de la secuencia.

Con un contrato rt-VBR, el usuario puede tolerar una cierta tasa de pérdida de células debido a la multiplexión estadística. Si en cambio, se utiliza un contrato (ATM Block Transfer/ Delayed Transmission) ABT/DT la pérdida de células se evita y si el contrato es ABR, y la fuente adapta su tráfico se espera el mismo comportamiento respecto a la tasa de pérdida de células.

Es posible evaluar el impacto de la tasa de pérdida de células en un contrato rt-VBR comparado con una versión elástica de ABT/DT[7]. Reducir el tamaño en bits de una imagen a la mitad consigue la misma degradación de calidad que una sola célula perdida. Por tanto, reducciones pequeñas en el tamaño de la imagen en bits obtienen una degradación de imagen no detectable por el ojo humano, haciendo el contrato ABT/DT más apropiado en términos de calidad de imagen. Con un contrato ABR es posible reducir el tamaño en bits de una imagen pero de distinta forma que con ABT/DT.

Para ello se utiliza la Agregación de Vídeo en lugar de modificar el parámetro de cuantificación.

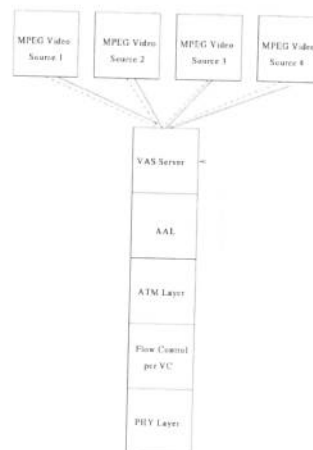
Para medir la calidad de imagen se utiliza como métrica la energía del ruido y puesto que la energía que lleva un código del flujo MPEG es igual al cuadrado de la amplitud del componente no cero de la DCT, la energía del ruido de un macrobloque durante el proceso de agregación es igual a la suma de las energías de los códigos descartados (o no enviados). Esto coincide con PSNR.

4.2. Agregación de Vídeo MPEG

Tal y como se observa en la Fig. 2, el Servidor de Agregación de Vídeo recibe los bits de cada fuente de Vídeo MPEG y realiza todo el proceso de agregación.

La Agregación de Vídeo MPEG consigue que todos los macrobloques contenidos en la correspondiente unidad espacial (imagen o rebanada) de todas las fuentes de vídeo tengan aproximadamente la misma calidad de imagen. Se considera que los coeficientes de la DCT de menor frecuencia tienen más importancia que los de alta frecuencia. Se envía toda la información de cabecera, los vectores de movimiento más los primeros b coeficientes de cada bloque de 8×8 . Esto utiliza un determinado ancho de banda. El resto de coeficientes están sujetos al proceso de agregación con el ancho de banda restante B . Se garantiza este ancho de banda con un contrato CBR de forma constante en el tiempo[8],[9] o con un contrato ABR de forma variable para cada unidad espacial considerada[10].

Figura 2. Sistema de Agregación de Vídeo



Nótese que en cada unidad temporal de agregación se dispone de una cantidad de bits a compartir entre todas las fuentes de vídeo. Esta cantidad de bits se determina, en el caso de ABR, de forma variable en

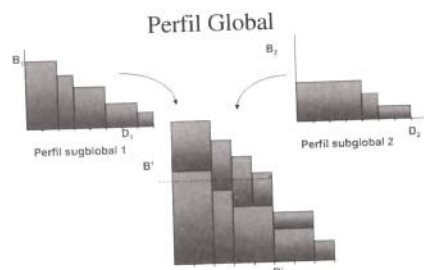
el tiempo entre el MCR y el PCR. El agregador obtiene este ancho de banda disponible de la red.

La Agregación de Vídeo MPEG desarrollada está basada en el perfil bits-vs-distorsión de cada macrobloque, tal y como se muestra en la Fig. 3. Primero se calculan el número de bits que se requieren para transmitir todos los coeficientes del macrobloque. Esto corresponde con un nivel de distorsión cero. A continuación se calculan el número de bits que se requieren así como el nivel de distorsión si se elimina el coeficiente de la DCT de frecuencia más alta distinto de cero. Este proceso se repite hasta que queden sólo los *b* primeros coeficientes que seguro se envían.

A continuación se construye un perfil global de la agregación de todos los macrobloques sumando los perfiles individuales en la dirección vertical y de esta forma se obtiene la cantidad de bits para un nivel de distorsión general. Una vez obtenido este perfil global, se determina el punto donde la envolvente del perfil intersecta la línea horizontal que indica el ancho de banda disponible para este intervalo de tiempo. Así se obtiene el nivel de distorsión global *D'* para todos los macrobloques del intervalo. La cantidad de bits a enviar de cada secuencia de vídeo viene determinada del perfil individual de cada secuencia de vídeo con este nivel de distorsión global.

Nótese que la agregación es transparente al decodificador MPEG. Los datos enviados pueden colocarse fácilmente en formato MPEG estándar tras el proceso de agregación.

Figura 3: Perfil bits-vs-Distorsión



5. ATM API

Para el acceso a ATM nativo se ha empleado la API XTI (X/ Open Transport Interface) [2], la cual proporciona una serie de funciones para el acceso a los protocolos de transporte independientemente del protocolo que se trate. Entre los protocolos soportados se encuentran los propios de ATM. El empleo de esta API se justifica por facilitar la portabilidad del código al evitar el empleo de la API propietaria de las tarjetas ATM utilizadas. Hay que destacar que se han encontrado ciertas dificultades ya que XTI todavía no soporta todo el conjunto de funciones especificadas por el ATM Forum.

Aunque XTI proporciona tanto servicios de transporte orientados a conexión como no orientados, para ATM sólo son posibles los servicios orientados a conexión. Para los servicios orientados a conexión se distinguen cuatro etapas: Control local, Establecimiento de la conexión, Transferencia de datos y Desconexión. Algunas de las funciones asociadas a cada etapa se muestran en la tabla 2.

Para acceder a la API desde JAVA se utiliza el JNI (JAVA NATIVE INTERFACE).

Tabla 2. Etapas y funciones de API XTI

<i>Etapas</i>	<i>Funciones</i>
Control local	t_bind() t_close() t_open() t_unbind()
Fase de conexión	t_accept() t_connect() t_listen()
Transferencia de datos	t_rcv() t_snd()
Desconexión	t_snddis()

6. Desarrollo del Sistema

Se ha desarrollado una aplicación JAVA denominada JAMPEG para desarrollar la Agregación de Vídeo en redes ATM.

La aplicación general Jampeg se divide a su vez en dos aplicaciones concretas, una para ejecutarse en el extremo transmisor de las secuencias de vídeo (*JampegCaller*) y otra para ejecutarse en el lado receptor (*JampegCalled*). La aplicación realiza una transmisión multiplexada de varias secuencias de vídeo en sentido unidireccional con una calidad de imagen mínima especificada por el usuario a través de una conexión ATM. Esta aplicación se ha desarrollado empleando el estándar XTI que trabaja directamente sobre ATM nativo.

6.1. JampegCaller

JampegCaller es la aplicación que se ejecuta cuando un usuario desea transmitir varias secuencias de

vídeo en formato MPEG con una calidad mínima hacia otra estación remota empleando una conexión ATM. Una vez que ejecuta la aplicación, el hilo principal genera para cada secuencia de vídeo que se desea transmitir un hilo secundario que se encarga de controlar dicho flujo de vídeo, tanto en su análisis como en su transmisión.

No existe ningún tipo de comunicación entre los hilos secundarios, tal y como se observa en Fig. 4, pero si existe un paso de información entre los hilos secundarios y el hilo principal.

Resulta fundamental que la sincronización entre los todos los hilos sea perfecta, para evitar problemas de falta de coordinación, los cuales pueden originar que los flujos de vídeo resultantes no cumplan con el estándar MPEG.

Una vez que el hilo principal obtiene los parámetros indicados por el usuario, debe realizar una estimación de la tasa de bits mínima necesaria para establecer la conexión ATM. Esta tasa de bits mínima debe ser suficiente para transmitir las secuencias de vídeo con la calidad especificada por el usuario y con un retardo de imagen aceptable.

Para realizar tal estimación, el hilo principal realiza las siguientes operaciones:

1. Analiza la primera imagen de cada secuencia de vídeo para obtener el número de rebanadas (*NUM_SLICE*) que posee, el número de macrobloques (*NUM_MB*) y la tasa de imágenes por segundo (*PICTURE_RATE*) que especifica la cabecera de la secuencia de vídeo. Como información adicional también se obtiene el tamaño horizontal y vertical de la secuencia.
2. Calcula la tasa de bits mínima necesaria para cada flujo mediante la siguiente expresión:

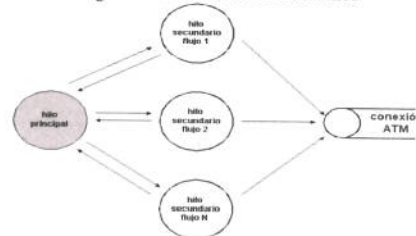
$$TBM = PICTURE_RATE * NUM_SLICE * \left(CAB_SLICE + \frac{NUM_MB}{NUM_SLICE} * (CAB_MB + 6 * COEFF * calidad) \right)$$

Siendo el significado de los parámetros:

- *TBM*: Tasa de bits mínima estimada.
- *CAB_SLICE*: Número de bits de la cabecera de una rebanada.
- *CAB_MB*: Número de bits de la cabecera de un macrobloque.
- *COEFF*: Número de bits para cada coeficiente de un bloque.
- *calidad*: Calidad deseada para la transmisión de las secuencias de vídeo.

Todos estos parámetros son fijados por el usuario.

Figura 4. Comunicación entre hilos



3. La tasa de bits total se calcula como la suma de las tasas de bits que se han calculado para cada flujo. Esta tasa de bits mínima estimada puede ser aumentada por el usuario.

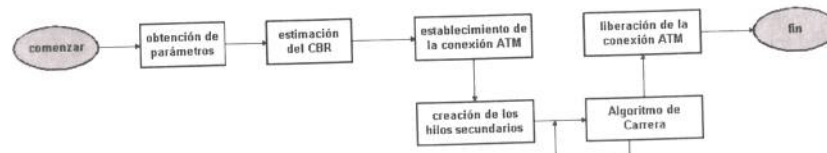
$$TBM_T = \sum_{i=0}^N TBM_i$$

A continuación el hilo principal crea un hilo secundario para cada secuencia de vídeo y entra en un bucle donde recibe de forma coordinada de cada hilo secundario la información referente a una rebanada, aplica el Algoritmo de Agregación de Vídeo y pasa la información correspondiente a cada hilo. Dicha información consiste en el número de bits que pueden transmitir cada hilo para la rebanada que en ese momento están analizando.

También son tareas del hilo principal el establecer la conexión ATM con el extremo remoto así como su posterior liberación al finalizar la transmisión de todas las secuencias de vídeo. Estas tareas pueden observarse en Fig. 5.

Para establecer la conexión ATM se especifica la estación remota con la que se quiere comunicar y el tipo de tasa de bits deseada. Debido a las características de las librerías que controlan la tarjeta ATM no resulta posible establecer la conexión con un usuario concreto. Tampoco resulta posible especificar otro tipo de servicio que no sea CBR. Sin embargo, la aplicación ofrece la posibilidad de simular un servicio ABR sobre CBR mediante un archivo de datos que contiene en cada fila un número que indica la cantidad de bits que en ese momento permite la red transmitir para el conjunto de todos los flujos y que será empleado por el Algoritmo de Agregación.

Figura 5. Tareas del hilo principal



Cada hilo secundario se encarga de gestionar la secuencia de vídeo que tiene asociada. En Fig. 6 se observan las tareas a desarrollar. Para ello analiza el flujo de bits correspondiente y para cada rebanada almacena una tabla de 64 componentes y dos dimensiones. Cada componente indica los bits asociados a la transmisión de un determinado número de coeficientes y también la distorsión que se produce si se transmiten solamente dicha cantidad de coeficientes por rebanada. Esta tabla se pasa al hilo principal, el cual aplicando el Algoritmo de Agregación determina para cada hilo secundario la cantidad de coeficientes que puede transmitir para dicha rebanada. Cuando cada hilo posee esta información se coordinan entre ellos para la transmisión de su rebanada, y de nuevo vuelve a comenzar el ciclo hasta finalizar la transmisión de la secuencia de vídeo que tienen asociada.

6.2. JampegCalled

JampegCalled[11] es la aplicación que se ejecuta en la estación de trabajo donde se desea recibir las secuencias de vídeo. Al ejecutarse la estación de trabajo permanece a la escucha esperando una conexión.

Esta aplicación consta de un hilo principal (Fig. 7), y varios hilos secundarios, que se encargan cada uno de ellos de la gestión de una secuencia de vídeo. Estos últimos a su vez se descomponen en otros hilos terciarios para la representación por pantalla de la secuencia de vídeo.

Parte del código de la aplicación JampegCalled se apoya en el código de la aplicación MpegJava realizado por Joerg Anders en la TU Chemnitz Fakultät fuer Informatik de Alemania[13]. Esta aplicación se trata en concreto de un applet para visualizar el contenido de un fichero en formato MPEG. El código JAVA de MpegJava ha tenido que ser modificado, siendo el cambio más significativo el sustituir el flujo de entrada, para que en vez de ser un fichero MPEG sea la tubería donde se introducen los datos que llegan por la conexión ATM.

Figura 6. Tareas de los hilos secundarios



Al ejecutarse la aplicación Jampeg, el hilo principal permanece esperando la llegada de una conexión ATM. Al llegar una conexión se presenta por pantalla la información correspondiente a las secuencias de vídeo que el usuario remoto desea transmitir. El usuario será el responsable de aceptar o no dicha conexión.

Si la conexión es aceptada, el hilo principal genera para cada flujo de vídeo un hilo secundario para el control de la secuencia de vídeo. A continuación se encarga de recibir los datos demultiplexándolos de tal forma que a cada hilo secundario entrega los datos correspondientes a la secuencia de vídeo que tienen asociada. Cuando se han recibido todas las secuencias de vídeo y se ha finalizado la representación de las mismas por pantalla, se procede a cerrar la conexión y la aplicación.

Cada hilo secundario se encarga de pasar la tubería creada al código modificado de la aplicación MpegJava para el paso de datos. Este código modificado de la aplicación MpegJava se encarga a su vez de generar nuevos hilos para descodificar el flujo MPEG y representarlo por pantalla.

6.3. Formato de los paquetes de datos

Para la comunicación entre las aplicaciones JampegCaller y JampegCalled se ha definido un formato para los mensajes que se intercambian a través de la conexión ATM. Dicho formato es muy simple, y no posee ningún campo para la corrección de posibles errores.

Se pueden distinguir dos tipos de paquetes de datos:

1. Mensajes de control.
2. Mensajes de datos.

En todos los mensajes el primer octeto indica el tipo de mensaje. Los mensajes de control sirven para el establecimiento de la conexión con la estación remota una vez que se ha establecido la conexión ATM, para informar de algún posible fallo en la transmisión de algunos de los flujos, etc.

Figura 7. Tareas del hilo principal



Los mensajes de datos contienen la información correspondiente a una rebanada de una de las secuencias de vídeo.

6.4. Interfaz con el usuario

Al ejecutar JampegCaller aparece el cuadro de diálogo de la Fig. 8.

El cuadro de diálogo consta de cuatro fichas: JAMPEG, ATM, FICHEROS y AVANZADO. La ficha JAMPEG sirve de presentación y no se introduce en ella ninguna información. En las otras tres fichas es donde el usuario introduce los parámetros que desee.

Existen tres botones que son comunes a todas las fichas y que son: OK ALL, CLEAR ALL y EXIT.

La ficha ATM (Fig. 9) posee los siguientes campos: NOMBRE DE LA ESTACIÓN REMOTA y TASA DE BITS.

Figura 8. Ficha JAMPEG



En el campo NOMBRE DE LA ESTACIÓN REMOTA se introduce el nombre de la estación con la que se desea establecer conexión y transmitir las secuencias de vídeo.

En el campo TASA DE BITS se especifica el tipo de servicio para la conexión ATM, este servicio puede ser CBR o también puede ser una simulación de ABR.

Si se elige SIMULACIÓN ABR hay que introducir el fichero que contiene en cada una de sus líneas un dígito que se corresponde con el número de bits disponibles en cada momento para la transmisión agregada de una rebanada por secuencia de vídeo.

La ficha FICHEROS (Fig. 10) consta de dos campos: FICHEROS y CALIDAD DE LAS SECUENCIAS DE VÍDEO. En el campo FICHEROS se introducen los ficheros en formato MPEG que se desean transmitir.

En el campo CALIDAD DE LAS SECUENCIAS DE VÍDEO se selecciona la calidad mínima que se desea que tengan las secuencias de vídeo al transmitirse. Los valores permitidos están en el rango de 1 a 64. Siendo 64 la calidad máxima y 1 la mínima.

La ficha AVANZADO (Fig. 11) posee tres campos: Bits por cabecera de rebanada, Bits por componente de matriz y Bits por cabecera de macrobloque.

Figura 9 Ficha ATM



Figura 10 Ficha FICHEROS



Figura 11 Ficha AVANZADO



El valor de cada uno de estos campos se corresponde de forma respectiva con las variables CAB_SLICE, CAB_MB y COEFF de la expresión utilizada para estimar la tasa mínima de bits para cada flujo de vídeo.

Respecto a la aplicación JmpegCalled, una vez que es ejecutada permanece a la escucha esperando la llegada de un mensaje de conexión. Al llegar dicho mensaje se muestra por consola el número de secuencias de vídeo que se van a recibir, la calidad especificada, y los nombres y tamaños vertical y horizontal de cada una de la secuencias de vídeo. Se ofrece al usuario la posibilidad de aceptar o no dicha conexión.

Si la conexión se acepta se procede a la recepción de las secuencias de vídeo y a su automática representación por pantalla hasta su finalización.

7. Conclusiones

Se ha desarrollado un entorno de Agregación de Vídeo MPEG sobre conexiones CBR y ABR ATM. Utilizando el lenguaje JAVA.

Para poder acceder a las tarjetas ATM, ha sido necesario utilizar el JNI que permite el acceso a una API en C.

Se ha codificado un algoritmo para determinar un ancho de banda CBR que asegura una calidad de vídeo mínima para un conjunto de fuentes MPEG.

Pese a que JAVA es un lenguaje no apropiado para aplicaciones de gran transferencia de información en tiempo real, se consiguen multiplexar, transmitir y representar hasta 4 fuentes de vídeo MPEG simultáneamente.

Para que JAVA pueda utilizarse de forma masiva en aplicaciones de vídeo, sería necesario que se desarrollase las APIs correspondientes que permitan un acceso rápido a las funciones de vídeo. En el momento del desarrollo de esta aplicación, la API JMF (Java Media Framework) de Sun no permitía un funcionamiento en modo *Streams* y por tanto sólo era utilizable en aplicaciones de vídeo almacenado en su totalidad.

Como trabajo futuro, en cuanto JMF permita el modo *Streams* se adaptaría el código en todo lo concerniente a la representación y manejo del vídeo MPEG, consiguiendo una mejora en las prestaciones.

Si bien el concepto de agregación de vídeo MPEG es general, esta aplicación está más orientada a MPEG 1. Se está actualmente adaptando el código para soportar MPEG 2.

Agradecimientos

Merece mención especial en estos agradecimientos la labor desarrollada por Alfonso Tena Venegas que desarrolló la versión inicial del código como parte de su proyecto fin de carrera.

Referencias

- [1] ATM Forum. "Traffic Management Specification Version 4.0", abril 1996.
- [2] Open Group Technical Standard "X/ Open Transport Interface (XTI)", <http://www.rdg.opengroup.org/public/pubs/catalog/c523.htm>
- [3] R. Handel, M.N. Hubber, S. Schroder "ATM Networks, Concepts, Protocols, Applications", Addison-Wesley. 1994.
- [4] J.M. Fornés "Contribuciones a la Transmisión de Vídeo MPEG sobre ATM ". Tesis Doctoral, 1998
- [5] ISO/IEC 11172.11172-2,11172-4. "MPEG parts 1,2,4", aug. 1993.
- [6] A. Webster, C. Jones, M. Pinson, S. Voran, S. Wolf. "An Objective Video Quality Assessment System Based on Human Perception". Human Vision, Visual Processing and Digital Display, Vol. Spie-1913, pp.15-26, 1993
- [7] S. Gara, G. Pujolle. "Study of MPEG VBR sequence quality transmitted with rt-VBR and ABT/DT contracts". Proceedings on the IEEE ATM'97 Workshop. Lisboa, Portugal, pp.149-157, Mayo 1997.
- [8] J.M. Fornés, F.R. Rubio. "A Real Time Video Aggregation over ATM ". Proceedings on the IEEE International Workshop on Factory Communications Systems. WFCS'97. Barcelona, pp.231-238, Noviembre 1997. ISBN: 0-7803-4182-1.
- [9] J.M. Fornés, F.R. Rubio. "DSP Implementation for MPEG Video Aggregation ". Proceedings on the COST211ter Workshop on Image Analysis for Multimedia Interactive Services. WIAMIS'97. Lovaina-la-Nueva, Bélgica, pp.197-202, Junio 1997.
- [10] J.M. Fornés, J.A. Ternerero, F.R. Rubio. "MPEG Video Aggregation over ABR-ATM". Proceedings on the IEEE International Conference on ATM. ICATM'98. Colmar, Francia, pp.59-65, Junio 1998. ISBN: 0-7803-4982-2.
- [11] J.M. Fornés, F.R. Rubio. "JAVA Implementation of MPEG Video Aggregation over Native ATM". To appear on Proceedings on the IASTED Conference on Applied Informatics. AI'99. Innsbruck, Austria, 1999.
- [12] S.C. Liew, C. Tse. "Video Aggregation: Adapting Video Traffic for Transport Over Broadband Networks by Integrating Data Compression and Statistical Multiplexing". IEEE JSAC, Vol.14, No.16, Agosto 1996.
- [13] J.Anders, TU-Chemnitz, http://rnvs.informatik.tu-chemnitz.de/~jal/MPEG/MPEG_Play.html

Visión general de las técnicas de marcas de agua (watermarking) y sus aplicaciones

Juanjo Unzilla, Iñaki Goirizelaia, Eduardo Jacob, Armando Ferro
Grupo de Ingeniería Telemática. E.T.S.I.I. e I.T. de Bilbao (UPV/EHU)
Alda Urquijo s/n, 48013 - Bilbao
Telf: 94 601 4206, Fax: 94 601 4259
{[jtpungaj](mailto:jtpungaj@bi.ehu.es), [jtpgoori](mailto:jtpgoori@bi.ehu.es), [jtpjatae](mailto:jtpjatae@bi.ehu.es), [jtpfevaa](mailto:jtpfevaa@bi.ehu.es)}@bi.ehu.es

Abstract

This paper presents an overview of digital watermark techniques and its applications. Watermarking has become a very active research field and new algorithms are proposed quickly. The main application of watermarking is done in intellectual property rights protection, where the marks inserted in multimedia material can be used to prove who the owner of the work is. Most of the proposed algorithms work in the frequency domain, but there are also methods that work in spatial one. Recently it has been described a new idea which looks for geometric properties in the images to embed the watermark, calling geometric algorithms.

1. Introducción

Es evidente que Internet juega un papel crucial en la actualidad y se prevé que será mayor en un futuro próximo. Por lo tanto, para soportar operaciones de comercio electrónico con material multimedia, es preciso disponer de herramientas que garanticen los derechos de autores y compradores.

Pero para que dichas herramientas sean empleadas deben definirse nuevos marcos legales y nuevos servicios disponibles tanto para autores como para el gran público. El principal campo de aplicación del marcado de material multimedia (imágenes, audio, vídeo, documentos) es la protección de la propiedad intelectual, lo que implica esquemas de utilización del material sujeto a este tipo de restricciones dentro de un contexto jurídico y social que en la actualidad se considera sin definir completamente.

La utilización de técnicas de marcas de agua (watermarking) se ha convertido en un campo de investigación con gran actividad, en el que sólo durante el año 1998 se han contabilizado 103 publicaciones internacionales. La idea básica consiste en insertar información en la imagen mediante la realización de modificaciones sobre la misma, con el objetivo de proporcionar pruebas sobre quién es el propietario de la imagen o a quién a sido enviada o vendida. La realización de este proceso sobre la imagen deberá ser imperceptible para el usuario, no afectando por tanto a su calidad. El objetivo es facilitar el control que se hace de materiales protegidos por derechos de autor, no sólo garantizar que el acceso a la información es el acordado, sino principalmente que no se va a hacer un uso ilegítimo de la misma, es decir, comerciar con segundas copias. Es en este punto donde supone un importante valor añadido para los autores ya que con los sistemas criptográficos podemos autenticar

al comprador, y garantizar que el material ha sido entregado a la persona esperada y sin que nadie haya podido realizar copias ilícitas durante la transmisión. Sin embargo, una vez que el material multimedia está en poder del comprador, éste puede usarlo, copiarlo, revenderlo, etc... sin control por parte del autor.

2. Definición de marca de agua

Una marca de agua puede definirse como una señal digital que puede introducirse en la imagen original y posteriormente ser utilizada para probar quién es el propietario de la imagen. Como aparece recogido ampliamente en la bibliografía especializada [1], de forma general una marca de agua puede ser definida de acuerdo con los siguientes requerimientos:

- ❑ Debe ser invisible a la percepción humana y no debe afectar a la calidad de la imagen.
- ❑ La marca recuperada debe identificar de forma no ambigua al propietario.
- ❑ No debe ser detectada mediante pruebas estadísticas.
- ❑ Debe ser difícil (imposible) de eliminar excepto por el propietario. La marca debe ser robusta a los diferentes algoritmos de procesado de imagen como pueden ser filtros, compresiones, reescalados, etc..

Sin embargo, a medida que se ha ido avanzando en la investigación de nuevas técnicas y nuevos métodos, se han propuesto diferentes escenarios de aplicación para las marcas de agua. Así F. Mintzer y colaboradores [2] proponen hasta ocho posibles escenarios. Para satisfacer las necesidades de estas aplicaciones presentan una clasificación de las marcas en tres categorías:

- ❑ Marcas visibles frágiles.
- ❑ Marcas invisibles frágiles.

- Marcas invisibles robustas.

Las marcas visibles robustas tienen menor interés dado que afectan a la calidad de la imagen y pueden considerarse como logotipos.

2.1. Marcas visibles

Los requisitos que deben cumplir las marcas visibles pueden definirse de la siguiente forma:

- La marca debe ser fácilmente visible.
- La marca debe ser discreta.
- La marca debe ser muy difícil de eliminar.
- La marca debe ser insertada automáticamente con una consistencia visual importante a lotes de imágenes diferentes.

Los tres primeros requisitos son relativamente sencillos de conseguir, sin embargo el cuarto resulta más complejo, ya que pretender que la misma marca pueda ser aplicada a imágenes muy diferentes puede originar que en algunas imágenes la marca sea claramente visible mientras que en otras no llegue a verse.

2.2. Marcas invisibles frágiles

Los requisitos que deben cumplir las marcas invisibles frágiles son los siguientes:

- La marca debe ser invisible para el observador humano.
- La marca debe ser alterada en cuanto se aplican las técnicas más comunes de procesamiento de imagen.
- La marca no puede ser insertada por personas no autorizadas.
- La marca debe poder ser extraída de forma rápida por las personas autorizadas.
- La marca debe permanecer aunque se recorte la imagen.
- La marca extraída debe indicar dónde se han producido las alteraciones.

Uno de los puntos más complejos es determinar si la marca es estrictamente invisible, ya que en muchos casos esta invisibilidad dependerá del visualizador, de la imagen y de cómo es presentada, es decir, de la iluminación, de si es impresa o visualizada en un monitor, de la resolución, etc.. Para ayudar en la creación de la marca se emplean diversos tests que parten de imágenes preparadas en las que la marca tiene más posibilidades de ser visible.

El tercer requisito (personas no autorizadas no deben poder insertar la marca) pretende evitar que se extraiga una marca de una imagen y se inserte en otra. Para ello debe cumplirse que es extremadamente difícil para terceros determinar si la imagen está marcada o no. Una de las propuestas más aceptadas es la utilización de claves para

realizar el proceso de extracción, de forma que estas claves se almacenan de forma separada en una base de datos segura.

El quinto requisito (la marca permanece pese a que sea recortada) es deseable en algunas aplicaciones en las que las imágenes son recortadas (o troceadas), y sin embargo puede no ser tan importante en aplicaciones que no ofrezcan esta opción. En cualquier caso es interesante que el proceso de extracción de la marca permita detectar si se han producido recortes o no.

2.3. Marcas invisibles robustas

Los requisitos que deben cumplir las marcas invisibles robustas son los siguientes:

- La marca debe ser invisible para la percepción humana.
- La marca debe permanecer en la imagen marcada incluso si se realizan sobre ella operaciones de procesamiento.
- La marca debe ser difícil de detectar por personas no autorizadas.
- La marca debe poder ser extraída rápidamente por personas autorizadas.
- La marca debe permanecer en la imagen y debe ser extraíble tras su impresión y posterior escaneado.

Diseñar métodos de marcado para cumplir los requisitos tercero y quinto (difícil de detectar y resistente a impresión y escaneado) es una tarea muy compleja. La forma de crear marcas muy robustas es analizar todos los posibles ataques, lo que supone una tarea ardua y costosa. Por ello se definen unos cuantos requisitos para que la marca sea resistente a las técnicas de procesamiento de imagen más comunes, entre las que se encuentran las empleadas por los programas comerciales para la edición, visualización, impresión o transmisión. De forma resumida podemos citar las siguientes:

- Recortes.
- Modificaciones de brillo y contraste.
- Operaciones de reducción, alargamiento y rotación.
- Compresión con pérdidas JPEG.
- Adición de ruido correlado o no correlado (ataque intencionado).

Si el diseño de métodos que soporten estos requisitos es una tarea compleja, la detección de las marcas en imágenes que han sido sometidas a las operaciones anteriormente descritas resulta aún más complicado. Esto hace que haya que buscar un compromiso entre los requerimientos de creación de marcas robustas y a la vez rápida y fácilmente extraíbles por las personas autorizadas. Por último, para ofrecer sistemas que impidan la inserción de marcas a personas no autorizadas puede emplearse,

al igual que en el caso anterior métodos que utilicen una clave para realizar la extracción.

3. Técnicas relacionadas: Esteganografía

La esteganografía es una técnica que se basa en la inserción de información que permanece oculta sobre otra información (por ejemplo una imagen). Puede utilizarse para enviar información cifrada sobre redes en las que no está permitido el uso de criptografía, siempre que el que controla el canal desconozca que se emplea para ello o simplemente desconozca la clave o el sistema empleado para ocultar la información.

Las marcas de agua pueden verse por tanto como un subconjunto de técnicas de esteganografía. La esteganografía es conocida desde muy antiguo, pero que gracias a la utilización de potentes ordenadores y la representación digital de la información ha cobrado gran importancia en los últimos años. Este interés es también extensible a las marcas de agua [3].

4. Marcas de agua en imágenes digitales

Como ya se ha indicado en los últimos años el marcado de imágenes digitales ha sido un campo de mucha actividad investigadora, y se han propuesto un número importante de nuevos métodos. Como se describe en [1], la mayoría de ellos se aplican a la imagen en el dominio de la frecuencia, usando la DCT (Discrete Cosine Transform, Transformada discreta del Coseno), u otra transformación de la imagen para determinar dónde insertar la señal (marca) que puede ser usada como prueba de propiedad.

4.1 Métodos basados en procesado en el dominio de la frecuencia.

Como se ha indicado anteriormente, la mayoría de los métodos de marcado propuestos operan en el dominio de la frecuencia. Como se describe en [1], los primeros desarrollos se centraron en ocultar información en una señal, sin considerar los requerimientos citados anteriormente. En una aplicación en la que se desea disponer de un canal protegido (seguro) entre dos partes, la resistencia a intromisiones no es estrictamente necesaria si sólo las dos partes conocen la existencia del mismo. Por lo tanto, las primeras propuestas cabe encuadrarlas en la esteganografía.

De forma resumida, vamos a citar algunos de los desarrollos propuestos dentro de la definición de marcado realizada previamente. Caronni [4] describe un procedimiento en el que se añaden patrones geométricos tenues a una imagen. La marca es independiente de la imagen, pero al ser también una imagen tiene una distribución espacial de frecuencias que pueden ser visibles al ser insertadas. Tanaka y colaboradores [5] proponen un método

para insertar una señal en una imagen cuando la imagen es representada por impresión monotonó. Posteriormente, Matsui y Tanaka [6] sugieren diferentes métodos para codificar la marca basándose en si la imagen es representada por codificación predictiva, por impresión monotonó o fax. También propone un método basado en DCT para marcado de secuencias de vídeo.

Koch y colaboradores [7] describen varios procedimientos basados en modificaciones de pares o tripletas de coeficientes de frecuencia computados como parte del procedimiento de compresión JPEG. El orden de rango de estos coeficientes frecuenciales se usa para codificar dígitos binarios. Seleccionan frecuencias de rango medio que habitualmente resisten la compresión JPEG. Bors y Pitas [8] sugieren una restricción lineal alternativa entre los coeficientes DCT seleccionados. Hsu y Wu [9] describen un método en el cual la marca es una secuencia de dígitos binarios que son insertados en las frecuencias de banda media de los coeficientes 8x8 de la DCT.

Swanson y colaboradores [10] describen un sistema de adición lineal a los coeficientes 8x8 de bloques de la DCT de secuencias PN (Pseudo Noise) que previamente han sido adaptadas a las características del sistema humano de visión. Rhoads [11] describe un método en el que se añaden a la imagen N patrones pseudoaleatorios de PN. Cada patrón tiene el mismo tamaño que la imagen y con ello se pueden codificar palabras de N bits.

Bender y colaboradores [12] describen varios métodos, y en particular "patchwork" codifica la marca modificando propiedades estadísticas de la imagen. Consideran la diferencia entre cualquier par de valores aleatorios de una distribución gaussiana con valor medio cero. Este valor medio puede ser desplazado mediante la selección de pares de puntos a los que se incrementa el valor de intensidad a unos y se les decrementa a otros. El espectro de la marca resultante es mayoritariamente de alta frecuencia. Sin embargo, los autores reconocen la importancia de colocar la marca en zonas perceptualmente significativas y por lo tanto modifican la aproximación de forma que se modifican conjuntos de pixels en vez de pixels individuales, con lo que la marca se distribuye en regiones significativas del sistema visual humano.

Paatelma y Borland [13] proponen un método en el que se buscan patrones que aparecen con frecuencia en imágenes y en zonas próximas a estos pixels se introducen modificaciones. Concretamente estos pixels son aquellos que están en una fila con un número predefinido de pixels en la que todos son diferentes de sus vecinos inmediatos. Estos pixels son modificados con un valor predefinido, proporcionando diferencias de intensidad entre éstos y los que no superan un umbral. Con este método se

insertan las marcas en zonas que tienen información de alta frecuencia.

Cox y colaboradores [1] describen un sistema en el que se modifican de forma no lineal los coeficientes de la DCT más significativos. En su método el espectro de la marca se adapta al de la imagen. Los autores destacan que las marcas binarias son menos robustas al ataque por colusión que las basadas en secuencias continuas de ruido pseudo-aleatorio.

O Ruanaidh y colaboradores [14] proponen una aproximación similar en la que la fase de la DFT es modificada. Los autores destacan que la información de fase es más significativa perceptualmente que la magnitud de los coeficientes de Fourier, y por lo tanto defienden que esta aproximación es más robusta a ataques, así como a cambios de contraste en la imagen. La marca insertada es independiente de la imagen.

Smith y Comiskey [15] analizan el proceso de marcado desde una perspectiva de comunicaciones. Proponen una técnica basada en espectro expandido que predistorsiona la marca antes de insertarla. Sin embargo, la marca no depende de la imagen, pero es filtrada previamente basándose en la compresión esperada de algoritmos tipo JPEG.

4.2 Métodos basados en procesado en el dominio del espacio.

Los primeros trabajos propuestos [16] para realizar el proceso de marcado en el dominio del espacio se basaban en la modificación del LSB (Less Significant Bit, Bit Menos Significativo) de los pixels localizados en áreas seleccionadas por secuencias de números pseudo-aleatorias. Estos métodos son muy sencillos de implementar y producen algoritmos muy rápidos, pero cualquier pequeña variación producida por filtros o compresiones altera casi siempre estos bits y por consiguiente elimina la marca. Otras propuestas insertan la marca en los contornos de la imagen o en la textura de la misma pero usando procesado en frecuencia.

Los autores han propuesto un algoritmo que trabaja en el dominio del espacio y utiliza segmentos lineales extraídos de los contornos de la propia imagen para guiar el proceso de inserción de la marca. Los segmentos lineales que se emplean para la codificación de la imagen también son empleados para definir la marca a partir de datos que identifican al autor, al comprador, a una entidad certificadora y un sello de tiempo [17], [18].

4.3 Métodos geométricos

Recientemente se ha propuesto algún algoritmo que se basa en propiedades geométricas de las imágenes. Los algoritmos geométricos se basan en localizar figuras geométricas en las imágenes (segmentos lineales y/o circulares, rombos, cuadrados, etc..) bien

definidas por los contornos de las figuras que aparecen en la imagen, bien mediante la selección de determinados pixels en las mismas (en función de propiedades previamente definidas, como niveles de luminancia elevada en zonas concretas) y el posterior ajuste de estos a figuras geométricas concretas [19]. El algoritmo definido por los autores, anteriormente citado, puede también encuadrarse dentro de estas técnicas, aunque con otro enfoque que el ofrecido en [19]. Como se ha comentado este es un campo que prácticamente acaba de enunciarse.

4.4 Productos comerciales

Ultimamente la irrupción del comercio electrónico en Internet ha creado grandes expectativas a los creadores de diversos materiales (músicos, escritores, artistas en general), dado el público potencial al que puede llegarse ofreciendo sus creaciones a través de Internet. Es por ello que varias empresas han comenzado a ofrecer servicios de marcado para diferentes materiales.

Sin ser exhaustivos pueden citarse las compañías MediaSec, Signum, Argent Q&A, Signafy, Digimarc Corporation, NEC e Informix, Aris Technologies o Alpha Tec que permiten fundamentalmente el marcado de audio y de imágenes fijas.

Cabe citar el trabajo que está realizando IBM para la digitalización de los contenidos de la Biblioteca del Vaticano para facilitar el acceso a los mismos desde Internet [20].

4.5 Proyectos Europeos

A nivel europeo se han desarrollado varios proyectos, centrados principalmente en el mercado de vídeo y en su distribución y monitorización sobre redes de banda ancha, como el proyecto TALISMAN [21]. Otros proyectos relacionados son OKAPI [22] continuación del proyecto ACCOPI [23], IMPRIMATUR [24] y OCTALIS [25].

5. Aplicaciones

La principal aplicación de las marcas de agua es la protección de la propiedad intelectual. La filosofía de Internet se basa en el libre acceso a la información (en general) y en una ausencia de controles. En los últimos años, y debido a la potencialidad comercial que presenta, se está produciendo un cambio en este sentido, motivado por la irrupción del comercio electrónico y las compañías comerciales.

El principal problema planteado radica en la propia naturaleza del material creado digitalmente, ya que una vez producido el original, la reproducción es sumamente sencilla y las copias son idénticas al original e indistinguibles de éste. Es por ello que el valor añadido del material se produce exclusivamente en la creación del original, al igual que sucede en la creación de software.

Para poder utilizar las marcas de agua como método de protección de la propiedad intelectual, debe garantizarse que bien en las marcas, bien en el algoritmo de inserción se identifican de forma única tanto a vendedores como a compradores. Esto puede lograrse empleando técnicas de autenticación y criptografía, pero es necesario que la marca incluya datos relativos a ambos.

Algunos problemas planteados en [26], son la posibilidad de marcar un material ya marcado, no con objeto de atribuirse su autoría, sino para impedir que el autor original pueda demostrar de forma inequívoca su propiedad intelectual, y por otra parte la posibilidad de crear marcas falsas que puedan ser aceptadas como válidas. La solución propuesta también en [26] se basa en el uso de sellos de tiempo u otro tipo de dato que permita asegurar quién fue el que insertó la marca en primer lugar y la definición de métodos que puedan garantizar la unicidad de las marcas.

Además los sistemas de marcado deben tener como requisito que la inserción de marcas sucesivas sobre la misma imagen no produzcan alteraciones sobre las marcas ya existentes, dado que en caso contrario el sobremarcado pasaría a ser un sistema de ataque contra el material protegido.

Paralelamente al incremento de nuevas propuestas y algoritmos para la definición e inserción de marcas, también han aparecido herramientas software que realizan varias operaciones de procesamiento de señal con el objetivo de eliminar marcas de imágenes o hacer que éstas sean irrecuperables o irreconocibles. Dos de los sistemas que realizan este tipo de operaciones son Unzing [27] y Stirmark [28]. Al igual que ocurre con los virus o con los ataques a los sistemas operativos y servidores, estos programas pueden y deben servir para perfeccionar las técnicas y algoritmos diseñados para hacerlos cada vez más rápidos, más sencillos y robustos.

6. Conclusiones

La apropiación indebida de materiales multimedia protegidos por derechos de autor, o su redistribución y/o venta sin permiso supone uno de los mayores problemas para los creadores de contenidos, especialmente si dichos contenidos van a poder ser accedidos desde Internet, que no es una red segura de por sí. Las marcas de agua son consideradas por los implicados de una u otra forma en la protección de la propiedad intelectual como el sistema más adecuado para su uso en Internet. Sin embargo, pese a la gran cantidad de algoritmos y sistemas propuestos este tema no puede considerarse ni mucho menos resuelto.

Es preciso, junto con los esfuerzos que se están realizando desde el campo de la investigación, que se desarrollen sistemas legales que determinen los

límites de su uso, garantizando tanto los derechos de los autores y de los usuarios, así como la generalización de una cultura de reconocimiento de la propiedad intelectual que fomente la adquisición de materiales originales a precios razonables.

El principal valor añadido que aporta la utilización de marcas de agua a los autores es la posibilidad de trazar cada copia generada y así detectar utilizaciones con contempladas en el contrato entre las partes. Pero para que puedan ser válidas como prueba de un delito es necesario que se produzcan cambios legales y que se incluyan en las transacciones a terceras partes confiables.

Finalmente, debemos indicar que las marcas de agua no son técnicas que sirvan para todos los casos, sino una herramienta que utilizada conjuntamente con otras puede reducir el riesgo de apropiación indebida en un determinado contexto. Así la criptografía, la inserción de sellos de tiempo o el registro del material pueden emplearse para complementarla. El objetivo perseguido desde un punto de vista comercial no es lograr un sistema perfecto, sino uno que garantice suficientemente al autor sus derechos de propiedad intelectual y que haga más cara (por las dificultades que plantea eliminar o invalidar las marcas) la falsificación que la compra del original.

Agradecimientos

El trabajo presentado ha sido realizado gracias a la ayuda recibida de la UPV/EHU en el proyecto UPV 147.345-TA119/97.

Referencias

- [1] I. Cox, J. Kilian, T. Leighton, T. Shamonn. "Secure Spread Spectrum Watermarking for Multimedia". IEEE Transactions on Image Processing, vol 6, no. 12, pp. 1673-1687. 1997.
- [2] F. Mintzer, G. W. Braudaway, M. M. Yeung, "Effective and ineffective digital watermarks". Proceedings of ICIP'97, International Conference on Image Processing. Vol III, pp. 9-12. S. Barbara, CA, USA, 26-29 de Octubre 1997.
- [3] R.J. Anderson, F.A.P. Petitcolas. "On the limits of Steganography". IEEE Journal of Selected Areas in Communications, pp 474-481. Mayo 1998.
- [4] G. Caronni, "Assuring ownership rights for digital images", Proc. Reliable IT Systems, VIS'95, Vieweg Publishing Company, 1995.
- [5] K. Tanaka, Y. Nakamura, and K. Matsui, "Embedding secret information into a dithered multi-level image", Proc., 1990 IEEE Military Communications Conference, pp. 216-220, 1990.
- [6] K. Matsui, Y. Tanaka. "Video-Steganography". IMA Intellectual Property Project Proceedings, vol. 1, pp. 187-206, 1994.
- [7] E. Koch, J. Rindfrey, J. Zhao, "Copyright protection for multimedia data". Proc. of the International Conference on Digital Media and Electronic Publishing, 1994.

- [8] A. G. Bors, I. Pitas, "**Image watermarking using DCT domain constraints**". IEEE International Conference on Image Processing, 1996.
- [9] C. T. Hsu, J. L. WU, "**Hidden signatures in images**". IEEE International Conference on Image Processing, 1996.
- [10] M. D. Swanson, B. Zhu, A. H. Tewfik. "**Transparent robust image watermarking**". IEEE International Conference on Image Processing, 1996.
- [11] G. B. Rhoads, "**Identification/authentication coding method and apparatus**". World Intellectual Property Organization, WIPO WO 95/14289. 1995.
- [12] W. Bender, D. Gruhl, N. Morimoto, A. Lu. "**Techniques for data hiding**". IBM Systems Journal 35 (3/4), pp. 313-336, 1996.
- [13] O. Paatelma, R. H. Borland, "**Method and apparatus for manipulating digital data works**", WIPO Patent WO 95/20291, 1995.
- [14] J. J. K. O Ruanaidh, W. J. Dowling, F. Boland. "**Phase watermarking of digital images**". IEEE International Conference on Image Processing, 1996.
- [15] J. R. Smith, B. O. Comiskey. "**Modulation and Information hiding in images**". Information Hiding: First International Workshop Proc. R. Anderson, ed. Vol 1174, Lecture Notes in Computer Science, pp. 207-226. Springer-Verlag. 1996.
- [16] S. Steve Walton. "**Image Authentication for a slippery new age**". Dr. Dobb's Journal, pp 18-26. 1995.
- [17] I. Goirizelaia, J.J. Unzilla, E. Jacob, X. Andiano. "*Seguridad en Internet: Utilización de marcas de agua en imágenes digitales*", Proceedings de las I Jornadas de Ingeniería Telemática, Bilbao, 15-17 Septiembre 1997.
- [18] I. Goirizelaia, J.J. Unzilla, E. Jacob, X. Andiano. "*A new watermarking method using high frequency components to guide the insertion process in the spatial domain*". Proceedings on the 8th International Conference on Computer Analysis of Images and Patterns CAIP'99, Ljubljana, Slovenia. 1-3 September 1999.
- [19] M.J.J.J.B. Maes, C.W.A.M. van Overveld. "*Digital Watermarking by geometric warping*". Proceedings on International Conference on Image Processing ICIP '98. Chicago.USA 1998.
- [20] F. Mintzer, J. Lotspiech, N. Morimoto. "*Safeguarding digital library contents and users*". D-Lib Magazine, Diciembre 1997.
- [21] <http://www.igd.fhg.de/www/igd-a8/projects/talisman/talisman.html>
- [22] <http://www.tele.ucl.ac.be/OKAPI/index.html>
- [23] <http://www.tele.ucl.ac.be/IMAGES/RACE/ACCOPI.html>
- [24] <http://www.imprimatur.alcs.co.uk/>
- [25] <http://www.octalis.com>
- [26] S. Craver, N. Memon, B. Yeo, M. Yeung. "**Can Invisible Watermarks Resolve Rightful Ownerships?**". IBM Research Report. 96-07.1996.
- [27] **Unzing**. <http://altern.org/watermark/>. 1998.
- [28] F. A. P. Petitcolas, R. J. Anderson, M. G. Kuhn. "**Attacks on copyright marking systems**". <http://www.cl.cam.ac.uk/~fapp2/papers/ih98-attacks/>. 1998.

TV Interactiva Terrestre - El Proyecto ACTS iTTi

J.A.Carral^(1,2), F.Scalise⁽³⁾, R.Kopp⁽⁴⁾, L.Lauer⁽⁵⁾, O.Souloumiac⁽⁶⁾

⁽¹⁾ Universidad Politécnica de Madrid, E, ⁽²⁾ Universidad de Alcalá, E, ⁽³⁾ ST Microelectronics, I,
⁽⁴⁾ TDF-C2R, F, ⁽⁵⁾ Philips LEP, F, ⁽⁶⁾ COMATLAS, F

Abstract

The project iTTi (interactive Terrestrial TV integration) was started in March 1998, in the framework of the ACTS Programme. This project aim to specify and develop a new system to handle wireless interactive TV applications. The downstream specification is based on the well-known European DVB-T standard (adopted not only in Europe but world-wide to broadcast digital TV in the UHF/VHF band). The upstream specs are based mainly on a new system (Synchronous Frequency Division Multiple Access, SFDMA), that was preliminary proposed in the framework of the ACTS Interact Project. In this paper the main results of the iTTi project so far are described. Furthermore, an overview of the future plans is also given.

1 Introducción

La TV digital se ha convertido en una realidad tanto desde el punto de vista de la difusión como desde el punto de vista de los descodificadores y la potencia de proceso necesaria para soportar la calidad de vídeo digital.

Gracias a las nuevas técnicas de compresión de vídeo y codificación de canal se han generado normas que han acelerado el desarrollo de nuevos equipos capaces de recibir una gran cantidad de programas vía satélite, red de cable o red terrestre mejorando la calidad de la imagen a la vez que hacen un uso más eficiente del espectro.

En este marco el siguiente paso natural en la evolución de la TV digital es el soporte de servicios interactivos. Esto explica el interés que este tema está despertando tanto en el plano de las discusiones técnicas como entre los consumidores finales.

El consorcio europeo DVB (*Digital Video Broadcasting*) [1] es el organismo central alrededor del cual ha girado el desarrollo de este nuevo paradigma durante los últimos años. Bajo su mano directora y guía se han generado distintas especificaciones para la difusión de TV digital sobre los diferentes medios físicos disponibles. Las normas así desarrolladas para la difusión satélite, por red de cable y Terrestre han alcanzado ya el *status* de “normas europeas (ETSI)” mientras que las referentes al soporte de servicios interactivos se encuentran todavía en distintas fases de madurez.

El mecanismo de funcionamiento es muy simple. El Instituto Europeo de Normalización delega en DVB conjuntamente con la Unión Europea de Radiodifusión (EBU) la tarea del desarrollo de las distintas normas. Estas, una vez alcanzado el consenso necesario, son adoptadas rápidamente en el seno de ETSI. Con ello se ha conseguido que todos los agentes interesados puedan participar en la elaboración de dichas normas a la

vez que se agiliza su desarrollo y se garantiza su interés para la industria.

Entre los distintos medios disponibles para el soporte del canal de retorno inherente a los servicios interactivos se están considerando la red telefónica, GSM, RDSI, Cable y satélite.

1.1 Antecedentes

En el caso particular de la TV digital terrestre se partía de una posición de desventaja ya que el medio físico impone severas condiciones como la multipropagación o la interferencia cocanal.

Ello desembocó en la introducción de una solución *ad-hoc* basada en COFDM (*Coded Orthogonal Frequency Division Multiplexing*). Como toda solución novedosa ha requerido de un fuerte impulso inicial para su despegue lo que, asu vez, ha conllevado un cierto retraso frente a sistemas basados en otros medios como el cable.

Hoy en día, gracias al trabajo llevado a cabo en proyectos de I+D como el proyecto ACTS DVBIRD (*Digital Video Broadcasting Integrated Receiver Decoder*) en el que se definió, diseñó y prototipó un *chip-set* básico para la recepción de TV digital terrestre compatible con la norma DVB-T (con soporte para FFT a 2K y 8K), las experiencias piloto se están multiplicando y su adopción avanza a gran velocidad no solo en Europa sino a escala mundial.

También en el programa ACTS el proyecto VALIDATE (*Verification and Launch of Integrated Digital Advanced Television in Europe*) ha realizado un trabajo fundamental en la verificación de las distintas especificaciones para asegurar que están fundamentadas en bases sólidas a la vez que demostrar sus posibles beneficios.

El principal hito en el desarrollo de la TV digital en Europa ha sido la puesta en marcha, en el Reino Unido, del primer servicio real a finales del año 1998.

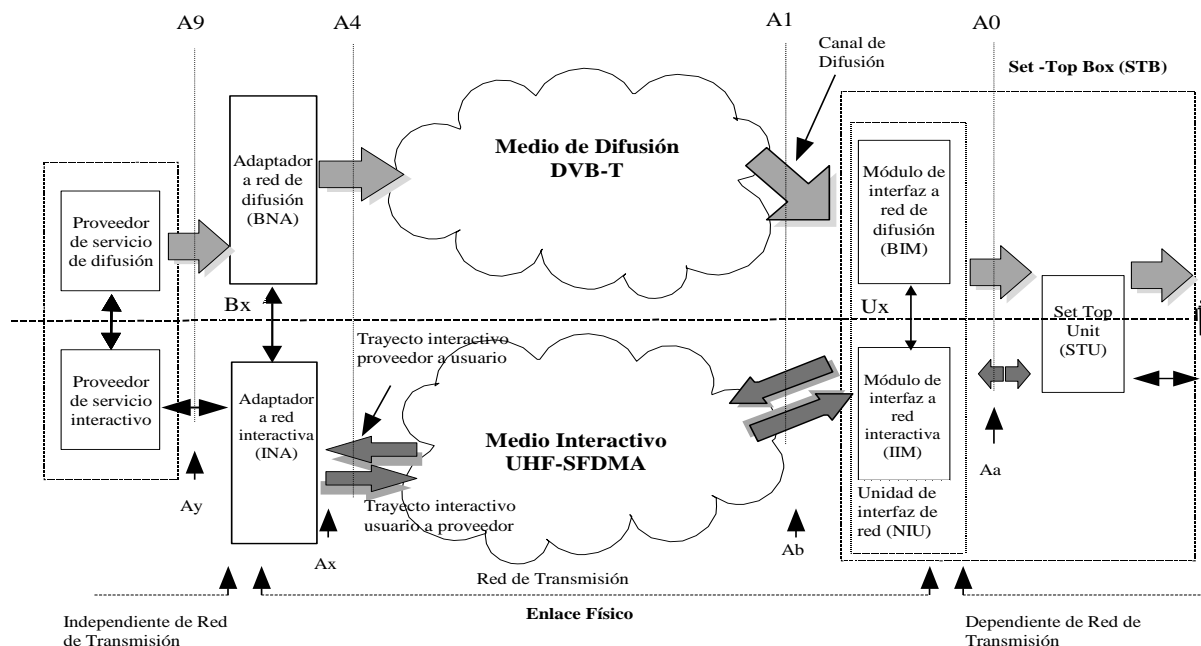


Fig. 1 Modelo general de sistema DVB

Relacionado con interactividad en entorno terrestre, el proyecto ACTS INTERACT (*Interactive Television Return Channel Standardisation and Trials*) especificó un nuevo sistema para el soporte de servicios interactivos asimétricos sobre un canal de retorno UHF terrestre [2], [3].

El proyecto ITTI se inició en Marzo de 1998 con el objetivo de un completo diseño del sistema para la puesta en marcha de servicios interactivos de TV digital terrestre. El plan de trabajo incorpora estudios sobre los aspectos relacionados con servicios interactivos soportados mediante un canal de retorno basado en la técnica SFDMA (UHF terrestre) en su sentido usuario a proveedor y embebido en el canal de difusión (transporte MPEG2) en el sentido servidor a usuario

El objetivo final del proyecto es la demostración práctica mediante un prototipo de servicios interactivos prevista para Septiembre de 1999.

2 El modelo DVB para TV interactiva

El foro DVB ha definido un modelo general de sistema para TV digital interactiva que puede verse en la Fig. 1. Está compuesto por dos canales entre el proveedor de servicios y el usuario final.

- Un canal de difusión unidireccional (desde el servidor al cliente) de gran capacidad sobre el que se envían tanto audio y vídeo como datos. También puede incluir el trayecto interactivo en el sentido servidor a usuario. Está canal está completamente definido en las distintas

especificaciones DVB relativas a difusión (en concreto la especificación para difusión terrena es conocida como DVB-T).

- Un canal interactivo bidireccional entre el proveedor de servicios y el usuario compuesto de dos trayectos:
 - Trayecto usuario a servidor: Es un canal de baja capacidad por el que se transmiten tanto las peticiones como las respuestas a cuestiones hasta el servidor.
 - Trayecto servidor a usuario: Es un canal que transporta datos hasta el usuario y que puede estar embebido en el canal de difusión junto con los programas de TV. También se usa para el transporte de otros datos necesarios para la provisión del servicio interactivo.

En la parte izquierda de la Fig. 1 se encuentra la cabecera de red con el proveedor de servicios. El adaptador a red de difusión (BNA) y el adaptador a red interactiva (INA) proporcionan los interfaces necesarios entre los proveedores de servicio y el medio físico correspondiente.

A la derecha de la figura se sitúa la instalación de usuario donde los interfaces mencionados se identifican como BIM (módulo de interfaz a red de difusión) e IIM (módulo de interfaz a red interactiva) respectivamente.

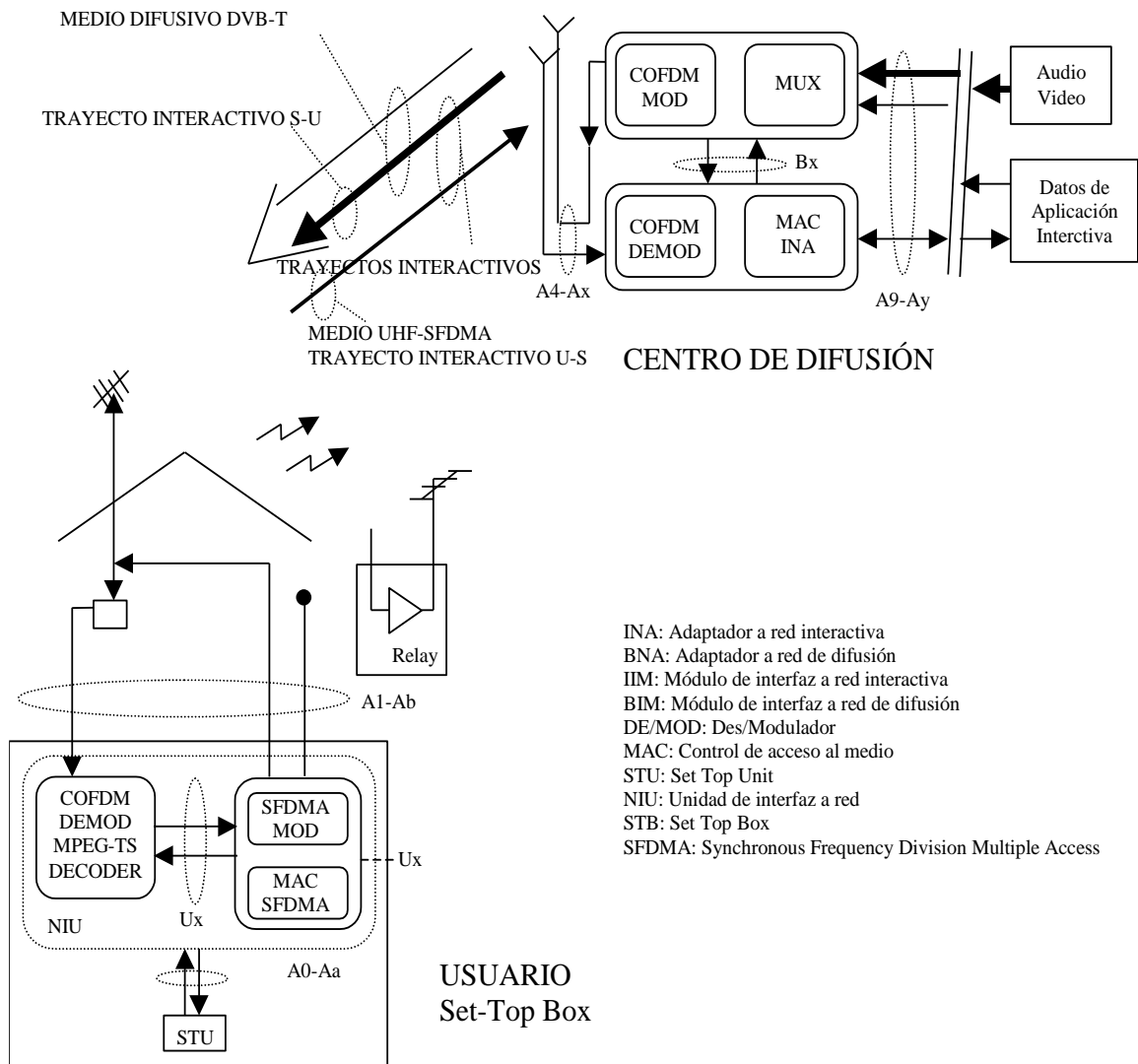


Fig. 2 Modelo del sistema iTTi

Además, debe existir un camino de comunicación entre los proveedores de servicios de difusión e interactivo para permitir la inyección, desde estos últimos, de datos de usuario en el canal de difusión (p. ej. películas en un servicio de vídeo bajo demanda VoD). Este enlace permite también que el proveedor de servicio incluya información de control de aplicación en el canal de difusión.

El enlace entre el INA y el BNA permite que datos referentes al control del MAC se transmitan sobre el canal de difusión. Esto es necesario al no disponer (en el caso de medio terrestre UHF) de un canal interactivo bidireccional.

Partiendo del modelo general comentado anteriormente se ha desarrollado el modelo de sistema iTTi que puede verse en la Fig. 2. y se describe en la siguiente sección.

3 La subred de acceso

3.1 Trayecto de retorno: acceso múltiple basado en SFDMA

La unidad elemental de transmisión en este sistema es una ráfaga SFDMA compuesta de un número especificado de símbolos en una única portadora.

Dependiendo del modo de acceso elegido el MAC del sistema selecciona y asigna la portadora adecuada para la transmisión. Estas portadoras están enganchadas en fase y su modulación sincronizada con una referencia común que se obtiene a partir del canal de difusión. Además, su potencia está regulada desde el INA mediante un mecanismo de bucle cerrado de realimentación. Todo ello permite la reconstrucción en el INA del múltiplex de portadoras sincronizadas.

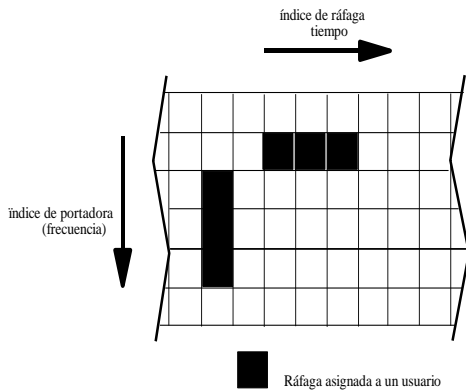


Fig. 3 Esquema de acceso por división en tiempo y frecuencia

El trayecto de retorno constituye así un sistema de multiplexación en dos dimensiones como queda reflejado en la Fig. 3 y la referencia [4].

Si un mensaje supera la longitud de una ráfaga puede ser transmitido en una secuencia de ráfagas contigua (bien en el tiempo, bien en frecuencia hasta un máximo de 32) siempre de acuerdo al modo de trabajo del MAC y la velocidad requerida.

3.1.1 Modos de transmisión

Actualmente se han definido cuatro modos de transmisión diferentes con el objetivo de facilitar la adaptación a distintas arquitecturas de red (en particular a sistemas celulares). En la *Tabla 1* se señalan los principales parámetros de cada uno de ellos.

Tabla 1. Parámetros básicos de los distintos modos de transmisión

	Modo 0	Modo 1	Modo 2	Modo 3
Ancho de banda	1 MHz	1 MHz	1 MHz	1 MHz
Sep. entre portadoras	500 Hz	1 kHz	4 kHz	16 kHz
Nº de portadoras	2000	1000	250	62
Duración de símbolo	2.5 ms	1.25 ms	0.312 ms	0.0781 ms
Duración de ráfaga	500 ms	250 ms	62.5 ms	15.625 ms
Modulación	$\pi/4$ DQPSK or $\pi/8$ D8PSK	$\pi/4$ DQPSK or $\pi/8$ D8PSK	$\pi/4$ DQPSK or $\pi/8$ D8PSK	$\pi/4$ DQPSK or $\pi/8$ D8PSK
Código	1/2, 2/3 or 3/4	1/2, 2/3 or 3/4	1/2, 2/3 or 3/4	1/2, 2/3 or 3/4
Radio de celda	70 km	40 km	5 km	2 km

3.2 Control de Acceso al Medio (MAC)

El MAC se encarga de gestionar el acceso concurrente de los usuarios al medio compartido (en

este caso red de acceso por UHF) situado entre los interfaces Ab y Ax en la Fig. 1.

Su principal labor consiste en controlar eficientemente el acceso al canal de retorno para asignar la capacidad disponible a un número potencial de usuarios tremendamente elevado.

Por tanto, el MAC debe resolver dos puntos básicos: la estrategia de multiplexación (FDMA/TDMA o lo es que es lo mismo SFDMA) y la asignación de recursos en ese doble eje tiempo/frecuencia.

3.2.1 Modos de acceso

El problema a resolver es el acceso de un gran número de usuarios (STBs) sobre un medio único, el canal de retorno.

Se han definido tres modos de trabajo para la transmisión de datos sobre el trayecto de retorno: modo paquete o sin conexión, modo conectado y modo controlado.

El modo paquete está definido como un protocolo sin conexión de acceso basado en demanda. Comienza con una petición según un esquema de acceso aleatorio que en caso de fructificar obtiene la asignación deseada desde el INA.

En el modo conectado se dispone de recursos reservados. Estos enlaces o conexiones están caracterizados por su capacidad y se establecen previamente a su utilización y se liberan una vez utilizados durante el tiempo necesario. Todo ello bajo el control del INA.

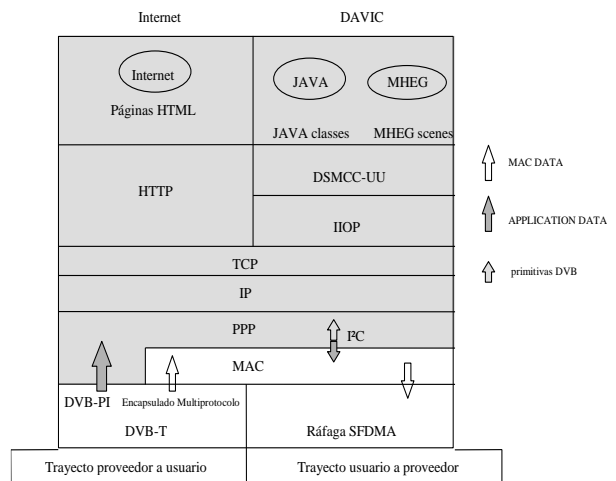


Fig. 4 Pila de protocolos en el STB

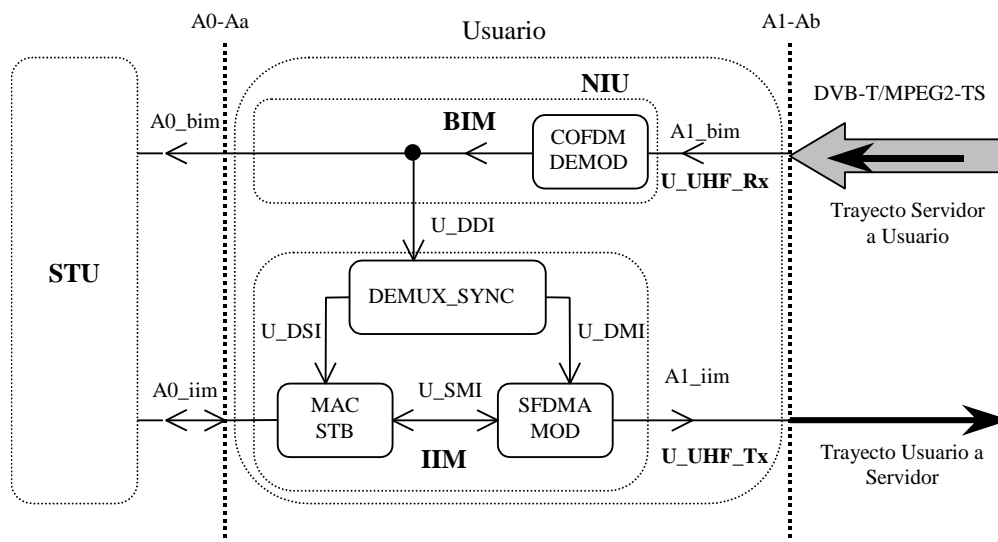


Fig. 5 Diagrama de bloques de un STB

El MAC también ofrece una técnica de sondeo mejorada que combina peticiones *multicast* con asignación de recursos determinista para la recolección de respuestas de un grupo de usuarios. A este modo de trabajo se le denomina “modo controlado”.

3.3 La pila de protocolos

DVB define un modelo de pila de protocolos simple constituida por solo tres capas (física, transporte y aplicación). Pero solo especifica los protocolos correspondiente a las dos capas inferiores dejando la última abierta a la libre competencia de mercado.

En la Fig. 4 puede verse la pila de protocolos correspondiente a un usuario (STB). La capa física está compuesta por el proceso de señal básico y los entramados correspondiente a DVB-T y SFDMA (para los trayectos proveedor a usuario y viceversa) respectivamente. Sobre ella, el protocolo que se ocupa de la cooperación entre ambos trayectos y los protocolos típicos de red y transporte usados para servicios interactivos (tanto los de tipo Internet como los relacionados estrechamente con programas de difusión). En la referencia [5] puede encontrarse una descripción más detallada de dichos protocolos.

3.4 El equipo de usuario (STB)

La Fig. 5 muestra el diagrama de bloques de un STB propuesto por el Consorcio iTTi.

Tal y como fue definido por DVB, un STB puede subdividirse en tres bloques principales: el bloque de interfaz al canal de difusión (BIM), el bloque de interfaz al canal de retorno (IIM) que incluye a SFDMA y el *Set-Top Unit* que incorpora las funciones comunes (protocolos de transporte hacia arriba).

3.4.1 Trayecto proveedor a usuario: DVB-T

Este trayecto tal y como ha sido definido por iTTi es completamente compatible con la norma europea de difusión de TV digital terrestre DVB-T [1].

La principal razón que ha apoyado esta decisión es la posibilidad de definir los servicios interactivos proporcionados por iTTi como una mejora sobre la capacidad básica de difusión proporcionada por DVB-T. De esta manera, se produce una realimentación mutua que esperamos beneficiará a ambos facilitando su rápida adopción.

El trayecto proveedor a usuario está embebido en el canal de difusión, los datos se encapsulan en paquetes del flujo MPEG. Estos datos pueden corresponder a tanto a la aplicación interactiva como al propio control del canal o el servicio interactivo en concreto.

Los datos referentes al control del canal (MAC) se inyectan en el flujo utilizando un identificador específico definido en la norma ETR 162 [6] mientras que los demás lo hacen mediante el encapsulado multiprotocolo definido en [7].

3.5 La cabecera de difusión

La Fig. 6 muestra el diagrama de bloques correspondiente a la cabecera de difusión. A continuación se incluye una breve descripción de los principales bloques que la componen.

El bloque RE-MUX se encarga de la inserción de los datos de control del MAC en el flujo MPEG2.

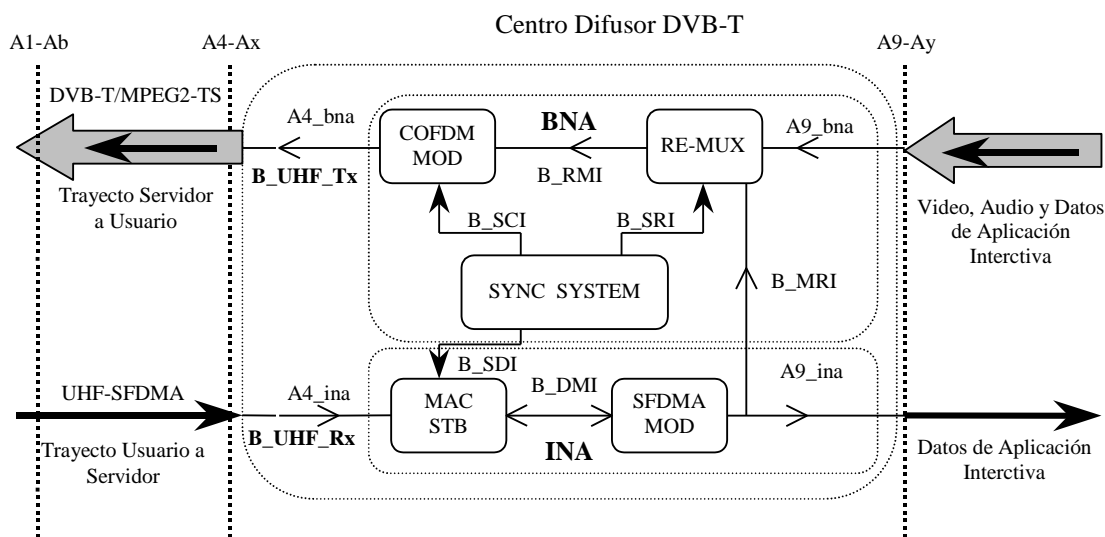


Fig. 6 Diagrama de bloques del centro de difusión

Los índices correspondientes a esos datos deben haber sido previamente incluidos en el flujo (tabla SI).

El bloque SYNC SYSTEM se encarga de distribuir la señal común de sincronización a los demás elementos del sistema basado en un reloj de 10 MHz y un pulso recibido mediante GPS.

El demodulador SFDMA entrega un flujo de 2 Mbps por MHz de ancho de banda del canal de retorno.

Finalmente, el bloque MAC es el controlador del INA y se encarga de gestionar el canal de retorno asumiendo la interfaz con la red de transporte para entregar los datos interactivos al proveedor de servicio.

4 Objetivos del proyecto iTTi

Este proyecto está directamente relacionado con tres aspectos (ligados entre sí) de un sistema digital terreno: los servicios y la arquitectura de red, las aplicaciones y el terminal de usuario digital e interactivo (DIT-STB). Sus principales objetivos son:

- Definir las especificaciones y la arquitectura de un terminal de usuario (STB) para servicios interactivos de TV digital terrena. Para ello, se tomarán en consideración tanto los trabajos previos relacionados en la sección 1.1 como los requisitos de usuario. Se analizarán en detalle los problemas relacionados con la interfaz entre los canales de difusión e interactivo haciendo especial hincapié en los aspectos de sincronización, optimización de la arquitectura e integración final del conjunto.

- Estudio de las implicaciones prácticas que puede haber sobre la transmisión DVB-T como pueden ser la multiplexación de flujos (interactivo y difusivo) y la arquitectura de red.
- Estudio de los protocolos de alto nivel y su posible optimización para el caso del canal de retorno SFDMA (baja capacidad).
- Estudio del impacto que este sistema puede suponer para las aplicaciones interactivas (tiempos de respuesta, baja capacidad, precarga...) con el objetivo de identificar puntos clave para el diseño de las mismas.
- Optimización de la escala de integración del sistema final mediante el estudio y modelado de nuevos ASICs.
- Implementación de un demostrador integrado que incluya al MAC, los protocolos de alto nivel y las aplicaciones.
- Diseño de una serie de experiencias de campo que incluyan difusión DVB-T, difusión de varios programas analógicos y canal de retorno SFDMA.
- Realización de Experiencias sobre arquitectura de red RF.

El proyecto tiene prevista su finalización en Septiembre de 1999 con la implementación de un demostrador público que permita analizar la viabilidad del sistema en condiciones realistas.

Conclusiones

El proyecto iTTi ha sido concebido con el fin de analizar la viabilidad de un sistema de TV digital interactiva operando en la banda UHF sin necesidad

de una conexión de retorno soportado sobre red telefónica, cable u otro medio distinto al propio medio difusivo.

Partiendo del trabajo realizado en el proyecto INTERACT se generarán nuevas y mejoras especificaciones del sistema así como una estimación de complejidad y de coste del terminal de usuario. Esto permitirá cuantificar el impacto que sobre el mercado de consumo pueda tener esta solución.

La relación directa que en todo momento se mantiene con el foro DVB asegurará tanto un amplio consenso sobre dicho sistema como la bondad de la solución técnica aportada.

El sistema iTTi será demostrado en el próximo congreso IBC'99 que se celebrará en Amsterdam en Septiembre de 1999.

Agradecimientos

Los autores de este trabajo quieren agradecer a los demás participantes en el consorcio que han contribuido y continúan haciéndolo a su adecuada finalización y a D. Francisco Guirao (oficial de responsable del proyecto representante de la Comisión Europea) por su inestimable ayuda y guía a lo largo del mismo.

Participan en el consorcio ITTI: ST Microelectronics (I) como coordinador, Politecnico di Torino (I), CCETT (F), TDF-C2R (F), COMATLAS (F), ITIS (F), Philips LEP (F), CNET (F), ERA Tech (UK), OWL (UK), TBS (F), RTE (IE) y DIT-UPM (E).

Referencias

- [1] DVB Project – ETSI , “Digital Broadcasting System for Television, Sound and Data Services; Framing Structure, Channel Coding and Modulation for Digital Terrestrial Television” ETS 300 744 standard, September 1996.
- [2] DVB Project, “Interim Specification for DVB Interaction Channel for Terrestrial Systems based on SFDMA” - version 3.3, doc. TM1894, June 1997.
- [3] R.Allan, D.Callonnec, P.Gardiner, P.Kasser “Performance and System Capacity of the SFDMA System UHF Interaction Return Channel” Proc. of IBC98, September 1998, Amsterdam (The Netherlands).
- [4] ITTi WP2A1 Partners “Draft Specification on the Frequency and Time Synchronisation and on Specific MAC Related Packets” Deliverable D04-partA, July 1998.

- [5] iTTi WP3 Partners “Review of Existing Work and Standards relating to Interactive Applications & Demonstrator and Set-Top Box Requirements” Deliverable D11, August 1998.
- [6] DVB Project, “Digital broadcasting systems for television, sound and data services; Allocation of Service Information (SI) codes for Digital Video Broadcasting (DVB) systems”, ETR 162, Octubre 1995.
- [7] J. Heinanen, "Multiprotocol Encapsulation over ATM Adaptation Layer 5", RFC1483, 07/20/93.

Capítulo 4A: Tecnologías de Agentes y Redes Activas

Planificación en Internet basada en una Arquitectura Multiagente.

D. Camacho Fernández, J.M. Molina López, D. Borrajo Millán (U. Carlos III de Madrid)

JavaMAST: Un Entorno de Desarrollo de Agentes Inteligentes.

M. García, M. Garijo, C.A. Iglesias, E. Martín, J.L. Mulas (U. Politécnica de Madrid)

Gestión de recursos en conmutadores mediante agentes inteligentes móviles.

A. Barba Martí (U. Politécnica de Cataluña)

Principios y Aplicaciones de las Redes Activas.

M. Calderón Pastor, M. Sedano Ruiz, S. Eibe García (U. Politécnica de Madrid)

Especificación y verificación formal de un protocolo de gestión de sistemas de administración de redes distribuido con arbitro.

M. Marrero y A. Suarez (U. Las Palmas)

Dimensionado del Nodo de Acceso con Control Explícito de Tasa DMRCA para Redes de Paquetes.

X. Hesselbach Serra, S. Sallent Ribes (U. Politécnica Cataluña)

Planificación en Internet basada en una Arquitectura Multiagente

David Camacho Fernández

José Manuel Molina López

Daniel Borrajo Millán

Departamento de Informática. Universidad Carlos III de Madrid

Avda. de la Universidad 30, 28911 Leganés (Madrid)

Telf: 91-6249416, Fax: 91-6249129

E-mail: {dcamacho,molina,dborrajo}@ia.uc3m.es

Abstract

In the last years, the amount of information stored in Internet has grown exponentially. This article presents a new approach to cooperative problem solving that uses the Web as a source of data. The architecture shown has been designed using Artificial Intelligence techniques: Multiagent System design, and classical resolution problem solving techniques, like planning. Both techniques are used to obtain a new architecture that is tested with knowledge obtained from Internet dynamically. The architecture has two different types of agents: planning agents and web agents. Planning agents pay attention to the user's queries and solve his problems, web agents obtain the required information from Internet.

1. Introducción

En la última década, Internet y especialmente la World Wide Web (WWW, Web), se ha convertido en un gigantesco almacén de información disponible para cualquier usuario conectado a la red. Esto ha originado diversos problemas. Uno de los más importantes es el crecimiento exponencial de la cantidad de información almacenada en la red. Este problema es tan grave, que cualquier proceso de obtención de información almacenada en la Web necesita ser filtrado, ordenado o planificado de alguna manera, si se desean obtener resultados satisfactorios.

Actualmente, se dispone de multitud de técnicas que pretenden lograr una extracción, manipulación o aprovechamiento eficiente de esa información, aunque la mayoría de estos sistemas se centran principalmente, en la cantidad de información recuperada [6]. En particular, una de estas disciplinas, denominada Inteligencia Artificial Distribuida (IAD) [7], que tradicionalmente se ha centrado en el estudio y diseño de sistemas semiautónomos capaces de interactuar con otros sistemas y con el mundo exterior, está utilizando la Web como un banco de pruebas donde realizar experimentos complejos [8]. Una de las áreas de mayor éxito dentro de este campo ha sido la de los Sistemas Multiagente (SMA).

La mayoría de los intentos para aproximar los SMA a la Web [6], pueden resumirse en la creación de "sofbots", "spiders", o cualquier otro tipo de sistema capaz de recolectar y filtrar la ingente cantidad de información disponible en la Web.

Trabajos como los realizados por Knoblock et. al., en sistemas como SIMS [9], dan una muestra de los intentos de integrar fuentes de información heterogéneas con el fin de manipular esos datos para

alcanzar nuevas soluciones. O sistemas como ARIADNE [2], que tratan de aplicar conocimiento heterogéneo (obtenido de distintas fuentes, incluida la Web), para aplicarlo a un proceso de planificación.

El hecho de que los datos que se están utilizando para resolver un problema se vean modificados, hace que todo el proceso de resolución del problema sea dinámico, puesto que una solución planteada para resolver un problema en un determinado instante, puede no ser válida pasado un determinado período de tiempo. Esto implica que las técnicas más tradicionales de resolución de problemas, donde la cantidad de información (y el contenido de la misma) es algo que no cambia en el tiempo, no van a poder ser aplicables directamente.

Otro de los problemas importantes que se plantea, viene determinado por el lugar desde donde se adquieren los datos (Web) y por su posterior manipulación. Como se describirá en posteriores apartados, parte de esos datos serán suministrados automáticamente desde Internet, con todos los problemas tecnológicos derivados de ese hecho.

En este artículo se presenta una aproximación Multiagente a la resolución distribuida y cooperativa de problemas en entornos dinámicos. En particular, se ha escogido como área de trabajo la resolución de problemas mediante la planificación en el dominio del turismo. La solución mostrada, trata de utilizar técnicas clásicas de resolución de problemas, como la planificación, junto a las ventajas que proporcionan los sistemas distribuidos de resolución de problemas como los SMA. Todo ello en un marco dinámico como Internet.

La organización del artículo es la siguiente: en el apartado 2 se describe la arquitectura del SMA propuesto; en el apartado 3 se describe un ejemplo de aplicación del SMA sobre la Web; y finalmente, en el apartado 4 se exponen las conclusiones del artículo.

2. Una Arquitectura Multiagente para resolución de problemas en Web

Uno de los principales problemas de los SMA sobre Web vistos en el punto anterior se reduce a la forma de trabajo de esos sistemas. En realidad, estos sistemas son básicamente aplicaciones software que aplican filtros de mayor o menor complejidad sobre las respuestas obtenidas por buscadores en Web [6], o bien sistemas donde se presupone que la información es homogénea y estable [11]. Es decir, hasta la fecha, la mayoría de los esfuerzos se han centrado en el análisis de la información devuelta, más que en su posible reutilización.

En este trabajo, se pretende diseñar una arquitectura multiagente orientada principalmente a la resolución concurrente y cooperativa de problemas. Se ha escogido como lugar de trabajo la Web por considerar que se trata de un dominio fuertemente dinámico en varios aspectos.

- El número de nodos (o enlaces) a los que se puede acceder pueden aumentar, debido a que aparezcan nuevos enlaces que no existían previamente y cuya información sea útil para resolver el problema.
- Enlaces que ya se utilizaban para resolver un problema puede que no sean accesibles en un instante de tiempo o que hayan desaparecido.
- La información proporcionada por un determinado enlace puede verse modificada, tanto en contenidos como en su estructura.

2.1. Estructura General de la Arquitectura

En la Fig.1, se da una descripción gráfica de la arquitectura propuesta. El sistema está formado por dos tipos de agentes: los agentes de planificación (*planning agents*) y los agentes Web (*Web agents*). Los primeros son los encargados de atender a los diferentes usuarios, así como de resolver los problemas que les sean planteados. Los agentes Web son los encargados de comunicarse con Internet y obtener la información necesaria para la resolución del problema. Como puede verse en la Fig.1, los agentes de planificación pueden comunicarse entre ellos o con otros agentes Web. La única forma de acceder a la información almacenada en la Web es a través de los agentes Web. Cada usuario dispondría de un agente encargado de atenderle (*planning agent*). Los agentes se encargan de responder a las posibles solicitudes de otros agentes, mientras atienden a las posibles solicitudes que les realicen los usuarios. Por lo tanto, deben estar resolviendo concurrentemente sus propios problemas, mientras están ayudando (cooperando) a otros agentes a resolver las cuestiones (*queries*) que le planteen. Las flechas punteadas hacen referencia a la capacidad que poseerán los agentes Web para conectarse directamente con fuentes de información como la Web. Las flechas sólidas indican que los diferentes agentes estarán en comunicación mediante algún

tipo de protocolo. El esquema general del agente pretende indicar que la *query* realizada por un usuario, junto con su perfil (que le caracteriza como individuo) puede hacer cambiar el comportamiento o funcionamiento del agente encargado de atenderle.

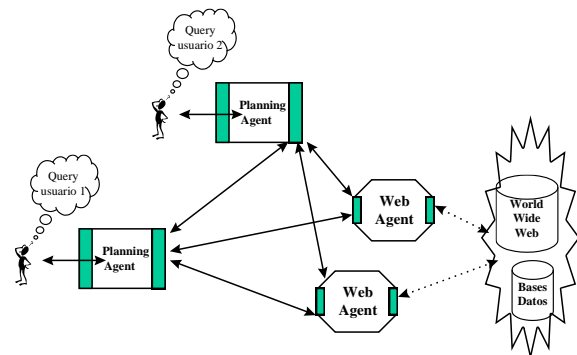


Fig.1.- Arquitectura del SMA.

Esta arquitectura comprende la mayoría de los aspectos básicos de cualquier arquitectura multiagente, aunque características como la movilidad o la seguridad del SMA se han relajado, para tratar de prestar mayor atención sobre otros aspectos como la capacidad de razonamiento y la cooperación de los agentes. Además, la arquitectura debe ser lo suficientemente flexible como para permitir una máxima ocupación del agente en la resolución de problemas.

Como característica innovadora se ha añadido la capacidad de caracterización automática del usuario que está siendo atendido por el agente así como de los otros agentes. Esto redundará en una mayor eficiencia y calidad en la respuesta que el sistema va a proporcionar al usuario. Este objetivo se logrará creando diferentes tipos de perfiles del usuario. Por un lado, se puede caracterizar al usuario a través de un formulario que él mismo puede completar, comúnmente denominado perfil o *profile*. Por otro lado, se puede caracterizar al usuario automáticamente, observando las consultas que ha realizado y de la respuesta ante la solución, o soluciones, mostrada por el sistema.

2.2 Localización del conocimiento necesario para la resolución de un problema

Se considera la existencia de tres fuentes diferentes de información, desde las que se puede adquirir el conocimiento que el agente necesita para resolver los problemas planteados.

La primera de ellas se encuentra en el entorno del propio agente. Se puede solicitar información a otros agentes, mediante la generación de preguntas o *queries*. Estas pueden estar dirigidas a todos los agentes o a algún agente en particular (debido a que su capacidad para la resolución de ese tipo de problemas sea conocida). La petición/recepción de *queries* se pretende desarrollar utilizando algún protocolo similar a KQML (Knowledge and Query Manipulation Language) [4], ya que este lenguaje proporciona un marco general para la comunicación y cooperación en SMA. En particular KQML proporciona un conjunto de protocolos para la identificación, establecimiento de conexión, e intercambio de mensajes entre agentes. Gracias a estos protocolos se puede solicitar que un problema sea resuelto y obtener una respuesta completa a la pregunta formulada (en principio, y por simplicidad, se puede suponer que las *queries* son dirigidas a todos los agentes del sistema y que el agente que solicitó la ayuda gestiona las diferentes respuestas obtenidas).

La segunda fuente de información es el propio agente, es decir, se trata de reutilizar la experiencia aprendida por el agente encargado de atenderle en antiguos casos que tuvo que resolver, utilizando para ello técnicas de razonamiento basado en casos [1]. Inicialmente se almacenarán aquellas respuestas que tuvieron éxito y que finalmente han sido aprobadas por el usuario. Se supone que deben almacenarse las respuestas completas, aunque es obvio que deberá realizarse un análisis de los resultados, para determinar qué partes de los mismos son las realmente relevantes.

La tercera y última fuente que se pretende utilizar para extraer información es la propia Internet. Cuando cualquier sistema pretende utilizar Internet como fuente de datos, aparecen multitud de problemas. Algunos de los problemas que se plantean son:

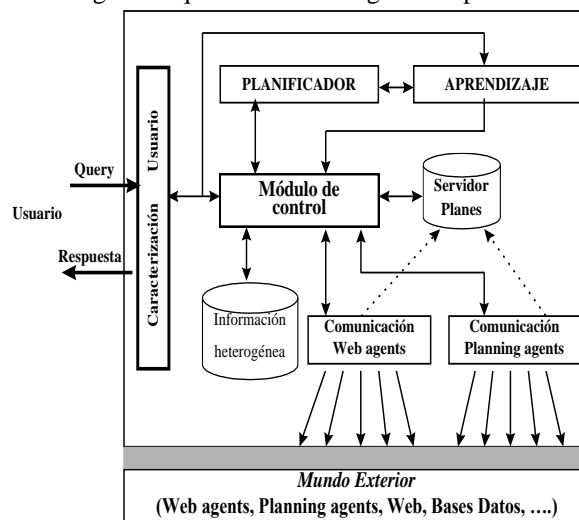
- Gestionar la inmensa cantidad de información que es accesible en un momento dado en Internet, teniendo en cuenta que esa información es fuertemente dinámica y que tanto sus contenidos como estructura se ven modificada en el tiempo.
- Caracterizar o estructurar las fuentes de información. Como aparece descrito en [3], la mayoría de las fuentes de datos en Internet son normalmente semiestructuradas. Habitualmente se trata de formularios que deben ser completados por el usuario y cuya información devuelta es totalmente heterogénea.
- Filtrar adecuadamente la información devuelta y obtener una representación útil para los agentes.

Entre otros, se pueden ver los intentos de construir Mediadores para la integración de fuentes heterogéneas de información [9], y Wrappers [3] para tratar de hacer accesible fuentes de información estructurada y semiestructurada.

2.3. Arquitectura de un Agente de Planificación

En la Fig.2 se muestra una descripción modular, de la arquitectura de un agente de planificación del sistema.

Fig.2.- Arquitectura de un agente de planificación



Se pueden observar algunas características interesantes que presentan los agentes como:

- Módulo de caracterización del usuario. Permite realizar un filtrado de las queries planteadas por el usuario, y junto con la información que se dispone de él, realizar una resolución del problema adaptada a sus necesidades.
- Acceso automático a Internet. Se trata de un módulo que permite automatizar todo el proceso de comunicación con los Web agents.
- Comunicación con otros agentes de planificación. Implementan el protocolo de comunicación entre los distintos agentes de planificación del SMA.
- Almacenamiento (información heterogénea) de datos útiles para el agente. Se trata de almacenar, tanto soluciones antiguas que puedan resultar de utilidad para futuras consultas, u otros tipos de información que pueda necesitar el agente, como información de distintos usuarios, o características de otros agentes.
- Servidor de planes. Se suministra el plan o subplan que se está tratando de resolver en un determinado instante.
- Un módulo de control encargado de gestionar el correcto funcionamiento de los diferentes módulos de los que está formado el agente. Entre las principales funcionalidades que debe poder desarrollar este módulo, se destacarán:
 - La inserción en una lista de las soluciones parciales alcanzadas hasta el momento, que deberán ser validadas con la información que se pueda adquirir desde otros agentes del medio o desde otros medios de información más heterogéneos.
 - La creación de una agenda que permita atender las diferentes preguntas de otros agentes.

- La gestión de posibles respuestas obtenidas a preguntas que haya realizado cada agente.
- La respuesta (con mayor o menor grado de precisión) que debe ser dada al usuario.
- La planificación de las propias tareas que deben ser realizadas por el agente para la obtención de la solución óptima (desde el punto de vista de las necesidades del usuario), o de la primera solución encontrada (minimizar el tiempo de espera).
- Módulos de aprendizaje, que realizan las tareas de la caracterización del usuario. Pueden modificar el comportamiento del sistema en función de la respuesta observada por el usuario ante las soluciones mostradas.
- Módulo de planificación, es el encargado de la resolución del problema planteado por el usuario. Dada su importancia, se detalla su funcionamiento en el próximo punto.

2.4. Módulo de Planificación

En la arquitectura mostrada, se empleará un planificador (se ha comenzado trabajando sobre el planificador no lineal PRODIGY 4.0 [10]) como herramienta principal de razonamiento. Pretende utilizarse la potencia que proporciona el planificador para realizar un primer análisis del problema. A partir de ese análisis se obtiene una descripción muy genérica de la solución al problema. De hecho, es posible obtener por parte del planificador múltiples soluciones del problema. Estas soluciones genéricas son fácilmente reordenables en función de alguna heurística o función de coste.

En la Fig.3, aparece una representación más detallada del módulo de planificación.

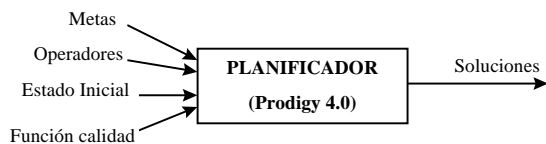


Fig.3.- Módulo de planificación.

En la Fig.4 se muestra cómo ha resuelto PRODIGY el problema de transportar a una persona desde una estación de trenes en una determinada ciudad (*trst0*), a la estación de trenes de otra ciudad diferente (*trst2*), se supone que, inicialmente el tren, *train2*, estaba en la estación de trenes, *trst2*. La idea consiste en que se define un dominio genérico formado por una serie de operadores (transformaciones entre estados) con el que el planificador puede resolver, a un nivel muy abstracto, el problema planteado. Una vez resuelto ese problema, el módulo de control toma (en un determinado orden) esas soluciones (o solución si sólo existe una) y trata de validar alguna o todas las soluciones posibles (inicialmente se ha considerado la solución óptima dada por el planificador).

```
1.-Solution:
take-train train2 trst2 trst0
```

```
geton-train persona train2 trst0
take-train train2 trst0 trst2
getoff-train persona train2 trst2
```

```
2.-Solution:
bus-trainst-to-airp bus0 trst0 airp0 city0
geton-airplane persona airplane0 airp0
take-airplane airplane0 airp0 airp2
getoff-airplane persona airplane0 airp2
bus-airp-to-trainst bus2 airp2 trst2 city2
```

```
3.-Solution:
take-train train2 trst2 trst0
geton-train persona train2 trst0
take-train train2 trst0 trst1
take-train train2 trst1 trst2
getoff-train persona train2 trst2
```

Fig.4- Posibles soluciones dadas por PRODIGY.

Como puede verse en la Fig.4, la respuesta del planificador se devuelve en un lenguaje propio (en este caso en el de PRODIGY), y por lo tanto es necesario una transformación posterior para que otras partes del agente puedan trabajar sobre la solución o soluciones proporcionadas por el mismo.

Como puede verse en los ejemplos mostrados, se puede utilizar la solución general dada por el planificador para descomponerla en un conjunto de subproblemas más simples. Por ejemplo, en la Fig.5 se muestran los dos operadores dados como parte de la respuesta del planificador, que se convertirán en *queries* o metas a conseguir, por el agente, o por otros. En este ejemplo en particular, se necesita saber si existe una conexión de tren entre la estación de trenes de la ciudad 0 y de la ciudad 2 (para la primera pregunta), o si existe un vuelo de avión entre los aeropuertos de ambas ciudades para la segunda. Esa información podría ser obtenida por el agente desde la Web o ser proporcionada por otros agentes.

```
take-train train2 trst0 trst2
take-airplane airplane0 airp0 airp2
```

Fig.5.- Operadores que deben transformarse en queries.

2.5. Agentes Web

Los agentes Web (Web Agents) de la arquitectura, se encargan de realizar todo el proceso de acceso automático, recolección y filtrado de la información almacenada en la Web. En la Fig.6, se muestra una descripción modular de uno de estos agentes.

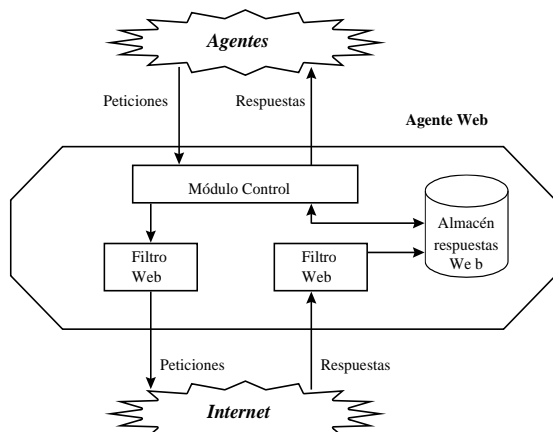


Fig.6.- Agente Web.

Un agente Web se encarga de atender las múltiples peticiones que recibe de otros agentes del sistema (módulo de control), transformando adecuadamente las diferentes peticiones en *queries* que puedan ser reenviadas a la Web (módulo de filtrado hacia la Web). Las respuestas recibidas desde la Web, serán convenientemente filtradas y almacenadas en una representación útil para los agentes de planificación de la arquitectura (módulo de filtrado de respuestas Web). Dado que un determinado agente Web puede conocer múltiples lugares a los que acceder para encontrar información a las peticiones planteadas, y que puede recibir múltiples respuestas de cada uno de estos sitios, se necesita un módulo que permita almacenar todas esas respuestas hasta que puedan ser reenviadas a los peticionarios (módulo de almacenamiento).

3. Aplicación a la planificación de viajes

En este apartado se muestra una descripción detallada de la representación del dominio considerado, así como un ejemplo de funcionamiento del sistema.

3.1. Planificación de un viaje

El problema planteado consiste en diseñar un Agente de Viajes Inteligente. El agente debe ser capaz de gestionar la petición por parte de un usuario de la planificación de un viaje. En la Fig.7 puede verse cómo se ha representado, a nivel gráfico, la existencia de diferentes ciudades, medios de transporte, alojamientos, etc.

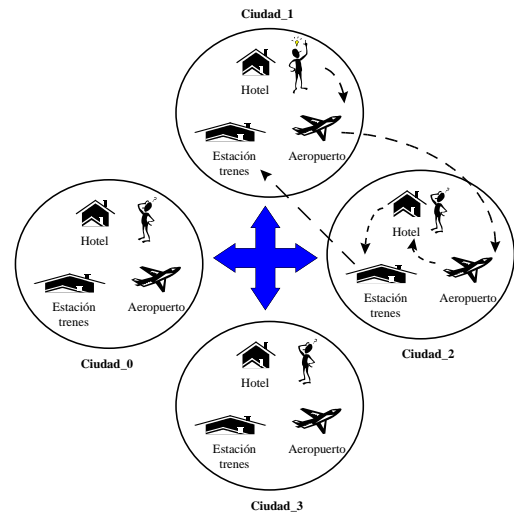


Fig.7.- Descripción gráfica del problema.

El problema a resolver consiste en gestionar los diferentes aspectos que pueden aparecer a la hora de realizar un viaje. Los aspectos más relevantes que se han considerado son; el desplazamiento desde el lugar de origen al de destino; la estancia en el lugar de destino; y finalmente la posibilidad de desplazarse en el lugar de destino.

Para tratar el primer aspecto, se ha considerado que el usuario va a desplazarse utilizando el avión como medio de transporte principal. La generalización a otros medios de transporte como trenes, autobuses o barcos se realizarán más adelante. Dentro del proceso de resolución del problema, se deben tener en cuenta aspectos como que un determinado individuo puede necesitar tomar un tren (local), autobús o taxi para poder desplazarse hacia el aeropuerto; es decir, el aspecto de las comunicaciones internas de la ciudad es un aspecto importante y complejo dada la cantidad de posibilidades que pueden aparecer. También se debe tener en cuenta que el viaje pueda hacerse con escalas o sin ellas (este hecho podría implicar serias restricciones).

Se debe tener en cuenta que pueden presentarse múltiples soluciones al problema planteado, pudiendo ordenarse en función de diferentes parámetros. Por ejemplo, se podrían mostrar las soluciones obtenidas en función del coste económico de la solución, de la cantidad de tiempo para completar el viaje, o de las preferencias del usuario (por ejemplo, que no desee utilizar el avión para llegar a su destino).

El problema se complica si se tiene en cuenta que las otras dos características del problema no son totalmente independientes, puesto que es frecuente que a pesar de tener confirmado uno o varios vuelos, la confirmación final de cualquier viaje no sea posible debido a una falta de alojamiento en el lugar de destino.

Por lo tanto, se pretende diseñar un sistema que sea sensible a la propia naturaleza de algunos problemas

que aparecen en la planificación de un viaje. Especialmente, aquellos donde la información es dinámica (debido a que la disponibilidad de vuelos o de plazas hoteleras puede cambiar en cada momento), y donde las fuentes de las que se pueden obtener la información son muy heterogéneas (bases de datos, Internet u otros agentes con su propia experiencia).

También se pretende que el sistema sea capaz de razonar sin tener toda la información en cada momento, puesto que muchas veces un plan que podría parecer perfecto, por ejemplo, viajar de Madrid a New York sin escalas y a un coste mínimo, los días comprendidos entre el 20 diciembre y el 1 de enero, puede no ser posible debido a que no se encuentre alojamiento para esa fecha en la ciudad de destino. Pero ese problema no debería detener el proceso de resolución del problema, sino tal vez tratar de buscar soluciones alternativas, como buscar un alojamiento en el lugar más próximo posible (buscando que exista una forma de conectar ambos lugares).

3.2. Descripción del Problema desde el punto de vista de la Planificación

Desde un punto de vista clásico, el problema anterior se podría resolver como una búsqueda en un espacio de estados. Por ejemplo, utilizando la planificación que parte de un conjunto de operadores que describen las operaciones que puede realizar el usuario. Algunos de estos operadores que se deberían tener en cuenta son:

- Desplazarse dentro de una ciudad, utilizando diferentes medios de transporte local. Normalmente estos operadores trabajarán sobre información conocida y estática (horarios de autobuses, etc.).
- Desplazarse entre diferentes ciudades, para ello se debe considerar que este tipo de información deberá ser consultada en ese momento.
- Reserva de alojamiento mientras dure la estancia.
- Posibilidad de reservar un vehículo en la ciudad de destino.

Cada problema a resolver viene descrito por (ver Fig.4): un estado inicial, que describe las necesidades del usuario, como por ejemplo: ciudad origen/destino, fecha salida/regreso, preferencias del viaje, preferencias de alojamiento, etc. A partir de los operadores y del problema, el planificador intentará encontrar una solución (plan).

El problema que se plantea es intratable desde el punto de vista de la planificación clásica, debido a varios motivos:

- El número de los operadores necesarios para implementar las diferentes posibilidades puede ser tan elevado que todo el proceso de planificación explote computacionalmente.

- El tipo de información que se está tratando es de carácter muy dinámico por lo que se tendría que recurrir a alguna técnica de planificación que permita tratar conjuntos de datos que no son conocidos “*a priori*” por el sistema.
- El volumen de los datos tratados excede con mucho la cantidad habitual de información manipulada por los planificadores (recuérdese que una de las fuentes de información es la Web).
- El número de planes que pueden aparecer para resolver un determinado problema es claramente exponencial. Piénsese, por ejemplo, en cuántas posibilidades existen para realizar el viaje [Madrid, Bruselas]. Supóngase que se tiene a disposición 20 compañías aéreas que pueden llevar al destino deseado, o que si no le gusta el avión se puede intentar llegar en tren, o autobús. También se podría desear minimizar el gasto en avión y no importarle realizar diversos transbordos, o aprovechar una oferta aérea hasta una ciudad intermedia y luego conectar con la ciudad destino empleando cualquier otro medio de transporte. Las posibilidades se disparan.
- En último lugar, si se tiene en cuenta la oferta de alojamiento de cualquier ciudad, y que por cada tipo de alojamiento y forma de desplazarse al lugar de destino se tiene un plan posible, el tratamiento final de todo el problema no puede estudiarse desde un punto de vista clásico.

El hecho de que la cantidad de información sea muy elevada no implica que deban desecharse técnicas como la planificación para resolver un determinado problema. Por el contrario, se puede utilizar la planificación como un medio muy eficiente de resolver problemas.

Tal y como aparece en la Fig.4, el planificador puede ayudar a encontrar un conjunto de soluciones al problema. Estas soluciones estarán dadas a un nivel abstracto. El concepto de plan abstracto con el que se ha trabajado [5] se refiere a que no se conocen los detalles del plan que se quiere resolver, sino el conjunto de pasos de alto nivel necesarios para ejecutar el plan con éxito. Cada una de las posibles soluciones deberán ser posteriormente *validadas* con información procedente de otras fuentes. Cada solución que se detecta como válida se guarda y se muestra al usuario para su posterior evaluación. Una vez encontrada una solución satisfactoria, el proceso de búsqueda se detiene si el usuario no solicita otras soluciones alternativas.

Con esta técnica pueden emplearse heurísticas que ayuden a buscar en un determinado orden sobre ese conjunto de soluciones. En particular, la heurística más simple (y la más utilizada en este tipo de problemas), consiste en ordenar el conjunto de soluciones posibles por coste (económico) mínimo para el usuario y tratar de validar la solución óptima encontrada por el sistema.

3.3. Ejemplo de Aplicación del Sistema Multiagente sobre Web

Supóngase que se dispone de un dominio donde las ciudades posibles son: [Madrid, Londres, Atlanta], y que se desea realizar un viaje a la ciudad de Atlanta. Considérese que el pasajero está emplazado en la estación de trenes de Atocha (situada en la ciudad de Madrid), y que el problema planteado consiste en planificar un viaje desde su situación actual, al hotel “*Sheraton Inn Atlanta Airport*” situado en la mencionada ciudad. Su estancia será de 4 días, y una vez transcurrido ese período, el usuario desea retornar a la estación de trenes de Chamartín (también situada en Madrid).

El problema del que se parte le proporciona al planificador un estado inicial, y sobre ese estado se aplican el conjunto de operadores que consiguen devolver al usuario un plan adecuado. En la Fig.8, aparece una posible especificación de una parte de ese problema. Se le indica al planificador, que el usuario desea desplazarse desde una estación de tren situada en una ciudad *city0*, al aeropuerto de otra ciudad *city2*. Se puede observar cómo se han representado de forma genérica las diferentes ciudades, transportes, alojamientos, etc.

```
;; definición de un problema en el entorno
de viajes.

(setf (current-problem)
  (create-problem
    (name jitel99)
    (objects
      (persona person)
      (city0 city)
      (city1 city)
      (city2 city)
      (airp0 airp)
      (airp1 airp)
      (airp2 airp)
      (airplane0 airplane)
      (trst0 trainst)
      (trst1 trainst)
      (trst2 trainst)
      (train0 train)
      (train1 train)
      (train2 train)
      (bus0 bus))
    (state(and
      ;; ciudad 0 (Madrid)
      (loc-at airp0 city0)
      (loc-at trst0 city0) ;Atocha
      (loc-at trst2 city0) ;Cham.
      (at-train train0 trst0)
      (at-train train2 trst2)
      (at-airp bus0 airp0)
      (at-airp airplane0 airp0)
      ;; ciudad 1 (Londres)
      (loc-at airp1 city1)
      (loc-at trst1 city1)
      ;; ciudad 2 (Atlanta)
      (loc-at airp2 city2)
      (at-train persona trst0)))
    (goal (at-airp persona airp2)))) ;;
la meta consiste en llegar a algún
aeropuerto de la ciudad de Atlanta
```

Fig.8.- Representación abstracta de una parte del problema planteado.

Una de las posibles soluciones genéricas, proporcionadas por el planificador aparece en la Fig.9.

```
Solution:
bus-airp-to-trainst bus0 airp0 trst0 city0
geton-bus-in-trainst persona bus0 trst0
bus-trainst-to-airp bus0 trst0 airp0 city0
getoff-bus-in-airp persona bus0 airp0
geton-airplane persona airplane0 airp0
take-airplane airplane0 airp0 airp2
getoff-airplane persona airplane0 airp2
```

Fig.9.- Solución de PRODIGY al problema planteado.

Con esta información el agente selecciona aquellos operadores que deban disparar procesos de consultas (sobre la Web o sobre otros agentes del sistema). En este ejemplo en particular, se dispararía la obtención de los operadores que aparecen en la Fig.10.

```
take-bus-airp-trainst bus0 airp0 trst0 city0
take-bus-trainst-to-airp bus0 trst0 airp0 city0
take-airplane airplane0 airp0 airp2
```

Fig.10.- Operadores para acceso a Web

Esos operadores se diferencian en que la información que se necesita para validar los dos primeros es de tipo local. El agente debe conocer la existencia de un autobús que conecta la estación de trenes, con el aeropuerto de salida. Esta información se le puede proporcionar a un usuario, indicándole la línea y el horario del mencionado autobús. En cambio, el tercer operador indica la necesidad de poder realizar un vuelo de avión entre ambas ciudades (Madrid, Atlanta). Esta información necesitará ser consultada *online* por el agente a otros agentes del mismo tipo o a agentes Web que puedan solicitar la información a compañías aéreas que se encuentren en la Web. De esa forma puede garantizarse al usuario que el desplazamiento deseado es posible.

Ese conjunto de operadores deberán ser validados si se desea que la solución en cuestión sea correcta. Una vez que se han conseguido validar todas los operadores, se puede mostrar al usuario el plan completo que necesitaría para realizar su viaje. Si la solución aportada no es la adecuada a sus necesidades, se podría continuar trabajando sobre alguna de las otras posibles soluciones dadas por el planificador (si éstas existen).

Un agente Web transformó el operador:

```
<take-airplane airplane0 airp0 airp2>
```

en una *query* que lanzó a diferentes compañías aéreas, una de las respuestas obtenidas (sobre una query que solicitaba un vuelo ida/vuelta Madrid/Atlanta, con fecha de salida 29 Marzo, y fecha de regreso 2 de Abril), se muestra en la Fig.11.

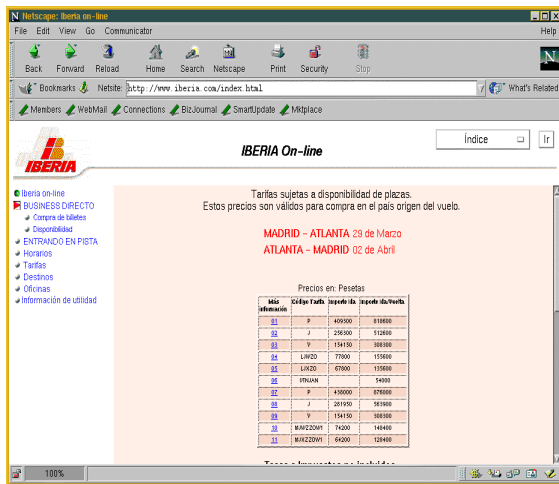


Fig.11.- Respuesta de la compañía aérea.

Como puede verse en la Fig.11, la información es devuelta por la compañía aérea en una página HTML. Esa página contiene información útil que deberá ser filtrada y almacenada en una representación adecuada para que el agente pueda utilizarla en su proceso de planificación.

En estas páginas se dispone de diferentes tipos de información, como existencia de vuelos; existencia de plazas en un vuelo (depende de la compañía utilizada); número total de vuelos; coste del vuelo sólo ida; coste del vuelo ida/vuelta, etc...

Dependiendo de la información que se desee utilizar para resolver el problema, se puede obtener un tipo de solución u otra; es decir, se pueden generar desde soluciones muy sencillas, donde únicamente se le indica al usuario la forma de realizar su viaje, a soluciones realmente complejas donde se le puedan indicar: costes, duración de los trayectos, compañías a utilizar, horarios, etc. La elección de un tipo de solución u otra, podría depender de las características o necesidades específicas del usuario.

4. Conclusiones

Se ha presentado una aproximación multiagente con agentes planificadores, al problema de resolución de problemas en entornos fuertemente dinámicos. La motivación de este trabajo va encaminada al estudio de cómo puede afectar, a una planificación de tipo dinámica, el hecho de que los datos sobre los que se debe realizar esa planificación también lo sean. También se pretende estudiar cómo se pueden utilizar las técnicas cooperativas de resolución de problemas sobre una cantidad masiva de datos. Se muestra un ejemplo de aplicación del sistema sobre la Web, utilizando para resolver el problema de planificación de un viaje, datos obtenidos automáticamente por el sistema.

Referencias

[1] A.Aamodt, E.Plaza. "Case-Based Reasoning: Foundational Issues, Methodological Variations, and

System Approaches". AICom-Artificial Intelligence Communications, IOS Press, Vol.7, pages 39-59.

[2] J.L.Ambite, C.A.Knoblock. "Planning by rewriting: Efficiently generating high-quality plans". In proceedings of the Fourteenth National Conference on Artificial Intelligence. 1997.

[3] N.Ashish, C.A.Knoblock. "Semi-automatic Wrapper Generation for Internet Information Sources". Second IFCIS Conference on Cooperative Information Systems (CoopIS), Charleston, South Carolina, 1997.

[4] W.Brenner, R.Zarnekow, H.Wittig. "Intelligent Software Agents. Foundations and Applications". Springer-Verlag, 1998. ISBN: 3-540-63411-8.

[5] J.Carbonell et al. "Prodigy4.0: the manual and tutorial". Carnegie Mellon, Computer Sciences, 1992, Technical Report. CMU-CS-92-150.

[6] O.Etzioni. "Moving Up the Information Food Chain". In AI Magazine, volume 18, nº 2, pages 11-18, summer 1997.

[7] Les Gasser. "An Overview of DAI". Distributed Artificial Intelligence: Theory and Praxis. Kluwer Academic Publishers. 1992.

[8] H.Kautz, B.Selman, and M.Shah. "The Hidden Web". In AI Magazine, volume 18, nº 2, pages 27-35, summer 1997.

[9] C.A.Knoblock, S.Minton, J.L.Ambite, N.Ashish, et. al., "Modeling Web Sources for Information Integration". In Proceedings of the Fifteenth National Conference on Artificial Intelligence, Madison, WI, 1998.

[10] M.Veloso, J.Carbonell, A.Pérez, D.Borrajó, E.Fink and J.Blythe. "Integrating planning and learning: The Prodigy architecture" Journal of Experimental and Theoretical AI, Vol.7, pages 81-120, 1995.

[11] D.E.Wilkins, D.L.Myers. "A Multiagent Planning Architecture". Proceedings on The Fourth International Conference on Artificial Intelligence Planning Systems. AIPS'98. June7-10, 1998.

JavaMAST: Un Entorno de Desarrollo de Agentes Inteligentes

Manuel García **, Mercedes Garijo *, Carlos A. Iglesias *,
Enrique Martín ** y Juan L. Mulas *¹

* Dep. de Ingeniería de Sistemas Telemáticos.
E.T.S.I. Telecomunicación E-28040 Madrid, España
Universidad Politécnica de Madrid
{mga, cif, juantxo}@gsi.dit.upm.es

** SGI Soluciones Globales Internet
Isaac Newton 11, PTM Tres Cantos, E-28760 Madrid, Spain
{megg, pemm}@esegi.es

Abstract

This article describes JavaMAST, a software development environment for building intelligent agents that can be used in conjunction with the agent-oriented methodology MAS-CommonKADS [10]. The environment consists of a agent definition language (ADL), a knowledge representation language (CKRL), an Java library for the agent creation and communication issues and an Agent Library, with standard agents and intelligent agents classes.

1. Introducción

JavaMAST es un entorno de desarrollo de agentes inteligentes, que puede ser usado junto con la metodología orientada a agentes MAS-CommonKADS [10].

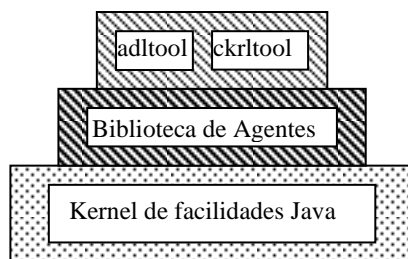


Figura 1: Arquitectura de Java-MAST

JavaMAST es una reimplementación completa en Java de MAST [11], desarrollado en C++, que consta de las siguientes herramientas (Fig. 1):

- *Facilidades de Agentes basadas en Java (biblioteca kernel)*: proporciona las funciones

básicas de creación y comunicación entre agentes.

- *Biblioteca de Agentes*: el entorno proporciona clases de agentes que pueden extenderse según el problema. Se definen dos clases genéricas (ver Fig. 2):
 - Agentes de Red: proporcionan funcionalidades básicas para comunicación del resto de agentes, tales como servidor de nombres de agentes, páginas amarillas y páginas blancas.
 - Agentes de Aplicación: son los agentes que ejecutan las funciones de la aplicación. La clase de *agente básico* tiene la característica de ser multiservicio, esto es, puede ejecutar concurrentemente varias tareas y se programa en Java. De esta clase se ha derivado el *agente Jess*, que permite la construcción de agentes inteligentes con el lenguaje basado en reglas *Jess* [5] y con Java.
- *Herramienta ADL*: el lenguaje ADL (*Agent Definition Language*) [11] permite la definición de los agentes, indicando sus componentes. La herramienta ADL traduce esta definición en código C++ o Java. El lenguaje facilita la composición de los módulos de los agentes

¹ Los autores están listados en orden alfabético.

(servicios, planes, etc.) previamente desarrollados.

- *Herramienta CKRL*: los agentes pueden emplear dos lenguajes de representación del conocimiento: Jess [5] y CKRL (*Common Knowledge Representation Language*) [2]. La herramienta CKRL traduce las definiciones de las ontologías en CKRL a objetos Java/C++ y realiza las operaciones de serialización de objetos para la recepción y envío de conceptos CKRL.

2. Biblioteca Kernel

2.1. Introducción

JavaMAST incluye un conjunto de clases programadas en Java que ofrecen un conjunto de funcionalidades básicas, tales como:

- Los procesos de comunicación de los agentes. Permiten la comunicación por paso de mensajes, permitiendo el mensaje individual o a grupos.
- La gestión y manejo de la base de conocimiento del agente. Dispone de mecanismos que facilitan la modificación, interpretación y el almacenamiento de los datos que componen la base de conocimiento de la que dispone el agente.
- La creación de nuevos servicios. Para la incorporación de un servicio nuevo al agente es tan sencillo como sobrescribir un método de una clase de biblioteca con el código del servicio.

También se dispone de una biblioteca de clases que facilita la creación de agentes que se ajusten a las necesidades de usuario de *JavaMAST*.

La plataforma también dispone de facilidades de red [11] en el que se implementan los mecanismos de coordinación de agentes, de servicio de nombres, páginas blancas y amarillas, registro de agentes en la red, etc.

2.2. Agente Genérico Multiservicio

La arquitectura de un agente genérico [13] se basa en la versión en C++ (*MAST-C++*) que ha sido mejorada mediante la compartición de memoria del agente multiservicio con hebras. Un agente dispone de los siguientes elementos:

- *Control*: se corresponde con una hebra, es la secuencia principal del agente y en ella se llevan a cabo las operaciones fundamentales de control del agente: lectura del buzón para procesado de mensajes de otros agentes, análisis de los cambios en el entorno, comprobación del estado del agente, desencadenamiento de ejecución de servicios, etc. En un agente "sin inteligencia" este bloque es sencillo, las actividades del agente

se reducen a la lectura del buzón y ejecución de servicios. En un agente "inteligente" las actividades llevadas a cabo por esta secuencia de programa son más complejas como se comenta más adelante.

- *Comunicaciones*: secuencia de programa encargada de recoger las peticiones de comunicación de otros agentes y de almacenar dichas peticiones en el buzón del agente. La lectura del buzón la realiza el control según la política de lectura de buzón que tenga definido el agente.
- *Servicios*: conjunto de hebras que aparecen en función de los distintos mensajes recibidos por el agente o como consecuencia de la evolución de las condiciones del entorno o el estado del agente. Estas hebras se encargan de la ejecución de determinados servicios de los que dispone el agente. Una de los ejemplos más comunes que desencadena la ejecución de un servicio es la recepción de un mensaje solicitando al agente la ejecución de un determinado servicio.
- *Base de Datos*: Conjunto de variables que almacenan información sobre el entorno, el agente, el estado del agente y otra serie de datos globales. Todas estos elementos constituyen el conocimiento del agente. En función del estado de esta base de datos el agente puede evolucionar hacia un estado u otro.

Cuando se lanza un agente, en primer lugar se ejecutan las acciones iniciales. Entre estas acciones iniciales se encuentran: el lanzamiento de los servicios iniciales u objetivos, en el caso de que existan, y la ejecución de la hebra de comunicaciones, encargada de almacenar los mensajes en el buzón. El Control, entre otras tareas, (puede ejecutar un programa que dote al agente de cierta inteligencia) realiza sucesivas lecturas del buzón en busca de mensajes procedentes de la sociedad de agentes. Alguno de los mensajes puede desencadenar la ejecución de uno de los servicios que ofrece el agente a la sociedad, el servicio puede ser concurrente (hebra) o no (lo ejecuta el propio Control).

JavaMAST define un agente básico que cuenta con los elementos descritos anteriormente. También ofrece una jerarquía de agentes. Esta jerarquía cuenta con dos tipos de agentes principales:

- *Agente de Red o Agente YP*, encargado de la gestión y mantenimiento del modelo de red de la plataforma, como se comentó anteriormente se encarga de añadir y borrar agentes, actualizar las bases de datos de los agentes, comprobar el estado de los agentes, etc.

- *Agente de Aplicación o Agente Base*, dispone de los elementos necesarios para registrarse en la red, recibir actualizaciones de su base de conocimiento, etc.

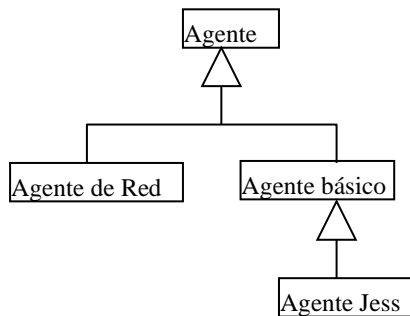


Figura 2: Jerarquía básica de *JavaMAST*

3. Agente Inteligente Multiservicio

3.1. Motivación

Los agentes genéricos de la plataforma son totalmente deterministas, ya que no poseen ninguna capacidad de toma de decisiones: simplemente intentan llevar a cabo los objetivos fijados en su nacimiento y atender las peticiones de servicio procedentes de otros agentes. La incorporación de un agente inteligente a la plataforma trata de eliminar, entre otros aspectos, el determinismo de los agentes genéricos. Dicho agente se caracteriza por ser deliberativo, exhibiendo cierta capacidad de razonamiento en base a unas reglas de comportamiento. Esta capacidad de razonamiento se consigue definiendo el estado del agente en términos mentales similares a los humanos, siguiendo el paradigma *AOP (Agent Oriented Programming)* [17]. Las reglas de comportamiento se construyen en función de estos términos mentales, lo que implica un comportamiento no determinista pues la activación de las reglas depende del estado actual del agente. En resumen, las mejoras aportadas por el agente inteligente frente a los genéricos son:

- *Deliberativo y basado en estados mentales*: convierten a los agentes inteligentes, a diferencia de los genéricos en no deterministas pues su comportamiento depende en todo momento de su estado mental actual, que puede variar en cada ejecución.
- *Declarativo*: la declaración de los agentes inteligentes deja de ser una tarea completamente procedimental debido al carácter declarativo de la creación de las reglas de comportamiento.

- *Posibilidad de adoptar objetivos en ejecución*: frente a la rigidez de los agentes genéricos, que están limitados a la ejecución de sus objetivos iniciales.
- *Mayor grado de autonomía*: otorgada por la capacidad de razonamiento basada en reglas que exhiben los agentes inteligentes, que les permite ser más autosuficientes (no requieren tanta ayuda o colaboración de otros agentes de su entorno pues poseen más inteligencia).

3.2. Desarrollo de una interfaz Jess-Java

La capacidad de razonamiento y de deliberación de los agentes inteligentes de la plataforma se ha conseguido gracias a la integración en la misma de un paquete denominado *Jess (Java Expert System Shell)* [5]. *Jess* se puede definir como un Entorno de Desarrollo de Sistemas Expertos en Java. Las principales funcionalidades que ofrece son:

- *Un motor de inferencias*: encargado de decidir qué regla de comportamiento del agente se dispara en cada momento. Puede considerarse que simula la capacidad humana de alcanzar una conclusión razonando.
- *Lenguaje declarativo basado en reglas*: permite la definición de las reglas de comportamiento y hechos que constituyen los estados mentales de los agentes inteligentes. Este lenguaje se ha extendido declarando nuevos términos que posibiliten la programación de los agentes en función de su estado mental.

El hecho de que *Jess* esté escrito íntegramente en Java posibilita que el comportamiento de los agentes sea programado usando *Jess*, Java o una combinación de ambos.

La integración de *Jess* en la plataforma consiste básicamente en la declaración de un objeto motor de inferencias, que dispone de todo el control sobre el agente: políticas de filtrado de mensajes y servicio, procesado de los mensajes recibidos, adopción de compromisos, ejecución de tareas, ...

El motor de inferencias se ejecuta en una hebra a la cual se la traspasa toda la información recibida del entorno o de la comunicación con otros agentes. Dependiendo de la información que reciba, el estado mental del agente variará, variando a su vez las reglas que el motor de inferencias disparará.

El usuario puede además interactuar en tiempo real con el agente inteligente a través desde su interfaz modificando su estado mental.

3.3. Modelo de agente inteligente

Tal y como se comentó con anterioridad, los agentes inteligentes están definidos mediante términos mentales humanos: creencias, capacidades y compromisos.

Las **creencias** son sentencias acerca del entorno de los agentes o sobre ellos mismos que cada agente considera ciertas. Los agentes pueden tener creencias de todo tipo, puesto que el usuario del agente puede introducirle todo el conocimiento que considere necesario va a ser de utilidad al agente para la ejecución de los objetivos que tiene prefijados y los servicios que ofrece al entorno multiagente. Los *objetivos* se pueden considerar como responsabilidades que un agente acepta, ya sea en su nacimiento como durante su vida. Ofrecer un *servicio* implica que el agente es capaz de adoptar un objetivo asociado al mismo y resolverlo.

Los agentes también deben disponer de la información relativa a las tareas que son capaces de realizar. Para representar las **capacidades** de los agentes, se ha empleado el concepto de *PSM (Problem Solving Method)* [9]. Representa simplemente una estrategia de resolución de un problema al que está asociado. Un PSM se puede ver como una estructura de control que descompone la ejecución de un problema (objetivo) en una secuencia de tareas. Una *tarea* representa una actividad que se ejecuta para llevar a cabo un objetivo. Lo que los agentes son capaces de ejecutar son tareas, y no objetivos, propiamente dichos.

Otro aspecto fundamental de los agentes inteligentes son los **compromisos**. Cuando un agente se compromete a realizar un servicio en nombre de otro agente, adopta la ejecución de dicho servicio como suyo propio, tratando de llevarlo a cabo lo antes posible.

Tomando como base todos estos conceptos, se ha desarrollado un pequeño lenguaje, extensión del lenguaje JESS, que permite la construcción de reglas de comportamiento basadas en ellos (ver [14]).

A modo de ejemplo, se muestra una regla (Figura 3) en la que el agente decide si puede comprometerse a efectuar un servicio solicitado, comprobando para ello si ofrece el servicio requerido y si dispone de la capacidad para llevarlo a cabo en estos momentos.

4. Protocolos de comunicación entre agentes

4.1. Introducción

En sus orígenes, la plataforma disponía de un lenguaje de comunicación entre agentes particular de la misma, hecho éste que impedía la interoperabilidad entre agentes de distintas plataformas. Con la intención de lograr esta interoperabilidad, se optó por integrar en la plataforma un lenguaje de comunicación estándar. Este lenguaje de comunicación debe ser de alto nivel para permitir un fácil intercambio de conocimiento entre los agentes.

El ACL (*Agent Communication Language*) que se decidió implementar fue el propuesto por la FIPA (*Foundation for Intelligent Physical Agents*) [4]. La FIPA es una asociación no lucrativa registrada en Suiza cuyo único propósito es promover el desarrollo de aplicaciones y servicios basados en la tecnología de agentes.

El ACL propuesto por la FIPA se caracteriza por ser un lenguaje declarativo pues se basa en el intercambio de actos comunicativos de alto nivel que permiten al receptor del mensaje interpretar su contenido. Posibilita de este modo el intercambio de conocimiento entre agentes en tiempo real y su cooperación en la resolución conjunta de problemas.

La integración del FIPA ACL en la plataforma ha implicado el cambio del formato de los mensajes. Para ello, se ha sustituido la clase mensaje del lenguaje previo por una con todos los campos de los mensajes de la FIPA [14].

```

(defrule commitToDoAGoalToAnotherAgent
  (service (name ?goal) (agent ?agent) (input $?input) (output $?output))
  (offered-services (services $?services) (agent myself))
  (test (member$ ?goal $?services))
  (available-PSMs (goal ?goal) (PSMs $?psms))
=>
  (agree:receiver ?agent)
    :in-reply-to ?rw
    :conversation-id ?id)
  (assert (commitment-to (goal ?goal) (input $?input) (output $?output)
    (agent ?agent) (PSM (choose-PSM $?psms))))
    ;elección de uno de los posibles PSMs
)

```

Tabla1: Ejemplo de regla de compromiso

4.2. Diseño de los protocolos

Los actos comunicativos son la base sobre la cual se establecen modelos de interacciones de alto nivel entre agentes. Las conversaciones entre agentes suelen caer en patrones típicos de intercambio de actos comunicativos. Esto es a lo que se denomina protocolos de comunicación. Su implementación simplifica el diseño de los agentes ya que estos no tienen la necesidad de comprender el significado exacto de los mensajes; les basta simplemente con conocer la secuencia de actos comunicativos que conforman el protocolo empleado.

Aunque no es necesario que un agente implemente ningún protocolo de interacción entre agentes, los agentes inteligentes de la plataforma poseen implementado por defecto en *Jess* [14] dos de los varios propuestos por la FIPA: *fipa-request* y *fipa-contract-net*. El primero sirve para pedir servicios o información a otro agente del sistema multiagente, mientras que el segundo es un protocolo de negociación.

Para cada protocolo implementado, se declaran los posibles intercambios de actos comunicativos, de tal forma que si se recibe un acto comunicativo correcto, se responde con el siguiente especificado. A modo de ejemplo se explica la implementación del protocolo *fipa-request*. La implementación en *Jess* se muestra en la Figura 4.

Aparte, para cada uno de los actos comunicativos recibidos hay que definir una regla que lo procese de la forma adecuada (asertar un objetivo en la base de hechos o memoria del agente, enviar un mensaje de aceptación (*agree*)).

Esta forma de definición de protocolos, permite extender fácilmente los protocolos soportados por los agentes.

5. Herramientas de desarrollo

5.1. Definición del conocimiento

Una parte importante de las herramientas proporcionadas por la plataforma de desarrollo multiagente MAST tienen que ver con el intercambio de conocimiento entre agentes, entendido éste como un conjunto de conceptos compartido por una sociedad de agentes y que determinan la ontología o grupo de ontologías necesarias para que la sociedad pueda llevar a cabo la resolución de aquellos problemas u ofrecer aquellos servicios para los que fue concebida.

Antes de abordar el diseño de las herramientas de intercambio de conocimiento, en los inicios del proyecto MIX [11], fue necesaria la elección de un lenguaje que permitiera representar conocimiento, es decir, especificar los conceptos que necesitan manejar los agentes.

```

(deffacts
  (protocol (name fipa-request) (state request) (next-state agree))
  (protocol (name fipa-request) (state request) (next-state refuse))
  (protocol (name fipa-request) (state agree) (next-state inform))
  (protocol (name fipa-request) (state agree) (next-state failure))
)

```

Tabla 2: Definición de protocolo fipa-request

De entre todas las alternativas posibles, se tuvieron en cuenta dos lenguajes especialmente diseñados para conseguir el propósito perseguido: KIF (*Knowledge Interchange Format*) y CKRL (*Common Knowledge Representation Language*) [8], decantándose la elección inicial por el segundo de ellos, el cual ha sido adaptado a las necesidades específicas del entorno MAST. Como consecuencia de esta elección, se han implementado un conjunto de herramientas que hacen posible el intercambio de conocimiento entre los agentes de la plataforma, permitiendo obtener clases y objetos C++ y Java a partir de definiciones especificadas en CKRL, tanto en tiempo de compilación como dinámicamente por los agentes durante su ejecución. Este conjunto de herramientas se denomina *ckrltool* [7] y está formado por los siguientes componentes: una biblioteca de clases C++-CKRL y Java-CKRL, un traductor estático (*ckrl2c++* y *ckrl2java*), y un traductor dinámico (*ckrl2c++* y *ckrl2java*).

La versión C++ de *ckrltool* permite traducir la descripción de conceptos CKRL a clases y objetos C++ a descripciones CKRL. Esto se realiza en tiempo de compilación para aquellas aplicaciones que usan CKRL como estructuras de datos para representar conocimiento y en tiempo de ejecución para las aplicaciones que transfieren conocimiento utilizando las facilidades de comunicación interagente proporcionadas por la plataforma MAST.

La biblioteca de clases C++ define un conjunto de clases que representan los elementos básicos de CKRL. Las descripciones CKRL de usuario (propiedades, conceptos, tipos, hechos y ejemplares) derivan de estas clases base, definidas para representar tipos de datos simples, listas homogéneas de tipos simples, listas heterogéneas y listas compuestas por tipos complejos.

El traductor estático *ckrl2c++* traduce una ontología CKRL en un conjunto de clases basándose en la biblioteca de clases CKRL. Los trabajos no solo se han enfocado al diseño e implementación de la herramienta en sí. Ha sido necesario extender la sintaxis del CKRL-MAST para permitir trabajar con herencia de conceptos CKRL y tipos de datos complejos. La herramienta *ckrl2c++* ha sido programada íntegramente utilizando el paradigma

orientado a objeto para lo que se han empleado herramientas como *flex++* y *bison++*.

El traductor dinámico *ckrl2c++* es básicamente una clase que permite a los agentes plasmar sus objetos C++ en descripciones CKRL, empaquetarlos en mensajes CKRL y enviarlos utilizando las facilidades de comunicación ofrecidas por la plataforma. Así mismo hace posible que los agentes puedan recibir ejemplares CKRL, almacenarlos y convertirlos en objetos C++ en tiempo de ejecución. Esta visión orientada a objetos del problema permite que cada hebra de un proceso pueda tener su propio traductor dinámico proporcionando procesado multihebra en el envío y recepción de ejemplares CKRL, puesto que los elementos recibidos se almacenan en el mismo objeto que contiene los métodos para manejarlos, propiedad de la hebra.

La versión Java de *ckrltool* dota a la versión Java de la plataforma MIX de las mismas funcionalidades que ofrece *ckrltool* C++. Al igual que la versión C++ está formada por una biblioteca de clases Java CKRL, un traductor estático y un traductor dinámico. Su diseño se ha llevado a cabo teniendo en cuenta las peculiaridades tanto de la nueva versión de la plataforma, como las del lenguaje Java en sí. Así por ejemplo se ha utilizado la serialización de objetos que ofrece Java a la hora de implementar los métodos que permiten enviar y recibir los ejemplares CKRL, a diferencia de lo ocurrido en la versión C++, donde se utilizaron métodos desarrollados en la plataforma para obtener la serialización de objetos. Las herramientas descritas se encuentran en continua evolución. Quizás, la línea de trabajo que más fuerza ha tomado últimamente es aquella que intenta ampliar el número de lenguajes que, como CKRL, se pueden utilizar en la plataforma. La evolución actual pretende ofrecer una herramienta muy completa, en la que se pueda especificar tanto el lenguaje para definir las ontologías como el lenguaje de programación utilizado en el desarrollo de la sociedad multiagente.

5.2. Definición de agentes

ADL2Java es una herramienta que permite crear agentes especificados en ADL (*Agent Definition Language*). Cada agente tiene una estructura formada por objetos que modelan sus servicios, objetivos y recursos entre otros, y que puede especificarse en un

lenguaje como ADL. Utilizando una jerarquía de clases Java de la biblioteca JavaMAST, el compilador ADL2Java crea a partir de la especificación en ADL los agentes en ella especificados.

Todo esto facilita la creación de los agentes y limita las tareas de programador a especificar los servicios y/o objetivos del agente. Además, mediante el uso de directivas, permite al programador personalizar aspectos de la sociedad tales como la ubicación del servicio de nombres (YP_AGENT), o las ontologías que emplearán los agentes para el intercambio de información.

El resultado es una herramienta que construye agentes Java, a partir de especificaciones en un lenguaje propio de arquitectos de sistemas y analistas (ADL), dejando en un segundo plano las tareas de codificación.

6. Conclusiones

En este artículo se ha presentado un entorno de desarrollo de agentes inteligentes que permite su programación mediante un lenguaje orientado a objetos (Java y C++) y un lenguaje declarativo basado en reglas (*Jess*).

Recientemente, existen entornos comerciales de desarrollo de agentes inteligentes en Java, tales como JACK [1] o ZEUS [15] y de agentes móviles en Java, tales como *Voyager* [16] o *Aglets* [12].

Sin embargo, estos entornos presentan las siguientes limitaciones:

- *Limitada noción de agencia.* Principalmente en los entornos de agentes móviles, los productos evaluados (tales como *Voyager* o *Concordia*) ofrecen la movilidad de objetos y la infraestructura para su movilidad. Sin embargo, estos objetos carecen de capacidad de razonamiento o autonomía.
- *Dificultad de extensión.* En los entornos comerciales, se ofrecen capacidades de razonamiento, y no se ofrecen facilidades conjuntas de movilidad e inteligencia. La no disposición del código fuente, limita las posibles extensiones que se deseen realizar en dichos entornos, y su uso queda condicionado a que los productos sigan en mantenimiento y se adapten a las nuevas versiones del cambiante lenguaje Java. *JavaMAST* emplea la potencia de un lenguaje tan difundido como *Jess* (versión Java del popular CLIPS).
- *Plataformas no abiertas:* estos entornos utilizan estructuras de mensajes propietarias, no implementando estándares de comunicación entre agentes, como KQML o la FIPA. Dentro de los

entornos de libre distribución, cabe destacar *JATLite* [6], que ofrece un módulo para intercambio de mensajes entre agentes empleando el protocolo *KQML* (*Knowledge Query and Manipulation Language*) [3]. *JavaMAST* ofrece una implementación de parte de los protocolos definidos en la FIPA, por lo que puede interoperar con plataformas de agentes que los implementen.

- *Falta de herramientas de desarrollo:* a excepción de ZEUS, el resto de entornos no proporciona ninguna herramienta de desarrollo. *JavaMAST* ofrece un lenguaje de definición de agentes y las herramientas asociadas, así como un depurador de los agentes inteligentes, que permite la ejecución paso a paso y la inspección de las variables del agente.

Entre los trabajos futuros de la plataforma, se ha iniciado ya la adición de movilidad a los agentes, la comunicación con CORBA, una herramienta de gestión del sistema y un entorno de análisis y diseño integrado de un sistema de agentes.

6.1. Referencias

- [1] Paolo Busetta, Ralph Rönquist, Andrew Hodgson, and Andrew Lucas. JACK Intelligent Agents - Components for Intelligent Agents in Java. *AgentLink Newsletter*, 1(2), January 1999.
- [2] Karine Causse et al. Final discussion of the Common Knowledge Representat Language (CKRL). Deliverable D2.3, MLT Consortium, ESPRIT project 2154, Laboratoire de Recherche en Informatique, Université d Paris Sud, Orsay, France, May 1993.
- [3] Tim Finin, Richard Fritzon, Don McKay, and Robin McEntire. KQML as an agent communication language. In *Proceedings of the Third International Conference on Information and Knowledge Management (CIKM'94)*, November 1994.
- [4] FIPA. Foundation for Intelligent Physical Agents. FIPA'97 draft specification. revision 2.0. Technical report, FIPA. Foundation for Intelligent Physical Agents, June 1997.
- [5] Ernest J. Friedman-Hill. *Jess, The Java Expert System Shell*. Distributed Computing Systems, Sandia National Laboratories, Livermore, CA, version 4.1 edition, June 1998.

- [6] Robert Frost. *JATLite User Manual*. Center for Design Research. Stanford University, 1998.
- [7] Manuel García García. Desarrollo de un conjunto de herramientas para el intercambio de conocimiento entre agentes inteligentes. Master's thesis, E.T.S.I. de Telecomunicación. Universidad Politécnica de Madrid, October 1997.
- [8] Michael Genesereth and E. Fikes, Richard. *KIF. Knowledge Interchange Format. Version 3.0. Reference Manual*. Computer Science Department. Stanford University, June 1992.
- [9] Carlos A. Iglesias. *Definición de una metodología para el desarrollo de sistemas multiagente*. PhD thesis, Departamento de Publicaciones, E.T.S.I. Telecomunicación, Universidad Politécnica de Madrid, February 1998.
- [10] Carlos A. Iglesias, Mercedes Garijo, José C. González, and Juan R. Velasco. Analysis and design of multiagent systems using MAS-CommonKADS. In M. Wooldridge, M. Singh, and A. Rao, editors, *INTELLIGENT AGENTS IV: Agent Theories, Architectures, and Languages*, volume 1365, pages 313-329. Springer-Verlag, 1998. (A reduced version of this paper has been published in *AAAI'97 Workshop on Agent Theories, Architectures and Languages*).
- [11] Carlos A. Iglesias, José C. González, and Juan R. Velasco. MIX: A general purpose multiagent architecture. In M. Wooldridge, K. Fischer, P. Gmytrasiewicz, N.R. Jennings, J.P. Müller, and M. Tambe, editors, *INTELLIGENT AGENTS II: Agent Theories, Architectures, and Languages*, volume 1037 of *Lecture Notes in Artificial Intelligence*, pages 251-266. Springer-Verlag, 1996. (A reduced version of this work was published in the *Proceedings of the IJCAI'95 Workshop on Agent Theories, Architectures and Languages*, Montréal, Canada, August, 1995, pages 216-224.).
- [12] Danny Lange and Mitsuru Oshima. *Programming and Deploying Java[tm] Mobile Agents with Aglets[tm]*. Addison-Wesley Longman, 1998.
- [13] P. Enrique Martín Martín. Desarrollo e implementación de un modelo de agente multiservicio en java. Master's thesis, E.T.S.I. de Telecomunicación. Universidad de Valladolid, May 1997.
- [14] Juan Luis Mulas Platero. Desarrollo de una arquitectura de agente inteligente multiservicio con la plataforma javamast. Master's thesis, E.T.S.I. de Telecomunicación. Universidad de Valladolid, February 1999.
- [15] Divine Ndumu, Jaron Collis, Gilbert Owusu, Matt Sullivan, and Lyndon Lee. ZEUS: A Toolkit for building Distributed Multi-Agent Systems. *AgentLink Newsletter*, 1(2), January 1999.
- [16] Object Space. *Object Space. Voyager. Core Technology 2.0. User Guide*, 1998.
- [17] Yoav Shoham. Agent-oriented programming. *Artificial Intelligence*, 60(1):51-92, March 1993.

Gestión de recursos en conmutadores mediante agentes inteligentes móviles

Antonio Barba Martí

Dept. de Matemática Aplicada y Telemática. Escuela Técnica Superior de Ingenieros de Telecomunicación
Universidad Politécnica de Catalunya (UPC).

c/ Jordi Girona 1-3, Módulo C-3, Campus Norte, 08034 Barcelona, SPAIN

email: telabm@mat.upc.es

Abstract

Broadband area networks require of new resource management mechanisms for switching nodes that allow to manage a network in real time with the objective to avoid congestion situations. In this article is presented a scheme of the resource management that is implemented in a switch using multiple mobile intelligent agents. A specific switch has been designed in order to support the operation of these cooperative multi-agents infrastructure. This design is developed as a solution with the purpose of providing an optimum quality of service to the subscribers ¹.

1. Introducción

En este artículo se presenta el diseño, actualmente en fase de construcción, de un conmutador que sirva como base para la implementación en software de una estructura cooperativa de agentes inteligentes móviles con el propósito de proporcionar una gestión de recursos de la red. De esta forma, se plantea una arquitectura de conmutadores que mediante estos agentes permite reducir las situaciones de congestión en una red y mejorar por tanto la calidad de servicio[1].

Los agentes inteligentes se pueden definir como entidades de software autónomas que se comportan de acuerdo a una inteligencia autocontenida. El agente inteligente es un termino que define desde interfaces de usuario adaptativos hasta comunidades de procesos inteligentes que cooperan unos con otros para conseguir una tarea común. Como agentes móviles representan objetos activos o transportables que se mueven desde un sistema a otro para acceder a recursos remotos o incluso encontrar otros agentes y cooperar con ellos.

La distribución de tareas de gestión en la gestión de red se conoce como gestión por delegación, adopta un paradigma de gestión descentralizada que tiene la ventaja del incremento de potencia computacional en nodos de la red y decrecimiento de la presión en sistemas de gestión de redes centralizados y en anchos de banda de red. La gestión por delegación habilita tanto la distribución temporal, mediante el procesado de operaciones simultáneamente en la plataforma de gestión y en cada nodo de la red, como la distribución espacial p.e. mediante distribución sobre nodos de red diferentes.

El objetivo básico consiste por tanto, en traer la inteligencia de gestión, p.e. los servicios de gestión, tan cercanos como sea posible a los recursos gestionados y a su representación lógica en forma de

objetos gestionados. Esto es, mediante la delegación. El agente en el nodo gestionado tiene que verse como un entorno de agente de ejecución específico, soportando la ejecución remota de scripts de gestión. Estos scripts pueden activarse según el tiempo, actividades de gestión o la aparición de eventos específicos en el agente gestionado.

2. Arquitectura multi-agente para la gestión de tráfico

El marco de funcionamiento de los controles de tráfico y congestión que se definan viene determinado por el modelo de la arquitectura de red que se presenta a continuación.

Los nodos se han diseñado de forma que puedan procesar de manera explícita diversos tipos de tráfico independientemente. A tal efecto se disponen estructuras de colas M/D/1 con prioridades distintas según el tipo de tráfico

El mecanismo de gestión de tráfico que se propone está basado en el uso de agentes predictivos distribuidos en los nodos de la red. Estos nodos a su vez, soportan algoritmos regidos por diversas condiciones de disparo según el nivel de tráfico detectado en la red. El propósito de este control de tráfico es prevenir y por tanto minimizar la congestión.

Los agentes forman parte de una red inteligente que trata de predecir el tráfico anticipándose de alguna manera a las variaciones de tráfico que pudieran llevar a una situación de congestión a la red. Los agentes situados en los conmutadores y en otros elementos de red son autónomos y se comunican entre si de forma que puedan adaptarse al comportamiento del tráfico de la red.

Estos sistemas cuentan además con un control de admisión de conexiones (CAC, Connection

¹ Este trabajo tiene el soporte de la CICYT:
TEL 98 - 0398.

Admission Control) que permite el multiplexado estadístico de múltiples llamadas según una calidad de servicio (QOS, Quality of Service) específica para cada llamada.

En este artículo, se profundiza en la parte correspondiente al algoritmo de los agentes, que basado en un nivel de disparo, permite adaptarse y reactivamente controlar el flujo de paquetes a la entrada de los nodos de acceso para evitar la congestión.

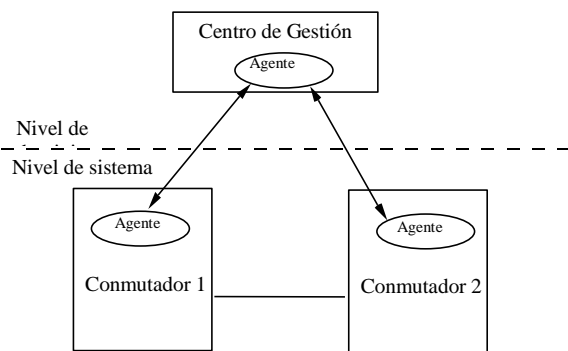


Fig. 1. Esquema de la arquitectura multi-agente de la red.

El proceso de gestión que se desarrolla en la estructura multiagente tiene cuatro etapas diferenciadas. En primer lugar, cada nodo monitoriza la información de gestión relevante de forma periódica mediante el agente. En un segundo paso, esa información se envía al centro gestor para un procesamiento inteligente de la misma. Como consecuencia de ese procesamiento, se adoptan una serie de decisiones sobre los elementos de red. Mediante la delegación de agentes, se envían los agentes que siguen los caminos de las conexiones por los conmutadores entre los extremos finales. Finalmente, se procede a la aplicación de las decisiones adoptadas en los conmutadores gestionados.

Este esquema de diseño es más óptimo desde el punto de vista de retardo en la ejecución de las operaciones de control. En el siguiente apartado se plantea un estudio comparativo frente a uno de los modelos de gestión más conocidos, basados en interrogaciones de status como es el protocolo SNMP.

2.1 Evaluación de la gestión basada en agentes móviles y en el protocolo SNMP

En el escenario de evaluación del comportamiento en la gestión de red se parte de una estructura centralizada basada en secuencias de interrogaciones desde una plataforma de gestión sobre objetos gestionados (p.e. en SNMP) frente a un diseño mediante agentes móviles.

Sea T_r Tiempo de respuesta completo para cualquiera de los dos esquemas de gestión.

t_m representa el tiempo de cálculo local del proceso del gestor en el gestor.

t_a describe el tiempo pasado por los agentes (SNMP o agente móvil)

t_d representa el retardo de la red.

$$T_r = t_m + t_a + t_d$$

A_1, A_2, \dots, A_n representan los agentes distribuidos.

t_{m1} representa el retardo desde el gestor al agente A_1

t_{1m} representa el retardo entre el agente A_1 y el gestor.

Para el caso cliente/servidor tradicional, el gestor necesitará construir un mensaje de petición tantas veces como n , donde n es el número de nodos agentes. Además, un agente móvil sólo necesita procesar una vez. El tiempo de cálculo local del proceso gestor en cada caso será:

$$t_m^{SNMP} = nt_m$$

$$t_m^{MOVIL} = t_m$$

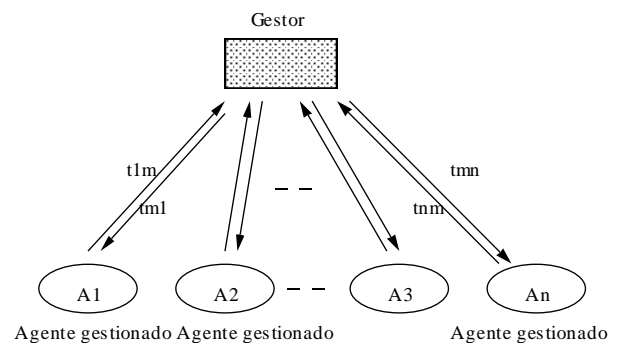


Fig. 2. Tiempo de respuesta de agentes SNMP.

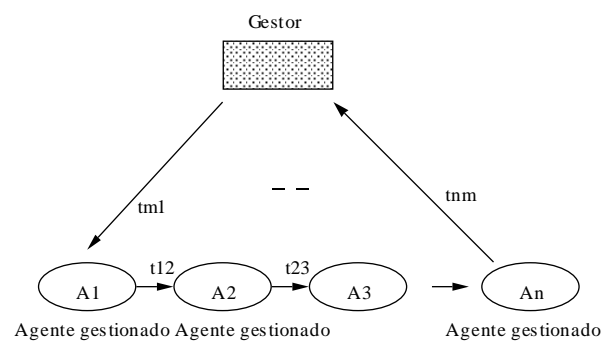


Fig. 3. Tiempo de respuesta de un agente móvil.

El retardo de la red total t_d es $2nt$ para SNMP y $(n+1)t$ para el caso móvil. Se asume que los retardos temporales entre todos los nodos distribuidos son los mismos. Si se asume también que t_m, t_a son constantes, el tiempo T_r será:

$$T_rSNMP = 2nt + n\alpha + \beta$$

$$T_rMOVIL = (n+1)t + \alpha + \beta$$

siendo

$$\alpha = t_m$$

$$\beta = t_a$$

Se observa que la ventaja del esquema basado en agentes inteligentes móviles se fundamenta básicamente en la descarga de procesamiento de las tareas de gestión desde el nodo gestor a los nodos gestionados, que utilizan los agentes inteligentes móviles para esa función. Este aspecto es más importante conforme crece el tamaño de la red a gestionar.

Para que la gestión basada en delegación sea más efectiva se requiere una cooperación entre los agentes con el fin de conocer el status completo de la red en tiempo real y poder actuar en las situaciones de congestión o fallo con la máxima rapidez..

3. Cooperación entre agentes inteligentes

Los agentes actúan sobre el conmutador para establecer una calidad de servicio adecuada valorando de manera diferente cada tipo de tráfico. Se definen cuatro clases de acuerdo al tipo de tráfico, esto es C1, C2, C3 y C4 en orden decreciente de prioridad. Los agentes adoptarán diferentes políticas de gestión según los requerimientos de calidad de servicio otorgados de acuerdo a cada clase de tráfico y nivel de prioridad [2].

En este artículo se plantea la gestión de tráfico según el tipo C4, que puede ser equivalente a un *best effort* utilizado por ejemplo en Internet.

3.1 Agente del nodo gestionado

Los nodos conmutadores de la red, a efectos de control del tráfico pueden modelarse mediante buffers específicos según las clases de tráfico que soportan. De igual forma se disponen sistemas de almacenamiento de paquetes compartidos a fin de disminuir las pérdidas de información al mínimo en caso de saturación de buffers o congestión de la red.

En el dibujo de la fig. 4 puede observarse la situación de un buffer C4 de un circuito integrado ASIC con umbrales variables en función de la carga de paquetes de entrada al conmutador.

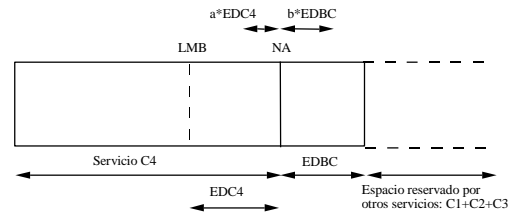


Fig. 4. Esquema de un buffer C4 de CI ASIC de un conmutador de la red troncal con los umbrales de funcionamiento de acuerdo al tratamiento de congestión.

Los parámetros adoptados por el algoritmo del agente del nodo gestionado son los siguientes:

- Longitud media del buffer C4 (LMB).
- Espacio disponible en el buffer compartido (EDBC) con $(a, b < 1)$.
- Tasa del puerto C4 de salida (TASA).
- Umbral alto del buffer C4 (NA).

En este caso, el resto de los umbrales podrían ser fijos en una primera aproximación al problema, una realización de la solución realmente inteligente exigiría unos márgenes variables de acuerdo al status del sistema gestionado.

Por otra parte, se pueden definir unas políticas de gestión en el ámbito de los nodos gestionados de la red. Es decir, una política P1 en donde el agente del nodo sólo puede utilizar un porcentaje b del espacio compartido disponible no usado por los otros servicios además del tamaño máximo asignado al servicio C4. Y de forma similar, una política P2 se define como aquella en la que el agente gestor sólo puede reducir el tamaño del buffer del nodo en un factor a.

Los resultados de las operaciones que realiza el agente del nodo gestionado se transfieren al agente del nodo gestor mediante los agentes inteligentes y con una periodicidad que depende de la prioridad marcada por el centro de gestión. La estructura de las agentes puede hacerse fácilmente compatible con los estándares definidos al efecto por el IETF y el ATM Forum. Los parámetros enviados como información por los agentes son los siguientes:

- Retardo: retardo de servicio medio según LMB y la TASA.
- Tmaxsum: incremento en $b \cdot (n^\circ \text{ slots de paquetes disponibles} / \text{TASA})$.
- Tmaxres: decremento en $a \cdot [(NA - LMB) / \text{TASA}]$.
- Tendencia: tendencia = $d(\text{EDC4}) / dt$.

El uso del parámetro Tmaxsum se circunscribe a la política de gestión P1 mientras que el parámetro Tmaxres estaría vinculado a la política de gestión P2. Hay que decir además que las medidas que se utilizan son de tipo temporal (en seg.) dado que las conexiones en la red pueden realizarse a velocidades de salida muy variadas con retardos de servicio de

paquetes diferentes y de esta forma se facilita el tratamiento de la información por parte del sistema.

3.2 Agente del nodo gestor

Junto a las políticas de optimización de los recursos de los nodos (P1, P2) se integran otra serie de políticas que tratan de mejorar la calidad de servicio en cada conexión de la red y a partir del status global. De esta forma el agente gestor interrelaciona el funcionamiento de los nodos conmutadores con los requerimientos de calidad del servicio contratados por el abonado. El agente del nodo gestor es el que se ocupa de aplicar las políticas de gestión correspondientes al resto de nodos de la red. Entre estas políticas consideradas más orientadas a servicio, se pueden destacar las siguientes: Política P3 mediante la cual el agente gestor distribuye el tiempo de servicio de crédito entre los conmutadores de una forma ponderada. Una política P4 en la que el conmutador más cargado recibirá más créditos que el que lo está menos. Si es necesario, se decremента el NA para mantener el mismo retardo de tránsito del paquete (RTP) especificado en la fase de establecimiento de la conexión. Finalmente, una política P5 en la que para cada dominio, el retardo de tránsito máximo experimentado por un paquete al pasar por un dominio sería siempre acotado por el máxRTP negociado en el establecimiento de la conexión.

3.3 Agentes inteligentes

El centro de gestión realiza un control inteligente mediante los agentes definidos. El sistema forma una infraestructura jerárquica distribuida de agentes. De forma integrada éstos operan en cada nodo el control de los buffers a nivel de registros de configuración para regular el envío de los paquetes de información de usuario. A continuación se documentan unos algoritmos utilizables como soporte de los agentes inteligentes.

Se construye una función de coste que se basa en el nivel correspondiente de congestión en todos los nodos de la red. Para ello se recoge la información de ocupación de las paquetes de los buffers correspondientes a los diferentes tipos de tráfico (C2, C3, C4). Se definen los siguientes términos:

- $f(\text{CNMi})$: Función de coste del conmutador i (CNMi).
- $d(\text{EDC1})/dt$, $d(\text{EDC2})/dt$, $d(\text{EDC3})/dt$, $d(\text{EDC4})/dt$: Tendencias correspondientes a los buffers de tipo de tráfico C1, C2, C3 y C4 respectivamente.
- T_{maxsumC1} , T_{maxsumC2} , T_{maxsumC3} , T_{maxsumC4} : Incrementos en $b \cdot (n^\circ \text{ slots de paquetes disponibles/TASA})$ para los distintos tipos de tráfico y buffer.
- T_{maxresC1} , T_{maxresC2} , T_{maxresC3} , T_{maxresC4} : Decrementos en $a \cdot [(NA - LMB)/TASA]$ para cada distinto tipo de tráfico y buffer.

- RetardoC1, RetardoC2, RetardoC3, RetardoC4: Retardos para los distintos tipos de tráfico y buffer.

Luego, se define una función de coste genérica:

$$f(C_i) = K_1 \cdot f(C_1) + K_2 \cdot f(C_2) + K_3 \cdot f(C_3) + K_4 \cdot f(C_4).$$

con $0 < K_1, K_2, K_3, K_4 < 1$.

siendo cada una de las tendencias definida análogamente como $f(C_4) = d(\text{EDC4})/dt$

El nodo gestionado actúa proporcionando los siguientes parámetros de forma continua, en este caso para el tráfico C4:

$$\text{Retardo_CNMi} = \text{LMB}/\text{TASA}$$

$$T_{\text{maxsum_CNMi}} = b \cdot (n^\circ \text{ paquetes libres}/\text{TASA})$$

Si $\text{LMB} < \text{NA}$

$$\text{entonces } T_{\text{maxres}} = a \cdot [(\text{NA} - \text{LMB})/\text{TASA}]$$

sino $T_{\text{maxres}} = 0$.

$$f(C_4) = d(\text{EDC4})/dt$$

El nodo gestionado actualiza su umbral de disparo NA a la recepción de cada agente siendo:

$$\text{NA} = \text{NA} + t \cdot \text{TASA}$$

En cuanto al agente del nodo gestor, de forma continua realizaría las siguientes operaciones para cada buffer integrado en los nodos. Además se limita el retardo global de la conexión CTD particularizando para el tráfico C4:

Si $\text{MaxRTP} > \text{Retardo_CNM1} + \text{Retardo_CNM2}$ entonces

/*Política P5*/

Caso

$f(C_4_CNM1) < 0$ y $f(C_4_CNM2) > 0$ hacer
Política P4

$f(C_4_CNM1) > 0$ y $f(C_4_CNM2) < 0$ hacer
Política P4

$f(C_4_CNM1) > 0$ y $f(C_4_CNM2) > 0$ hacer
Política P3

Los agentes gestores realizarían procesados más complejos a partir de la información proporcionada por los agentes de los nodos teniendo en cuenta la función $f(\text{CNMi})$. Actualmente se está evaluando el impacto de estos algoritmos en el funcionamiento de la gestión de una red avanzada desde un punto de vista más global.

4. Estructura de los conmutadores

Actualmente, en nuestro departamento se está procediendo a la construcción de una gama de conmutadores de banda ancha de diseño propio. Estos conmutadores utilizan circuitos integrados específicos (Texas Instruments) junto con DSP's y

otros procesadores de propósito general [3]. El objetivo consiste en utilizar esta plataforma hardware conocida para soportar el funcionamiento de una estructura software basada en agentes inteligentes cooperativos.

La arquitectura del conmutador se apoya en una placa base de PC que utiliza el conjunto de chips Pentium de Intel a la que se ha añadido el kernel del sistema operativo Linux. En uno de los slots PCI se conecta una tarjeta stand-alone que aloja un chipset Thunderswitch de Texas Instruments (TI) junto con un controlador Intel de la serie 8051 y un DSP de altas prestaciones, TMS320C6201 de TI.

La gestión, basada en los agentes inteligentes móviles, se apoya en la máquina virtual Java soportada por el procesador de la tarjeta PC. La gestión de los recursos del conmutador se realiza a partir del controlador y del procesamiento operacional del DSP. Éstos circuitos actúan sobre la configuración de los ASICs de TI.

Los agentes inteligentes cooperan en la gestión del tráfico y congestión de la red. De esta forma, configuran los registros de los circuitos ASIC del conmutador y posibilitan las siguientes funcionalidades:

Configuraciones de puertos de 10 Mbit/s

- Disposición de 16 puertos en modo half y full duplex
- Estándar 802.1Q VLAN
- Autonegociación con el nivel físico para el control de las pausas, modo duplex y velocidades de transmisión.
- Control de flujo basado en las colisiones en modo half duplex
- Control de flujo en full duplex según el estándar IEEE 802.3x

Configuraciones de puertos de 100 Mbit/s

- Disposición en tres puertos de 100 Mbit/s
- Estándar 802.1Q VLAN
- Control de flujo basado en las colisiones en modo half duplex
- Control de flujo en modo full duplex según el estándar IEEE 802.3x

El control de flujo en modo full duplex sigue el estándar IEEE 802.3x. En este caso, el circuito del conmutador reconoce cuando está llegando a una situación de congestión mediante la monitorización del nivel de ocupación del buffer. Cuando el número de buffers libres cae por debajo de un determinado parámetro, el conmutador previene la entrada de nuevas tramas activando el control de flujo apropiado al modo de operación de cada puerto.

El control de flujo se activa o desactiva globalmente. Cada puerto individual puede pedir activarse en modo half o full duplex o en flujo según el estándar

802.3 para su negociación con el dispositivo a un nivel físico.

El control de flujo según el estándar IEEE 802.3X proporciona un medio de reducir la congestión de red en puertos que son operativos en modo full-duplex, via tramas de pausa especiales.

La tarjeta también soporta *port trunking* y compartición de carga. El algoritmo de *trunking* determina en qué puerto de entre varios posibles se transmite las tramas, distribuyendo la carga incluso entre esos puertos y manteniendo un orden en el envío de paquetes.

El conmutador también soporta *routing* con las opciones siguientes:

- Disposición de *Port mirroring*
- Soporte del estándar IEEE 802.1D Spanning Tree
- Configuración de hasta 32 VLANs virtuales
- Soporte para interoperar con el estándar 802.1Q VLAN
- Soporte de tablas de enrutado para 2.000 direcciones

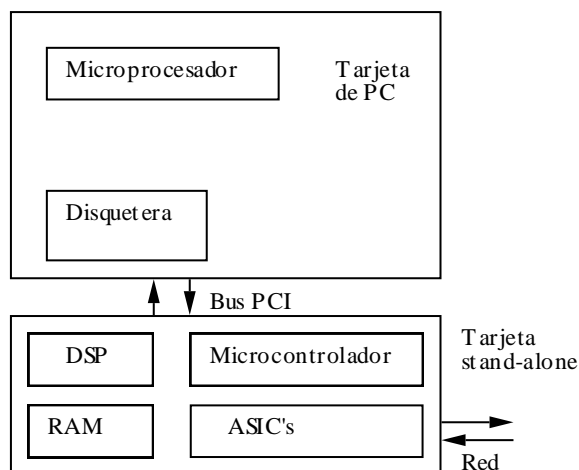


Fig. 5. Estructura circuital del conmutador

El procesador de la tarjeta PC recibe de la red a través de la tarjeta stand-alone el agente inteligente móvil definido en lenguaje java y que ejecuta mediante su máquina virtual. Las instrucciones obtenidas para el control de configuración de los circuitos integrados se envían al DSP para posteriores procesados si es necesario y finalmente se remiten a los registros de los ASIC's para asegurar un control dinámico de la configuración de éstos en tiempo real.

Las potencialidades del DSP en cuanto a su capacidad de procesamiento se aprovechan cuando se incorporan funcionalidades de la red correspondientes a niveles superiores, como es el caso del nivel de aplicación, en el diseño. En ese sentido, está previsto introducir medidas adicionales de seguridad para los nodos, funciones avanzadas de red inteligente, sistemas expertos, etc [4, 5].

5. Conclusiones

En el artículo se ha presentado una arquitectura de un conmutador que permite proporcionar gestión de calidad de servicio particularizada para un tipo determinado de tráfico (p.e. *best effort*, C4). Esta gestión de calidad puede aplicarse también a otros tipos de tráfico. El mecanismo estudiado que salvaguarda la pérdida de paquetes está basado en una estructura de agentes que actúan proporcionando un descarte selectivo de paquetes que tiene en cuenta la estructura de información del flujo de paquetes enviado a la red. Por otra parte, todo el sistema de gestión se basa en una arquitectura de múltiples agentes inteligentes sobre un diseño de conmutador propio. Existe aún todo un esfuerzo por realizar en torno a mejorar la eficiencia en la obtención de los parámetros que regulan los buffers de los conmutadores, así como en las tolerancias de los descartes en el control de flujo y en los demás recursos del sistema. El uso de los agentes inteligentes optimiza el comportamiento del sistema al hacer estos parámetros reconfigurables en tiempo real.

Referencias

- [1] David L. Tennenhouse, *A Survey of Active Network Research*. IEEE Communications Magazine. Enero 1997.
- [2] A. Barba. *Gestión de red ATM en una arquitectura multi-agente a partir de parámetros de control de congestión*. 1º congreso catalán de inteligencia artificial. p. 130-136. Octubre 1998. Tarragona.
- [3] <http://www.ti.com>
- [4] Jörg P. Müller. *The Design of Intelligent Agents. A Layered Approach*. Springer Verlag 1996.
- [5] Stuart Russell, Peter Norvig. *Inteligencia Artificial. Un enfoque moderno*. Ed. Prentice Hall. 1996.

Principios y Aplicaciones de las Redes Activas

□**María Calderón Pastor**
E-mail: mcalderon@fi.upm.es

□**Marifeli Sedano Ruiz**
E-mail: marifeli@aut.alcala.es

□**Santiago Eibe García**
E-mail: seibe@fi.upm.es

□Facultad de Informática. Universidad Politécnica de Madrid
Campus de Montegancedo, 28660 Boadilla del Monte (Madrid)

□Escuela Politécnica. Universidad de Alcalá de Henares
28871 Alcalá de Henares (Madrid)

Abstract

This paper presents an overview of a new network technology: active networks. The active networks goal is to produce a new networking platform, flexible and extensible at runtime to accommodate the rapid evolution and deployment of networking technologies and also to provide the increasingly sophisticated services demanded by users. A snapshot of the architecture being developed in DARPA active networks program is presented. Finally, potential applications of active networks are highlighted, along with some of the challenges that must be overcome to make them a reality.

1. Introducción

En las redes de paquetes actuales, los nodos intermedios (encaminadores o conmutadores) son sistemas cerrados e integrados verticalmente. El desarrollo de nuevos protocolos de red depende en primer lugar de comités de estandarización. Es necesario un largo proceso hasta que se logra una solución satisfactoria que sea consensuada por todos los miembros. Asimismo, el despliegue de nuevos protocolos también está sujeto a un largo proceso. Los fabricantes, de encaminadores o conmutadores, integran los nuevos protocolos únicamente en sus nuevos equipos. Consecuentemente aparecen problemas de incompatibilidad con los equipos antiguos que no soportan el nuevo protocolo. De aquí, que en muchos casos, los usuarios realmente no perciben los beneficios reales del nuevo protocolo hasta que pasado el tiempo, no son sustituidos los equipos antiguos.

Las tecnologías de redes de datos han evolucionado considerablemente a lo largo de las tres últimas décadas. Sin embargo estos avances no han modificado sustancialmente la funcionalidad básica consistente en el transporte transparente de bloques de datos de usuario entre dos puntos de terminación de red. En los modelos de red tradicionales los datos de los usuarios son transmitidos de forma opaca, la red es insensible a los datos que transporta y éstos son encaminados de forma transparente y sin modificación. El proceso que se realiza dentro de la red es muy limitado: procesamiento de la cabecera en una red de conmutación de paquetes con un procesamiento adicional para la señalización, en el caso de que la red proporcione servicio orientado a conexión. Este procesamiento de las cabeceras del paquete y de la información de señalización es independiente de la aplicación o proceso de usuario que genera el paquete. Por otro lado, debido a que el servicio es único para todas las aplicaciones (con un estrecho margen de opciones y facilidades) es preciso realizar estudios detallados de caracterización de

aplicaciones, y alcanzar compromisos de diseño para satisfacer la parte común de los requisitos del máximo número de aplicaciones.

Las Redes Activas [1] ofrecen un paradigma distinto permitiendo la programación de los nodos intermedios de la red, lo cual supone un gran salto conceptual en la evolución de las tecnologías de red. Las redes activas constituyen una arquitectura de red novedosa en la que los nodos de la misma pueden realizar procesamiento “a medida” sobre los paquetes que los atraviesan. Las redes activas producen un cambio en el paradigma de red: de llevar bits de forma pasiva, al de una máquina computacional más general. Otro gran salto conceptual se produce debido a que se propone que el comportamiento de los nodos pueda ser programado *en línea*, bien por el administrador de la red o por los propios usuarios en determinadas circunstancias, o lo que es lo mismo permite la modificación dinámica del comportamiento de la red.

La programación de la red puede resultar útil a distintos niveles, siendo sus principales potencialidades las siguientes:

- Acelerar la evolución de las redes, al permitir que nuevos protocolos y servicios sean introducidos en la red sin la necesidad de largos procesos, de estandarización y despliegue en la red. Desde el punto de vista de los proveedores de servicio de red, las redes activas permiten reducir el tiempo necesario para desarrollar y desplegar nuevos servicios de red.
- Permitir a los usuarios crear nuevos servicios o adaptar los ya existentes de acuerdo a las necesidades concretas de sus aplicaciones. Asimismo es posible imaginar usuarios que adapten los servicios, en base a opciones proporcionadas por “terceras partes” que vendan servicios de red mejorados.

- Facilitar la experimentación de nuevos protocolos o nuevos servicios de red. Desde el punto de vista de los investigadores, las redes activas ofrecen una plataforma en la cual experimentar con nuevos servicios de red de forma real, mientras que ésta continúa operando normalmente.

Un aspecto importante a reseñar es, que la utilidad de las redes activas está relacionada con la velocidad de la red [2]. En redes con poco ancho de banda, el procesamiento de todos los paquetes es factible. Pero cuando el ancho de banda aumenta, la capacidad de computación disponible por byte decremента, hasta tal punto, que el único procesamiento que se puede realizar por paquete es el relacionado con la función en encaminamiento. De aquí que sólo una pequeña fracción de los paquetes pueda ser procesados, y es necesario que en cada paquete se indique si se necesita un procesamiento *rápido* (el asociado al encaminamiento) o *lento*. La mayoría de los paquetes sólo necesitarán un procesamiento rápido (denominado usualmente *Fast-path processing* en terminología inglesa).

Con esta idea en mente, los investigadores en redes activas se han planteado como requisito de diseño, la construcción de un nodo activo que además de realizar procesamiento a medida sobre algunos paquetes que lo atraviesan, sean capaces de encaminar datagramas IP tradicionales, a una velocidad comparable a la conseguida por los encaminadores IP “pasivos” existentes en la actualidad.

Este artículo se estructura de la siguiente forma. En la siguiente sección se exponen una serie de aspectos que se han identificado como claves para el desarrollo de una plataforma de red activa, y que marcan las diferencias esenciales entre las distintas propuestas planteadas hoy en día. En la sección tercera se presentan las ideas básicas de la arquitectura de redes activas que está siendo desarrollada por DARPA. En la sección cuarta se plantean algunas posibles áreas de aplicación de las redes activas, haciendo especial énfasis en el área de multipunto fiable. Por último se presentan las conclusiones obtenidas.

2. Aspectos claves de una plataforma de red activa

En esta sección plantearémos diversos aspectos, que se han identificado como claves, al diseñar una plataforma de red activa. El modelo y funcionalidad de una plataforma concreta de red activa están determinados por las soluciones tecnológicas elegidas para resolver cada uno de estos aspectos.

2.1. API de Red

La programación de la red se hace en base a la utilización de una *Application Programming Interface de red* (API de red). En nomenclatura de redes activas, una API de red define una *máquina virtual* que interpreta un lenguaje específico. El API de red para IP incluye el lenguaje que define la sintaxis y la semántica de la cabecera IP y su efecto en los enca-

minadores de la red. En las redes tradicionales la máquina virtual es fija, y el poder expresivo del lenguaje limitado. Si podemos ver la cabecera IP de una red tradicional como la *entrada de datos* a una máquina virtual, podemos ver que en una red activa la entrada de datos a la máquina virtual son, además de los datos, los programas que porta el propio paquete.

Poder expresivo del lenguaje

De entre todos los aspectos que determinan cómo es el API de red, el más crucial es el lenguaje utilizado para programar la red.

El grado de programabilidad del API de red puede ser muy diverso. Las posibilidades van desde la selección de una simple lista de parámetros seleccionados de entre un conjunto predeterminado de posibilidades, a la utilización de un lenguaje completo equivalente Turing capaz de describir cualquier tipo de computación. Entremedias existe una infinita gama de posibilidades basadas: en restringir de distintas formas los lenguajes de propósito general o en la utilización de lenguajes de propósito específico.

Un aspecto esencial del lenguaje de programación es la facilidad de acceso a los recursos del nodo (por ejemplo, colas de salida o tablas de encaminamiento), para esto es determinante cuál es el conjunto de abstracciones asociados a éstos que proporciona el lenguaje.

La elección de un determinado lenguaje de programación está muy relacionada con la arquitectura de seguridad de la red activa. Los nodos se pueden programar, pero cualquier plataforma de redes activas debe ofrecer ciertos niveles de seguridad que garanticen que un programa inyectado en el nodo, ya sea de forma intencionada o inintencionada, no haga cosas “prohibidas” que pongan en peligro el funcionamiento del nodo. Con cosas prohibidas nos estamos refiriendo a programas que se meten en un bucle infinito, que acceden a zonas de memoria protegidas, o que hacen un uso abusivo de los recursos.

La ventaja de utilizar un lenguaje restringido es que limita los posibles comportamientos del nodo, simplificando el análisis de que el comportamiento programado es correcto, y por otro lado, acota de algún modo el efecto que un determinado paquete puede hacer en un nodo. En esta línea algunos de los lenguajes propuestos no admiten la existencia de estructuras de control del tipo bucles o bifurcaciones [3], o no admiten la utilización de punteros [4].

Alexander [2] ha identificado una serie de propiedades que debería poseer un lenguaje de programación válido para redes activas, siendo las más relevantes las siguientes: ser fuertemente tipado, disponer de un recolector de basura, posibilitar la carga dinámica, tener un mecanismo que permita definir diferentes interfaces para acceder a un mismo módulo y ofrecer buen rendimiento.

Almacenamiento temporal en el nodo

Otra característica importante del API de red es si permite que un programa incluido en un determinado paquete activo, deje residente información (por ejemplo información de estado), que pueda ser utilizada posteriormente por otros paquetes (por ejemplo que pertenecen al mismo flujo) que atraviesen el nodo posteriormente. Normalmente esta memoria es denominada *soft-state* pues tiene asociado un tiempo de vida pasado el cual si no se ha accedido a dicha información de alguna forma (para leerla o modificarla), dicha información se borrará del nodo sin que exista ningún tipo de notificación. Obviamente cuando se ofrece esta posibilidad, deben incluirse mecanismos que protejan de accesos no autorizados a dicha información de estado, y mecanismos que acoten la cantidad de memoria que puede utilizar un determinado usuario o flujo. La existencia o no, de memoria del tipo *soft-state* hace que sea muy diferente qué es posible programar en una red activa.

Composición dinámica de servicios

La meta última de las redes activas es el hacer más fácil el desarrollo de nuevos servicios de red. En concreto, debería proporcionar cierto soporte al proceso de creación de servicios. De aquí que una importante característica que debe incorporar el API de red sea la posibilidad de poder *componer* servicios a partir de bloques básicos. Estos bloques básicos son usualmente denominados *componentes*. Un API de red debería contener un mecanismo de composición para poder construir servicios compuestos en base a determinadas componentes. Existen trabajos que apuntan soluciones en esta línea siguiendo distintas aproximaciones, como por ejemplo el sistema LIANE [5] que plantea un sistema de composición basado en eventos, o el Netscript [6] donde se utiliza un modelo de computación de flujo de datos.

2.2. Sistema de Distribución de Código

Existen dos enfoques principales para realizar la descarga de código en el nodo: uno denominado discreto o fuera de banda y otro llamado integrado o en banda [1].

Enfoque discreto. Se mantiene el formato de paquete existente y se proporciona un mecanismo que soporta de forma separada la descarga de programas (*extensiones activas*), independizando esta descarga de programas, del proceso realizado sobre los paquetes. Cuando llega un paquete a un nodo se examina su cabecera y se lanza el programa adecuado para procesar dicho paquete. Esta separación puede resultar interesante en los casos en los que la selección de los programas que se deben inyectar en la red es realizada por administradores de red, en lugar de por usuarios finales. En este enfoque, el nodo programable no es tan “activo” como puede ser el enfoque integrado, pero es más fácil de implementar y desarrollar en las redes actuales.

Enfoque integrado. En este enfoque, los paquetes pasivos de las redes tradicionales son reemplazados por *paquetes activos* que incluyen tanto los datos como el programa activo, relativamente pequeño,

que será ejecutado en los nodos cuando los vaya atravesando el paquete activo. En realidad, en el paquete activo puede ir realmente el código o sólo una referencia a dicho código, de ahí las dos posibilidades siguientes:

- *Código embebido*, en el que el código asociado al paquete activo va incluido en el propio paquete activo [3]. Este enfoque es muy adecuado cuando el código asociado al paquete activo tiene un tamaño muy pequeño o cuando se trata de paquetes activos que se mandan muy pocas veces.
- *Carga bajo demanda*, en el que los paquetes activos identifican el proceso que se debe ejecutar en el nodo, y este se carga la primera vez que se recibe un paquete activo de ese tipo [7]. De esta forma el proceso que trata un tipo de paquetes activos sólo se cargará una vez por sesión o mientras se esté mandando paquetes activos de ese tipo.

En ambos casos, enfoque discreto o integrado, deberán existir mecanismos que permitan la carga dinámica de código. Este mecanismo de carga dinámica permite que el código se cargue en el nodo mientras este se encuentra operativo, sin que sea necesario interrumpir temporalmente su funcionamiento.

2.3. Seguridad

En una red activa existen dos tipos de funcionalidades encargadas de proporcionar seguridad al sistema.

Por un lado existe una funcionalidad, usualmente denominada seguridad (*security* en terminología inglesa) que se encarga de evitar que usuarios no autorizados puedan causar daños intencionados al sistema, incluso hasta el punto de dejar un nodo o la red no operativa. Estos daños pueden consistir en dejar a un nodo sin recursos (por ejemplo memoria), cambiar las tablas de encaminamiento, modificar una extensión activa o el código embebido en un paquete generado por otro usuario. Generalmente, los mecanismos utilizados para proveer esta funcionalidad se basan en utilizar mecanismos de autorización y autenticación, firmas digitales y listas de control de acceso.

Por otro lado, existe una funcionalidad (denominada unas veces *safety* y otras *robustness* en terminología inglesa) que se encarga de evitar que usuarios autorizados, ya sea de forma intencionada o no, puedan causar daños al sistema o a otros usuarios. Estos daños pueden consistir en enviar a un nodo un programa que se mete en un bucle infinito, acceder al espacio de memoria que está siendo utilizado por otro usuario o por el propio nodo, etc. Los mecanismos utilizados para lograr este tipo de seguridad están en muchos casos relacionados con las propiedades del lenguaje utilizado para programar la red, como es la utilización de lenguajes fuertemente tipados y con recolector de basura. También se ha propuesto la utilización de técnicas formales que permitan probar la *corrección* de un programa antes de que éste sea introducido en la red [8].

Uno de los principales retos para definir una arquitectura de seguridad satisfactoria es el coste de proceso de los mecanismos de seguridad asociados, que en muchos casos la hace inviable en una red activa en producción.

Dado que este aspecto constituye un factor crítico en las redes activas, en la arquitectura de red activa propuesta por DARPA y presentada en la sección 3 se plantea como objetivo de diseño que, deben siempre estar limitadas las consecuencias que puedan tener las acciones realizadas por un usuario, aunque éste disponga del nivel más alto de autorización.

2.4. Granularidad del Control

Este atributo se refiere a cuanto tiempo se mantendrá modificado el comportamiento del nodo (por ejemplo un cambio en el algoritmo de encaminamiento del nodo). Una primera posibilidad es que este cambio del comportamiento del nodo afecte a *todos* los paquetes que atraviesen el nodo hasta que no llegue otro paquete que modifique de nuevo dicho comportamiento. Otra posibilidad es que este cambio afecte únicamente al paquete que lleva el programa, granularidad por paquete. Y entre estas dos posibilidades existe otra intermedia que hace que el cambio en el comportamiento se restrinja a todos los paquetes pertenecientes a un determinado flujo. Siendo un flujo un conjunto de paquetes que comparten una característica común como por ejemplo tener una determinada dirección origen.

2.5. Propuestas actuales

A continuación haremos una breve descripción de las principales propuestas de modelos de red activa que se han realizado hasta la actualidad.

El proyecto Switchware [9] de la Universidad de Pensilvania. En Switchware se permite la opción de código embebido (programas PLAN contenidos en los programas activos) o la aproximación discreta de instalación de extensiones activas (*switchlets*). El PLAN (Programming Language for Active Networks) es un lenguaje funcional de propósito específico basado en OCAML. PLAN es muy simple y está diseñado teniendo como objetivo que los programas PLAN no puedan violar la política de seguridad[3]. Esta política se pretende que sea lo suficientemente restrictiva como para que los administradores de red permitan que los programas PLAN se ejecuten sin ser necesaria una autenticación previa. En esencia PLAN permite unas cuantas primitivas básicas, composición secuencial e invocación a *servicios externos* que pueden ser: extensiones activas (*switchlets*) o facilidades embebidas en el propio sistema. Estos servicios externos pueden requerir de una autenticación y/o autorización antes de permitir el acceso a los recursos.

En Switchware, la granularidad de control es por paquete para el caso del código PLAN, mientras que los *switchlets* están disponibles para todos los paquetes que atraviesen el nodo. En este proyecto han planteado una arquitectura de seguridad denominada SANE [10].

La herramienta ANTS [7] del MIT. El API de red ANTS consiste en una máquina virtual Java (JVM) aumentada con determinadas clases ANTS, que implementan métodos para permitir que los paquetes activos (*cápsulas* en terminología ANTS) sean decodificados e interpretados. ANTS permite disponer de *soft-state* en los nodos activos. La granularidad de control puede ser por paquete o por flujo. Como método de descarga de código utiliza el enfoque de carga bajo demanda.

El ANTS utiliza los mecanismos de seguridad propios de Java y además incorpora mecanismos de huella digital.

Dado que el estándar JVM no soporta acceso a recursos de transmisión a un nivel suficientemente bajo, la implementación de ANTS en plataformas estándares no puede soportar capacidades de QoS y está limitada a las capacidades de red básicas proporcionadas por Java.

El proyecto Smart Packets de BBN [4]. El objetivo central de este proyecto consiste en investigar cómo las redes activas pueden hacer más eficientes las tareas de administración y gestión de red. Para la distribución de código utiliza la aproximación de código embebido dentro de los paquetes activos. Uno de sus requisitos de diseño es que un paquete activo debe caber en un único paquete de nivel de enlace. Dada la hegemonía de Ethernet ellos han limitado que un programa activo debe tener como máximo 1024 bytes a los que se suman las cabeceras de control e información de autenticación. Han desarrollado dos lenguajes: el Sprocket y el Spanner. El Sprocket es un lenguaje similar a C modificado, para mejorar la seguridad (no dispone de punteros) y para añadir primitivas de red. Spanner es un lenguaje ensamblador, diseñado para permitir una codificación densa, en el que se compila Sprocket de modo que los paquetes activos portan código Spanner.

El API de red se implementa mediante una máquina virtual Spanner que se ejecuta como un demonio en un nodo activo.

Smart packets incorpora sistemas de seguridad basados en mecanismos de autenticación mediante el intercambio de certificados.

3. Arquitectura de Red Activa

En esta sección presentamos las principales ideas de la arquitectura de red activa que está siendo desarrollada por DARPA (Defense Advanced Research

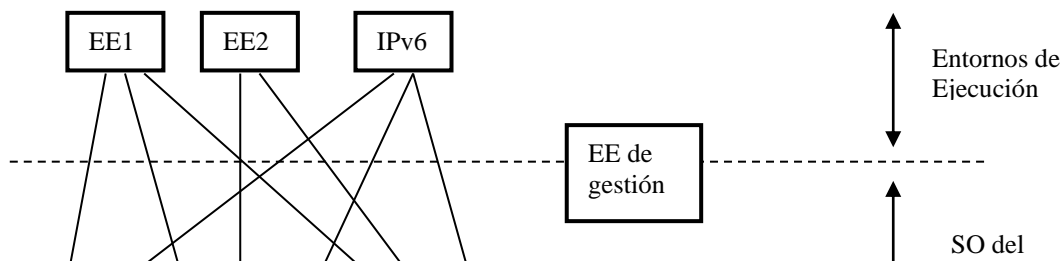


Fig. 1: Componentes

Projects Agency) dentro del programa de redes activas[11]. Esta arquitectura tiene como propósito definir cuales son los principales componentes de una plataforma de red activa y los interfaces entre éstos. Uno de sus requisitos es permitir que exista más de una API de red. Esto tiene varios objetivos. Por un lado, actualmente han sido propuestas varias APIs de red y todavía no existe la experimentación suficiente como para decidir cual es la mejor o incluso si debe existir una única API de red. Por otro lado, permite un marco en el que incorporar el “procesamiento rápido” de paquetes que hemos comentado previamente. Y por último, resuelve problemas de incompatibilidad, dado que la funcionalidad de IPv4 o IPv6 puede ser vista simplemente como otra API de red.

La funcionalidad de un nodo de la red activa se divide entre los entornos de ejecución (EE) y el sistema operativo del nodo (NodeOS). En la figura 1 se muestra la organización general de estos componentes que a continuación se analizan con más detalle.

3.1. Entorno de Ejecución

Cada EE exporta un API de red o máquina virtual que los usuarios pueden programar o controlar enviándole paquetes. En general, la ejecución de estas instrucciones puede causar que haya una actualización en la información de estado del EE o incluso del NodeOS, y puede causar que el EE transmita uno o más paquetes, ya sea inmediatamente o pasado un cierto tiempo.

Posibles EEs residentes en un nodo podrían ser: ANTs que sería un EE de propósito general o el Smart Packets que sería un EE específico de gestión de red. El EE oculta al NodeOS los detalles de la interacción con los usuarios.

Cada nodo activo dispone de *un entorno de ejecución de gestión* a través del cual se controlan ciertos aspectos de la configuración del nodo. Por ejemplo, el EE de gestión puede permitir la modificación de la base de datos con la política de seguridad del nodo (ver figura 1) o puede permitir el arranque de otros EEs.

3.2. Sistema Operativo del Nodo

El NodeOS es el nivel intermedio que opera entre los EEs y los recursos físicos (transmisión, computación y almacenamiento). El NodeOS proporciona la funcionalidad básica sobre la cual los entornos de ejecu-

ción construyen las abstracciones presentadas al usuario. El NodeOS gestiona los recursos del nodo activo y media entre los distintos EEs cuando se solicitan dichos recursos. El NodeOS aísla a un entorno de ejecución de los efectos que puede tener el comportamiento de otros entornos de ejecución.

Los usuarios y otras entidades de la red se representan por una abstracción denominada *principal*. Esta abstracción se utiliza en temas de seguridad y de contabilización de uso de recursos. Cuando un EE solicita un servicio del NodeOS, esta petición se acompaña del identificador (y posiblemente una credencial) del principal, en cuyo nombre se está realizando la petición. Dicho principal puede ser el propio EE u otra entidad (por ejemplo, un usuario) en cuyo nombre actúa el EE. El NodeOS presenta esta información al motor de cumplimiento de la seguridad que verifica la identidad (ver figura 1) y autoriza o no al principal a obtener el servicio solicitado.

Procesamiento de los paquetes

El NodeOS implementa la abstracción de *canal de comunicaciones* sobre la cual se envían y se reciben paquetes. Existen dos tipos de canales: los *anchored* y los *cutthrough*. Los canales *anchored* están anclados a un determinado EE y son la abstracción sobre la cual los EEs envían y reciben paquetes. El flujo lógico de los paquetes a través de un nodo activo es el siguiente: cuando un nodo activo recibe un paquete desde un enlace físico, lo clasifica en base al contenido del mismo (normalmente con la cabecera), y de acuerdo con esto: el paquete será descartado o bien asignado a un determinado canal.

El NodeOS utiliza patrones para asociar cada paquete recibido a un determinado canal. Dichos patrones son proporcionados por el EE cuando crea dicho canal (por ejemplo paquetes con una determinada combinación de protocolo IP y puerto TCP).

Además de los canales *anchored* también pueden existir los canales *cutthrough* que por el contrario no están asociados a ningún EE. Cuando un NodeOS recibe un paquete sobre un canal de este tipo, dicho paquete no será interceptado ni procesado por ningún EE, y la única operación que se realizará con dicho paquete será la de almacenamiento-reenvío, típica de una red *pasiva*. Cómo se puede observar esta abs-

tracción se utiliza para implementar el procesado rápido de paquetes ya antes comentado.

3.3. Active Network Encapsulation Protocol

El *Active Network Encapsulation Protocol* (ANEP) [12] proporciona los mecanismos para que un usuario pueda especificar a que EE particular debe dirigirse el paquete que está generando. Un campo en la cabecera ANEP es un identificador del EE que debe ejecutar dicho paquete. En la actualidad estos identificadores de EE son asignados por la Autoridad de Números Asignados para Redes Activas.

Es interesante señalar que un paquete que no contiene una cabecera ANEP puede ser procesado por un EE siempre que dicho EE cree el canal *anchored* adecuado. Este podría ser el caso de un paquete que ha sido generado en un sistema final no activo y se desea que se procese en un EE. Un ejemplo de aplicación podría ser un servicio de mejora del control de congestión de TCP implementado en un EE.

4. Algunas Areas de Aplicación

Ya hemos mencionado que el control dinámico de la red activa permite que los servicios “se ajusten a medida”, de las aplicaciones y de las condiciones cambiantes de la red. Este ajuste a medida tiene como objetivo mejorar el rendimiento percibido por las aplicaciones, en comparación con las soluciones extremo-a-extremo. En esta sección proporcionamos varios ejemplos de cómo las redes activas pueden mejorar el rendimiento de las aplicaciones.

Los esfuerzos de mejorar el rendimiento de las aplicaciones usando las redes activas cubren una amplia variedad de áreas [13]. Entre estas áreas las que parecen más prometedoras son las siguientes.

4.1. Adaptación dinámica

Un área interesante de investigación está relacionada con la adaptación dinámica a las condiciones cambiantes de la red.

Principalmente, las redes activas pueden aportar mejoras de rendimiento significativas en situaciones donde sea crucial la respuesta rápida ante cambios en la información local (en un nodo o conjunto de nodos de la red).

Por ejemplo, la calidad de servicio obtenida por una aplicación se puede degradar de forma significativa, en situaciones de congestión en la red o de enlaces con una tasa alta de pérdida de paquetes. La aproximación tradicional para resolver este tipo de situaciones ha sido que la fuente se vaya adaptando a las condiciones de la red, pero este planteamiento tiene una serie de limitaciones como es el tiempo que necesita la fuente para detectar el cambio y reaccionar en consecuencia. Al introducir en los nodos inteligencia sobre como adaptarse a los cambios en la red, se hace que la adaptación se produzca antes con el consiguiente aumento de rendimiento.

En esta área ya se han obtenido resultados interesantes en el Protocolo Boosters [14]. En este trabajo se

muestra como un protocolo se puede adaptar a condiciones tales como un aumento en la tasa de pérdidas en la red añadiendo dinámicamente funcionalidad de corrección de errores alrededor del área de la red donde se están produciendo las pérdidas.

Otros trabajos en esta área, relacionados con control de congestión son: el de Zegura y otros [15] que han propuesto estrategias inteligentes de descarte de paquetes en situaciones de congestión para preservar la calidad de vídeo MPEG en esos periodos, y el de Faber [16] que ha planteado como introduciendo en la red inteligencia para actuar en situaciones de congestión, TCP puede mejorar su rendimiento en un 18%.

4.2. Gestión de Red

La manera tradicional de realizar la gestión de una red, consiste en recolectar información de los nodos gestionados, en base a solicitar el valor de unas variables determinadas y comprobar si se detectan anomalías. Este enfoque concentra la inteligencia en las estaciones de gestión, lo que puede provocar cuellos de botella. Además esta aproximación limita seriamente la posibilidad de seguir la pista a problemas detectados, en escalas de tiempo razonables.

El principal grupo que ha realizado aportaciones en esta área ha sido BBN con su proyecto Smart Packets [4]. Este grupo ha planteado que la programación de la red hace posible que los nodos gestionados sean así mismo nodos programables. De esta forma, los centros de gestión pueden enviar programas a los nodos gestionados. Este enfoque aporta tres ventajas. Primero, la información de gestión que vuelve al centro de gestión puede ajustarse en tiempo real a los intereses que en un momento dado tenga dicho centro de gestión, reduciendo por tanto, el tráfico y la cantidad de información que debe analizar dicho centro de gestión. Segundo, muchas de las reglas de gestión empleadas por el centro de gestión pueden ser incluidas en programas que se envían al nodo gestionado, permitiéndole identificar y corregir problemas automáticamente sin la intervención del centro de gestión. Tercero, permite hacer más cortos los ciclos de monitorización y control.

4.3. Comunicaciones Multipunto

Por último finalizaremos esta sección mostrando como las redes activas pueden mejorar el rendimiento de un servicio de comunicaciones de multipunto fiable.

Después de una década investigando en protocolos de multipunto fiable ésta continúa siendo una línea de investigación abierta, debido a que las aproximaciones actuales presentan alguna de las siguientes deficiencias: (1) plantean la resolución de la problemática de aplicaciones específicas, no ofreciendo servicios genéricos que cubran un amplio rango de requisitos; (2) utilizan enfoques que les permiten escalar correctamente, cuando el número de receptores es muy elevado o cuando los participantes de la comunicación están muy alejados geográficamente, a costa de limitar el grado de fiabilidad proporcionado;

(3) no optimizan el retardo ni el ancho de banda de la red. Con objeto de resolver estas deficiencias en [17] se presenta el RMANP (*Reliable Multicast Active Network Protocol*) un protocolo de multipunto fiable que opera sobre la tecnología de redes activas.

Las redes activas pueden facilitar la provisión de un servicio de multipunto fiable que resuelva las deficiencias planteadas anteriormente. El hecho de que el número de receptores sea potencialmente muy elevado conlleva asociadas dos problemáticas: la implosión de ACKs y la implosión de NACKs. Respecto a la implosión de ACKs, el utilizar la tecnología de redes activas permite que cuando dichos paquetes atraviesen la red vayan siendo fusionados en determinados nodos activos que se encuentran en el camino entre los receptores y la fuente. La *fusión de ACKs* consiste en que el nodo activo por cada "n" ACKs que recibe, reenvía hacia el emisor un único ACK pero que lleva fusionada la información de los "n" ACKs recibidos. Con relación a la implosión de NACKs, el soporte que pueden ofrecer las redes activas consiste en realizar un filtrado de NACKs en determinados nodos que están en el camino entre los receptores y la fuente. El *filtrado de NACKs* consiste en que los nodos activos recuerdan los NACKs que han reenviado hacia la fuente, o lo que es lo mismo, los datos para los que se ha solicitado retransmisión, y cuando reciben un NACK lo reenvían si solicita otros datos distintos, o lo filtran (no lo retransmiten) si solicita datos ya pendientes de retransmisión. Con la fusión de ACKs y el filtrado de NACKs se elimina la problemática de la implosión de ACKs y NACKs, respectivamente.

Muchas aplicaciones multipunto muestran sensibilidad al retardo. Con relación a este aspecto, el soporte de las redes activas tiene dos vertientes: 1) realizar un almacenamiento, bajo la política del mejor esfuerzo posible, de los datos multipunto en determinados nodos activos en el camino entre la fuente y los receptores, 2) realizar un control intermedio de secuencia. El *almacenamiento intermedio* de los datos permite implementar un esquema de retransmisiones locales que evita que todas las retransmisiones tengan que realizarse desde la fuente. Cuando un nodo activo recibe un NACK viajando hacia la fuente, y tiene almacenado el dato solicitado, dicho nodo filtra el NACK y retransmite el dato solicitado. El realizar *retransmisiones locales* permite: 1) disminuir el retardo extremo a extremo, 2) optimizar el consumo de ancho de banda, 3) repartir la carga asociada a las retransmisiones entre la fuente y los nodos activos que llevan a cabo el almacenamiento intermedio de los datos, 4) realizar la recuperación ante diferentes errores de forma concurrente e independiente, en distintas partes de la red. El *control intermedio de secuencia* consiste en que un nodo activo lleva control de los datos multipunto recibidos y cuando detecta un salto en la secuencia genera un NACK que envía hacia la fuente. El control intermedio de secuencia permite una detección prematura de las pérdidas lo cual conlleva una disminución del retardo extremo a extremo.

Otra necesidad de algunas de las aplicaciones multipunto es el conocimiento de la identidad de los receptores por parte de la fuente. El soporte de las redes activas con relación a este aspecto está basado en la función de *agregación de información*. Los receptores pueden informar sobre su identidad enviando ACKs hacia la fuente. Dada la posibilidad de las redes activas de realizar una agregación de estos ACKs, la fuente conocerá la identidad de sus receptores sin que se produzca implosión. La fuente en lugar de procesar un ACK por cada receptor procesará un número reducido de ACKs, cada uno con información relativa a muchos receptores.

Adicionalmente a los requisitos específicos de las aplicaciones multipunto, todo servicio de comunicaciones debe tener como objetivo el optimizar el uso de recursos de la red. Las redes activas ofrecen en este sentido la posibilidad de realizar retransmisiones de ámbito restringido y filtrado de retransmisiones. La *retransmisión de ámbito restringido* evita la retransmisión de los datos solicitados a través de todas las interfaces de un nodo activo. En este caso, el soporte de los nodos activos consiste en recordar a través de que interfaces se solicitó un paquete de datos, restringiendo la retransmisión de ese paquete a dichas interfaces. Los nodos activos utilizan el *filtrado de retransmisiones* para evitar el envío de múltiples retransmisiones del mismo paquete de datos, cuando éstas han sido pedidas en paralelo por un conjunto de receptores o nodos activos que cuelgan del mismo interfaz.

El RMANP ha sido implementado sobre ANTS [7] habiéndose realizado un análisis de los requisitos de memoria en los nodos activos, y de los tiempos de proceso necesarios para procesar los paquetes activos RMANP en los nodos activos [18].

5. Conclusiones

Aquí hemos presentado una panorámica de la tecnología de redes activas y hemos discutido algunas de sus potenciales aplicaciones.

Las redes activas son un área investigación prometedora que en estos momentos se encuentra en una fase relativamente inmadura debido a que todavía no se ha realizado la suficiente experimentación y son muchas las líneas de investigación abiertas. Sin embargo, se han identificado dos retos que si no son superados de forma satisfactoria, comprometerán el éxito de la tecnología:

- *La arquitectura de seguridad*. El principal problema de los mecanismos de seguridad tradicionalmente utilizados es su coste, en tiempo y computación, que los hacen inviables para una red activa en producción, de aquí que la pregunta, aún sin resolver, sea: ¿Qué clase de mecanismos de seguridad son necesarios para poder seguir la pista de los permisos asignados a millones de usuarios en miles de encaminadores activos desplegados por todo el globo?

- El *rendimiento*. En los modelos de redes activas propuestos hasta el momento se pueden identificar dos tendencias: una primera, liderada por el MIT con su ANTS, consistente en ofrecer un API de red muy flexible que permita programar casi cualquier cosa aún a costa de obtener rendimientos bastante pobres, y una segunda, liderada por la Universidad de Pennsylvania con su Switchware, consistente en utilizar un API de red muy restringido y hacer que la red se programe esencialmente en base a las extensiones activas introducidas en el nodo por el proveedor de red, obteniendo buenos rendimientos a costa de limitar la flexibilidad. Quizá la solución pase por buscar un compromiso entre la flexibilidad y el rendimiento una vez resuelta la cuestión clave: ¿qué tipos de flexibilidad son realmente útiles?

Referencias

- [1] D.L. Tennenhouse, J.M. Smith, W.D. Sincoskie, D.J. Wetherall and G.J. Minden. "*A Survey of Active Network Research*". IEEE Communications Magazine, pp. 80-86, January 1997.
- [2] S. Alexander. "*A Generalized Computing Model of Active Networks*". Ph.D. U. de Pennsylvania. December 1998.
- [3] M. Hicks, J. T. Moore, D. S. Alexander, C. A. Gunter, S. M. Nettles. "*PLAN: A Programming Language for Active Networks*". Proceedings of International Conference on Functional Programming (ICFP) '98, September 1998.
- [4] B. Schwartz, W. Zhou, A. W. Jackson, W. T. Strayer, D. Rockwell and C. Partridge. "*Smart Packets for Active Networks*". January 1998. <http://www.net-tech.bbn.com/smtpkts/smart.ps.gz>
- [5] Kenneth L. Calvert, Samrat Bhattacharjee, Ellen W. Zegura and James Sterbenz. "*Directions in Active Networks*". IEEE Communications Magazine, pp. 72-78, October 1998.
- [6] Y. Yemini and S. Da Silva. "*Towards Programmable Networks*". IFIP/IEEE International Workshop on Distributed Systems: Operations and Management. L'Aquila. Italy, October 1996. <http://www.cs.columbia.edu/~dsilva/netscript.html>
- [7] D. Wetherall, J. Gutttag and D.L. Tennenhouse. "*ANTS: A Toolkit for Building and Dynamically Deploying Network Protocols*". Proceedings of IEEE OPENARCH'98, San Francisco, CA, April 1998.
- [8] Samrat Bhattacharjee, Kenneth L. Calvert and Ellen W. Zegura. "*Reasoning About Active Network Protocols*". Proceedings of ICNP '98, Austin, TX, October 1998.
- [9] S. Alexander, W. A. Arbaugh, M. W. Hicks, P. Kakkar, A. D. Keromytis, J. T. Moore, C. A. Gunter, S. M. Nettles and J. M. Smith. "*The SwitchWare Active Network Architecture*". IEEE Network, Special Issue: Active and Programmable Networks, 12(3):29-36, May/June 1998.
- [10] S. Alexander, W. A. Arbaugh, A. D. Keromytis and J. M. Smith. "*A Secure Active Network Environment Architecture: Realization in SwitchWare*". IEEE Network, Special Issue: Active and Programmable Networks, 12(3):37-45, May/June 1998.
- [11] Programa DARPA de redes activas. <http://www.darpa.mil/ito/research/anets>
- [12] S. Alexander, B. Braden, C. A. Gunter, A. W. Jackson, A. D. Keromytis, G. J. Minder and D. Wetherall. "*Active Network Encapsulation Protocol (ANEP)*". RFC DRAFT, 1997.
- [13] U. Legedza, D. Wetherall and J. Gutttag. "*Improving the Performance of Distributed Applications Using Active Networks*". Proceedings of IEEE INFOCOM'98, San Francisco, April 1998.
- [14] D. Feldmeier, A. McAuley, J. Smith, D. Bakin, W. Marcus and T. Raleigh. "*Protocol booster*". IEEE Journal on Selected Areas in Communications, 16(3):437-444, April 1998.
- [15] Samrat Bhattacharjee, Kenneth L. Calvert and Ellen W. Zegura. "*An Architecture for Active Networking*". Proceedings of High Performance Networking (HPN'97), White Plains, NY, April 1997
- [16] T. Faber. "*ACC: Using Active Networking to Enhance Feedback Congestion Control Mechanisms*". IEEE Network, Special Issue: Active and Programmable Networks, 12(3):61-65, May/June 1998.
- [17] M. Calderón, M. Sedano, A. Azcorra and C. Alonso. "*Active Network Support for Multicast Applications*". IEEE Network, Special Issue: Active and Programmable Networks, 12(3):46-52, May/June 1998.
- [18] M. Sedano Ruiz. "*Soporte de Redes Activas para un servicio integrado de multipunto fiable con control de Congestión*". Ph.D. Facultad de Informática, Universidad Politécnica de Madrid. Febrero 1999.

Especificación y Verificación Formal de un Protocolo de Gestión de Sistemas de Administración de Redes distribuido con árbitro

Mario Marrero y Alvaro Suárez
GAC (Grupo de Arquitectura de Computadores)
Dpto. de Ingeniería Telemática Universidad de Las Palmas de G.C.
Campus Universitario de Tafira (Aula Magna), 35017 Las Palmas de G.C.
Telf: 928-45-12-39, Fax: 928-45-12-43
email: mario@cicei.ulpgc.es alvaro@cic.teleco.ulpgc.es

Abstract

In this paper different network management architectures and their problems are discussed. In order to solve these problems we propose a new architecture based on distributed entities called SARDs. Furthermore, a new application protocol that communicates and synchronizes those entities is specified (PSSARD). We present the formal specification (using different tools, part of them implemented by our research group), and the verification of some fundamental properties of the protocol. We have proved that our protocol is efficient and it is safe. At this moment we develop a management tool following this protocol.

1. Introducción

La gestión de sistemas informáticos se puede definir como aquel proceso encargado de controlar una red de datos compleja con el propósito de maximizar su eficiencia y productividad. Un Sistema de Administración de Redes [1] es un conjunto de hardware y software diseñados para realizar la tarea fundamental de la gestión de sistemas informáticos.

En la actualidad, en muchos casos, la gestión de los sistemas informáticos es realizada por un administrador de forma manual o utilizando herramientas parcialmente automatizadas. Muchas empresas de venta de computadores disponen de este tipo de herramientas basadas, en general, en arquitecturas centralizadas que gestionan sistemas distribuidos. Esto es una contradicción. Todos los autores coinciden en que es necesario mucho trabajo de investigación en este campo para definir nuevos protocolos de administración.

En este artículo se exponen las principales arquitecturas de sistemas de administración existentes así como las principales dificultades que presentan y se justifica la creación de una nueva arquitectura acorde a las redes actuales.

La entidad fundamental en la nueva arquitectura es el SARD (Sistema de Administración de Redes Distribuido). Dentro de un sistema global de administración pueden existir n SARDs, dedicándose al control de diferentes dominios definidos dentro de la red. Para poder comunicar y sincronizar dichas entidades se define un protocolo de aplicación denominado PSSARD (Protocolo de Sincronización de SARD). También presentamos todos los pasos de diseño de dicho protocolo partiendo de una especificación informal del mismo, formalizando los requerimientos y verificándolos.

El análisis del PSSARD se realiza utilizando Técnicas de Descripción Formal. Estas técnicas son

herramientas matemáticas que aseguran la precisión y tratabilidad de las descripciones de sistemas. Una de estas técnicas es el lenguaje LOTOS [2]. Para poder realizar una especificación en LOTOS visual [3], utilizamos las herramientas SGLLOT [4] y GRLOT [5], mientras que para su verificación utilizamos el LOLA [6] y el ARA tools [7].

El artículo está dividido en 5 apartados. En el apartado 2 se describen las distintas arquitecturas de un sistema de administración de redes, así como la arquitectura propuesta. En el apartado 3 se presenta informalmente la entidad SARD y el protocolo que los sincroniza y comunica (PSSARD). En el apartado 4 se describen los pasos de diseño para verificar el protocolo PSSARD partiendo de los requerimientos informales. Adicionalmente se describen dichos pasos en profundidad para el nivel de abstracción más sencillo. Por último se presentan unas conclusiones y líneas de trabajo futuro.

2. Sistemas de Administración de Redes

Un Sistema de Administración de Redes es un conjunto de hardware y software diseñados para realizar la tarea fundamental de la gestión de sistemas informáticos.

En la Fig. 1 vemos un diagrama básico de las dos partes de todo sistema de administración. La *arquitectura* del sistema define la estructura de los módulos hardware, mientras que la *aplicación* define la disposición del software del sistema sobre esa arquitectura.

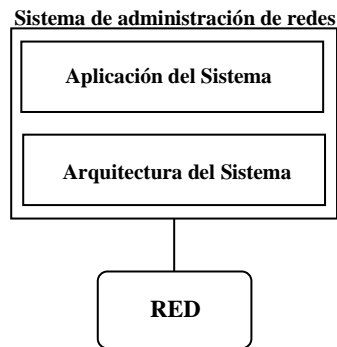


Fig 1. Diagrama básico de un sistema de administración de redes.

2.1. Arquitectura de un Sistema de Administración de Redes

El diseño de un sistema de administración de redes es una tarea compleja. El punto de partida debe ser la definición de la arquitectura que utiliza ese sistema, para posteriormente especificar e implementar las aplicaciones que se ejecutan sobre esa arquitectura.

No existen reglas definidas para el diseño de la arquitectura. Sin embargo, teniendo en cuenta las especificaciones que debe satisfacer este sistema, podemos elegir entre las siguientes generales:

- *Arquitectura centralizada.* Existe un sistema potente central que ejecuta la mayoría de las aplicaciones del sistema. Este sistema se encarga de extraer la información de la red usando un determinado protocolo de administración, y mantener las bases de datos. Cada uno de los sistemas críticos de la red realizan una serie de funciones, cuyos resultados pueden ser accedidos por el sistema central gracias a un agente. Ventajas de esta arquitectura son: Sencillez a la hora de implementarla, fácil manejo y bajo coste. No obstante, tiene una serie de dificultades [3]: a) Existe un *problema con la topología de la red*. Cualquier caída en un router hace que el sistema de administración no obtenga ningún dato del otro lado del router, b) Provoca *cuelllos de botella en el ancho de banda*, c) Los *enlaces con bajo ancho de banda* en la red (enlaces lentos) pueden confundir al sistema de administración (timeouts).
- *Arquitectura distribuida.* Las aplicaciones del sistema de administración están repartidas por la red sobre distintos sistemas que deben estar funcionando concurrente y distribuidamente. Bajo esta arquitectura, cada pequeño sistema se puede encargar de la administración de una determinada parte de la red (dominio). La ventaja fundamental de esta arquitectura es que presenta un mayor rendimiento (existen aplicaciones de administración funcionando en paralelo). No obstante, un sistema puramente distribuido es

complejo de administrar. El protocolo que interconecta los sistemas de administración debe sincronizarlos adecuadamente y esto no es una tarea sencilla. De igual forma, es complicada la realización del paso de información desde un sistema a otro cuando la aplicación lo requiera.

- *Arquitectura jerárquica.* En este caso se introduce el concepto de administrador de administradores (manager of manager MOM), en cual se sitúa en lo alto de la jerarquía y administra al resto de servidores o administradores. Dentro de esta jerarquía también se introduce el concepto de *dominio*. Esta arquitectura presenta como ventajas un alto rendimiento, escalabilidad y control sencillo. Algunas desventajas de esta arquitectura son: Bajo rendimiento de algunos sistemas. En el caso concreto del MOM, sólo se utiliza para coordinar al resto de sistemas de administración.

2.2. Sistema de administración de redes basado en una arquitectura distribuida con árbitro

La arquitectura que se adopta en este artículo es una solución híbrida entre la arquitectura jerárquica y la distribuida, de forma que se obtienen las ventajas de ambas. Nosotros hemos considerado un esquema jerárquico cuya topología se muestra en la Fig. 2.

En esta arquitectura, inicialmente, todos los sistemas están situados al mismo nivel de jerarquía. Por lo tanto, cada uno se ocupa de un dominio de agentes. A cada uno de estos sistemas los denominamos **SARDs** (Sistema de Administración de Redes Distribuido). Adicionalmente, uno de estos SARDs actúa como árbitro, puede ser cualquiera de los ellos, y sólo uno de ellos puede estar realizando las funciones de árbitro. Además llamamos **SGA** (Sistema Global de Administración) a todo el conjunto o cluster de SARDs.

3. SARD y PSSARD. Especificación informal

En la Fig. 3 podemos observar los distintos subsistemas que componen un SARD:

- *Subsistema de Acceso al Agente (SAA):* Incluye todos aquellos módulos que actúan sobre el protocolo SNMP y atacan a los agentes obteniendo los datos propios de su dominio.

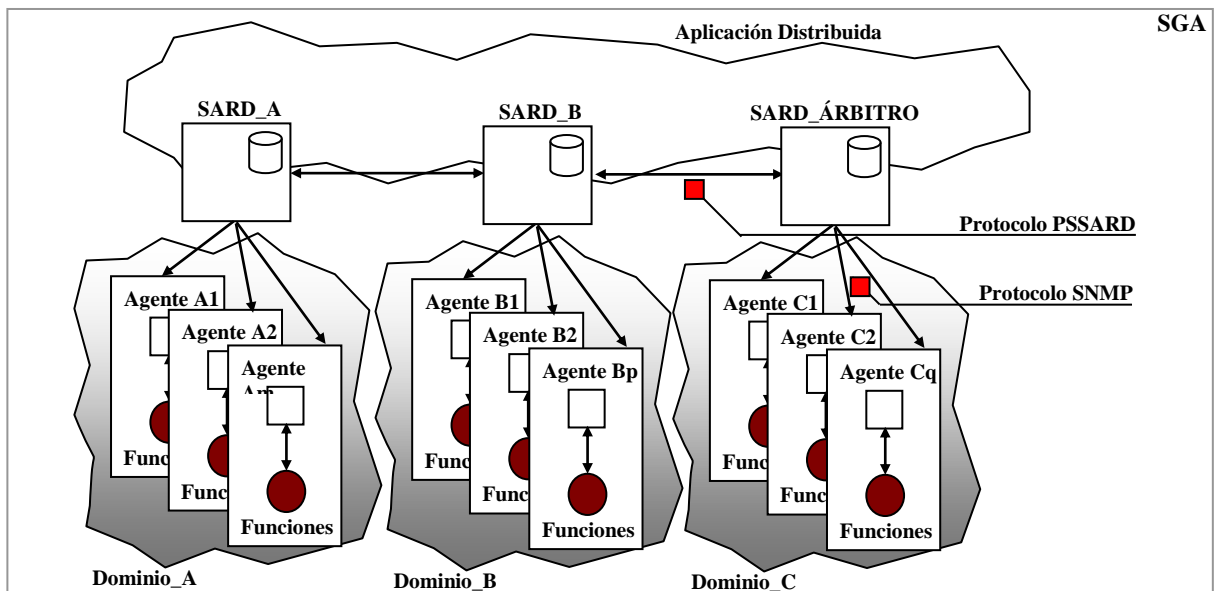


Fig 2. Sistema global de administración (SGA) distribuido con árbitro.

- *Subsistema de Aplicación (SDA)*: Incluye todos aquellos módulos encargados de realizar las partes de la aplicación distribuida correspondiente a este SARD. Cada SARD debe realizar una serie de procedimientos o funciones dentro de su dominio, asignados por el árbitro en funciones. A estos procedimientos los denominaremos PAD (Procedimientos de administración sobre dominio).

- *Subsistema de Sincronización y Autonomía (SSA)*: Encargado de implementar el protocolo PSSARD, mediante el cual se comunica y sincroniza con el resto de SARDs, incluido el árbitro. Adicionalmente, dependiendo de las condiciones del entorno, este subsistema puede hacer que el SARD se convierta en árbitro del sistema global.

- *Control*: Encargado de coordinar los subsistemas anteriores.

3.1. Protocolo PSSARD (sincronización entre SARDs)

Una vez analizados los subsistemas en que se dividen los SARDs, nos centramos ahora en el análisis del protocolo que sincroniza y comunica los distintos SARDs. Como vimos anteriormente, este protocolo está implementado con los subsistemas de sincronización y autonomía de cada uno de los SARDs.

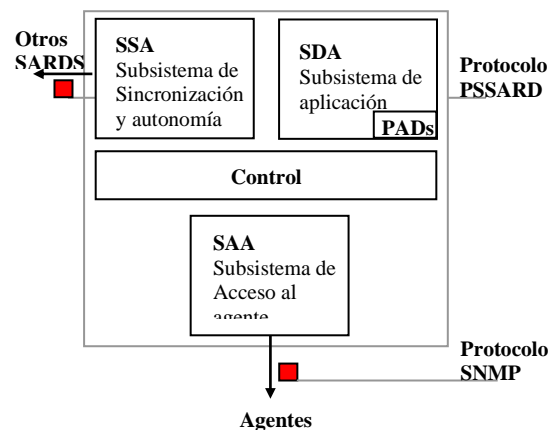


Fig 3. Subsistemas de un SARD.

Informalmente, el protocolo PSSARD se divide en tres fases críticas:

- *Fase de inicialización*: Esta fase se realiza cuando se inicia el sistema por primera vez, o tras un fallo producido por la caída de algún SARD (incluyendo al árbitro).

- *Fase de operación normal*: En esta fase, los SARDs operan normalmente realizando las partes de la aplicación distribuida que le correspondan, es decir realizan los PADs que le han sido asignados.

- *Fase de test de SARDs*: Donde se detecta si algún SARD ha caído o ha dejado de funcionar correctamente.

Un diagrama de estados muy abstracto para el PSSARD está representado en la Fig. 4. Veamos cada una de estas fases por separado.

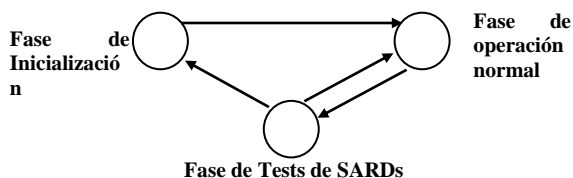


Fig 4. Diagrama de estados del PSSARD a alto nivel.

Fase de inicialización

Los pasos que se realizan en esta fase son los siguientes:

- En un tiempo inicial $t=0$, se inicializan todos los SARDs. Todos y cada uno de ellos entran en el modo ready, indicando que están listos para operar como SARD (árbitro o resto, indistintamente) (Fig. 5).

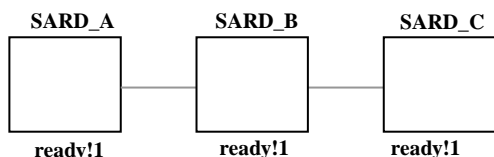


Fig 5. Cluster de tres SARDs indicando que están preparados.

- Sobre uno de los SARDs se actúa convirtiéndolo en SARD_árbitro (Fig. 6).
- El árbitro envía un broadcast solicitando qué SARD está ready. El árbitro espera un determinado tiempo durante el cual puede recibir respuestas.
- El árbitro, confirma a cada SARD enviándole un único identificador a cada uno de ellos. Este identificador establece, adicionalmente, un orden de prioridades para realizar las funciones de árbitro.
- El árbitro, en función del número de SARDs disponibles en el cluster, realiza una distribución de PADS. Cada PAD tiene asignado un determinado coste, de forma que la distribución es lo más equitativa posible.

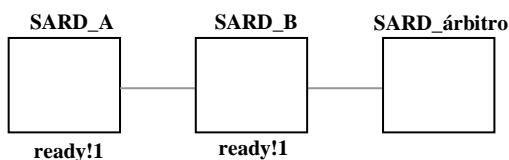


Fig 6. Uno de los SARDs toma las funciones de árbitro.

- El árbitro transfiere unas tablas de asignación de PADS a cada uno de los SARDs. Cada una de estas tablas contienen la información de aquellos PADS que

ejecuta cada uno de los SARDs en el cluster. Los campos de las tablas de asignación son los siguientes: a) Identificador: Es un número entero que identifica a cada uno de los SARDs, b) PADS_x: Es una lista de enteros que identifica los PADS asociados al SARD x. Tras este paso, los SARDs contienen unas tablas como las indicadas en la Fig. 7.

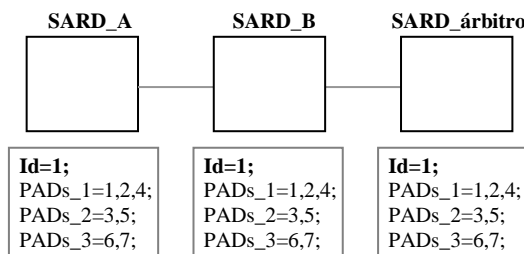


Fig 7. Todos los SARDs han obtenido las tablas de asignación de PADS. Finaliza la fase de inicialización.

Fase de operación normal

Durante esta fase, el SARD_árbitro realiza funciones distintas al resto de SARDs del sistema. La secuencia de pasos que debe realizar el árbitro es:

- Recibir peticiones de ejecución de algún PAD por parte de la aplicación distribuida. Estas peticiones pueden llegar al árbitro desde cualquier máquina de su dominio o de otro.
- Decodificar la petición y ejecutar si corresponde a su propio dominio.
- Si la petición pertenece a otro dominio, distribuirla hacia el SARD correspondiente.

La secuencia de pasos que deben realizar el resto de SARDs son:

- Esperar a recibir una orden desde el árbitro.
- Ejecutar dicha orden.

Fase de test de SARDs

Adicionalmente a la fase anterior, los SARDs deben ejecutar un procedimiento de test de SARDs, mediante el cual se realizan dos comprobaciones:

- Detectar si algún SARD ha dejado de funcionar correctamente. En este supuesto se retorna a la fase de inicialización donde el sistema se debe autoreconfigurar.
- Detectar si existe algún nuevo SARD que desea unirse al cluster. En este caso también se procede hacia la fase de inicialización para la autoreconfiguración.

4. Especificación formal y verificación. Niveles de abstracción.

A la hora de realizar la especificación formal del sistema, distinguimos tres niveles de abstracción. A medida que se avanza en el nivel de abstracción, se va añadiendo complejidad al sistema bajo estudio. Estos niveles son:

- *Nivel 0* (Menor complejidad). Sus características principales son: a) Se realiza una primera abstracción a la estructura global del sistema, b) Se abstrae todo el comportamiento de cada una de las tres fases que componen el funcionamiento del protocolo, c) Se profundiza en el sistema de control del protocolo, d) No se introducen tipos de datos abstractos, por lo tanto, se realiza íntegramente en Basic LOTOS.
- *Nivel 1* (Complejidad media). Sus características principales son: a) Se profundiza en cada una de las estructuras y comportamientos internos de las tres fases del protocolo, b) Se añaden tipos de datos abstractos sencillos (naturales y booleanos).
- *Nivel 2* (Mayor complejidad). Sus características son: a) Mayor profundización en el comportamiento de las fases del protocolo, b) Se introducen tipos de datos abstractos complejos (pilas, etc).

Una vez realizada la especificación informal del sistema, los pasos a realizar para cada uno de los niveles de abstracción son: a) Realización de la especificación estructural del sistema. En esta parte analizamos los procesos que entran en juego, sus puertas y la relación entre estos. Para esta parte utilizamos la herramienta SGLot, b) Realización del comportamiento interno. En esta parte se especifica el comportamiento interno (utilizando los operadores LOTOS), de cada uno de los procesos anteriores. Para la realización de esta parte hacemos uso de la herramienta GRCLot, c) Análisis lógico (informal) del comportamiento del sistema (de una forma estática), d) Verificación y análisis de las propiedades fundamentales. Haciendo uso de las herramientas ARA y LOLA, analizamos las propiedades de la especificación realizada y e) Refinamiento y repetición de los pasos anteriores. Se repiten los pasos anteriores tantas veces como sea necesario hasta obtener un sistema con las propiedades requeridas.

4.1. Especificación nivel 0

Dentro de este nivel de abstracción se analizan los procesos y puertas necesarias para realizar el control de las distintas fases del protocolo. Es, por tanto, un nivel de abstracción de sólo control donde se ignoran por completo los tipos de datos.

4.1.1. Especificación estructural

En este nivel, disponemos de los siguientes elementos estructurales:

- *Proceso Fase de Inicialización*: Realiza la primera de las fases del protocolo PSSARD. Contiene las puertas i_fase_1 , f_fase_1 e *inicialización*. Las dos primeras puertas indican el inicio y la finalización de esta fase, respectivamente. La puerta *inicialización* es una abstracción de todas las acciones que se deben realizar.
- *Proceso Fase de Operación normal*: Realiza la segunda de las fases del protocolo PSSARD. Contiene las puertas i_fase_2 , f_fase_2 y *ejecutar*. De forma similar a la fase anterior, la puerta i_fase_2 indica el comienzo de la fase mientras que la puerta f_fase_2 indica su finalización. La puerta *ejecutar* es una abstracción.
- *Proceso Fase de Test*: Realiza la última de las fases del protocolo. Sus puertas son i_fase_3 , f_fase_3 y *test*.
- *Proceso Control del PSSARD*: Realiza el control de las tres fases anteriores. Sus puertas son i_fase_1 , f_fase_1 , i_fase_2 , f_fase_2 , i_fase_3 y f_fase_3 .

En la Fig. 8 se presenta la relación entre estos procesos, realizada con la herramienta SGLot.

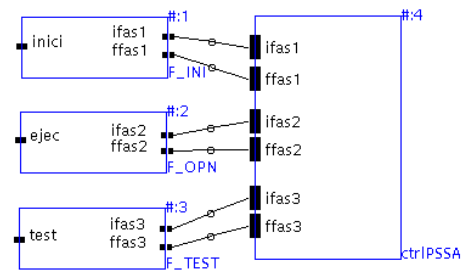


Fig 8. Especificación estructural del protocolo PSSARD a nivel 0 (menor complejidad).

La herramienta SGLot nos devuelve el código LOTOS textual de la estructura, la cual debe ser rellenada con el comportamiento de cada uno de los procesos.

4.1.2 Comportamiento interno del sistema

Utilizando la herramienta GRCLLOT especificamos gráficamente el comportamiento de cada uno de los cuatro procesos vistos anteriormente.

El comportamiento general del sistema se representa en la Fig. 9. Existen tres bandas de activación horizontal (BAH) sobre las cuales se lanza en paralelo los procesos *Fase_inicializacion*, *Fase_operacion_normal* y *Fase_test*, sincronizándose con el proceso *Control_PSSARD*.

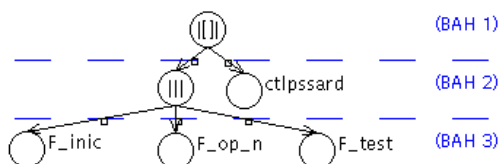


Fig 9. Comportamiento general del sistema.

En la Fig. 10 podemos visualizar el comportamiento del proceso *Control_PSSARD*, el cual realiza el control de las tres fases del protocolo vistas anteriormente mediante la habilitación del inicio y final de las mismas.

El proceso *Control_PSSARD* consta de los siguientes subprocesos (ejecutándose de forma secuencial):

- *activar_fase_inicializacion*: Activa el inicio y fin de la fase de inicializacion. Posteriormente acaba su ejecución (*exit*).
- *activar_fase_operacion_normal*: Activa el inicio y fin de la fase de operación normal. Posteriormente, dependiendo de un evento interno al proceso, se vuelve a ejecutar la fase de operación normal o se acaba el subproceso (*exit*). Esto último se debe interpretar como un *timeout* para acceder al siguiente subproceso en el comportamiento general de *Control_PSSARD*, es decir, *activar_fase_test*.
- *activar_fase_test*: Activa el inicio y fin de la fase de test. A continuación y dependiendo de un evento interno, se procede hacia el subproceso de activación de la fase de inicialización (test fallido) o activación de la fase de operación normal (test con éxito).

4.1.3 Análisis lógico de las propiedades del sistema (informal)

La especificación al nivel 0 debe cumplir con las siguientes propiedades fundamentales: a) *Propiedad (1)*. No puede existir ningún *deadlock*, b) *Propiedad (2)*. Los cuatro procesos involucrados en el protocolo deben estar continuamente en ejecución

(infinitas posibles transiciones) y c) *Propiedad (3)*. Se exige el cumplimiento del diagrama de estados para las fases expuesto en la Fig. 4.

4.1.4. Verificación de las propiedades fundamentales

Una vez obtenido el código textual de la especificación LOTOS global, pasamos a verificar sus propiedades utilizando las herramientas LOLA y ARA. Gracias al LOLA podemos realizar un seguimiento paso a paso de la expansión de la especificación

Utilizando las herramientas ARA se obtiene un grafo del LTS (Label Transition System) generado por el sistema completo. Dicho LTS se representa en la Fig. 11.

De la visualización del LTS y del análisis de los datos numéricos de la expansión se extraen las siguientes conclusiones positivas:

- En la expansión se contemplan las tres fases, siguiendo un orden de secuencialidad correcto.

Por otro lado, se extraen las siguientes conclusiones negativas:

- Existen tres *deadlocks* (puntos donde el LTS no posee salida alguna). Por lo tanto no se cumple con la propiedad *Propiedad (1)*.
- La causa de los *deadlocks* es la inexistencia de recursividad en los procesos *Fase_inicializacion*, *Fase_operacion_normal* y *Fase_test*. Una vez llegan al final dichos procesos, se paran. Por lo tanto no cumple con la propiedad *Propiedad (2)*.
- *No existe un bucle de ejecución para la fase de operación normal. Sólo se ejecuta una única vez, lo cual no está de acuerdo con los requerimientos de la especificación informal. Esto no está de acuerdo con la propiedad Propiedad (3)*.

Por lo tanto, la especificación realizada no es válida y es necesario introducir un segundo nivel de refinamiento.

4.1.5. Sucesivos refinamientos

Tras realizar la verificación formal de procede a un refinamiento del mismo con el objetivo de eliminar los *deadlocks* en la especificación anterior. Para esto se modifican los procesos *Fase_inicializacion*, *Fase_operacion_normal* y *Fase_test*, introduciendo recursividad.

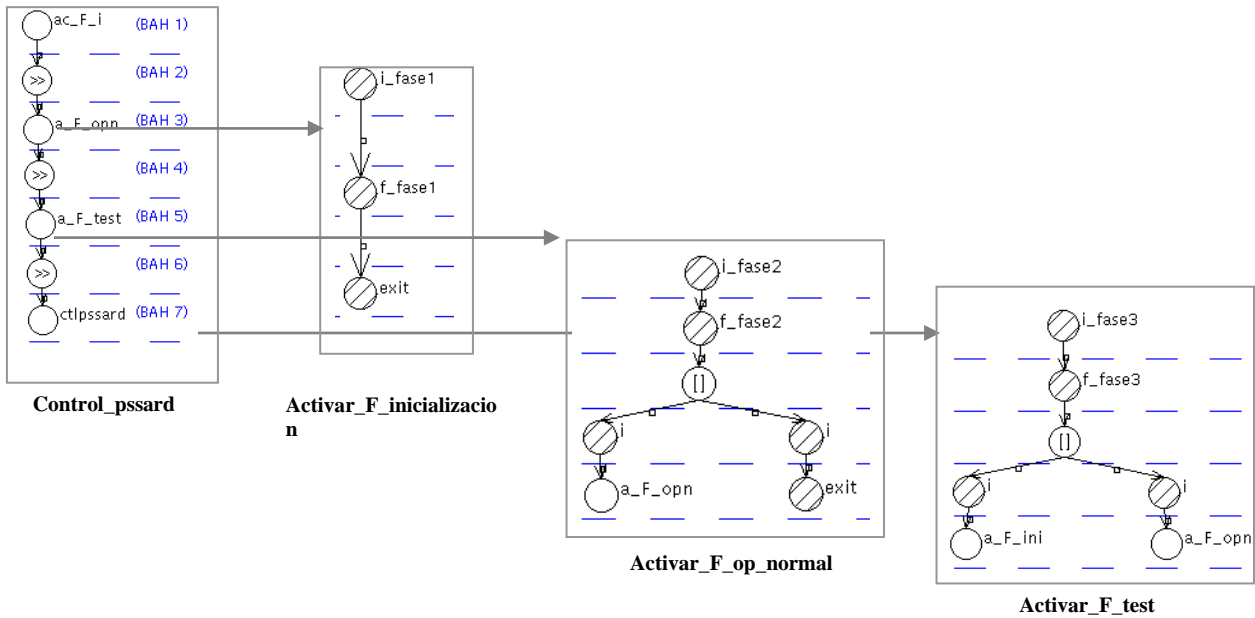


Fig 10. Comportamiento del proceso Control_PSSARD.

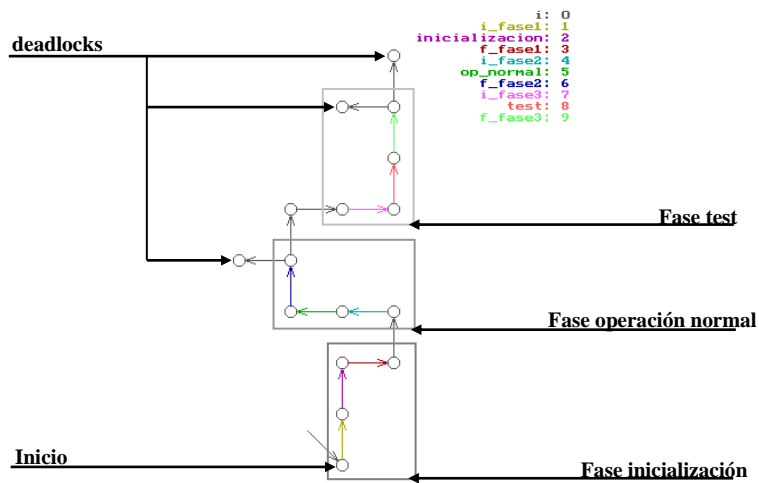


Fig 11. LTS de la especificación pssard1.lot.

Tras varios refinamientos se obtiene el grafo de comportamiento de la Fig. 12, correspondiente al proceso Control_PSSARD.

Del análisis del LTS para esta especificación obtenemos las siguientes conclusiones positivas: a) Se cumplen todas las propiedades requeridas en la especificación informal y b) Se ha realizado una optimización del LTS resultante donde obtenemos un menor número de estados y una reducción de los estados duplicados.

A partir de aquí se procede a la disminución del nivel de abstracción de forma que se consideren los requerimientos más importantes a verificar del protocolo.

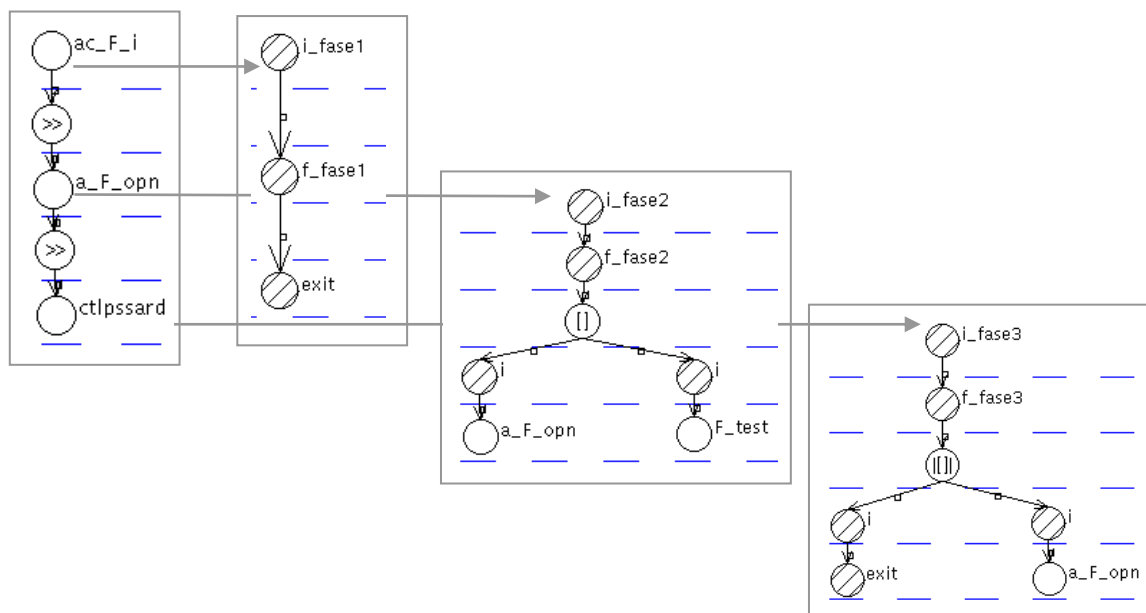


Fig 12. Comportamiento del proceso Control_PSSARD (pssard4).

5. Conclusiones y líneas de trabajo futuro

En este artículo se han presentado las arquitecturas de sistemas de administración de redes más usuales en la actualidad. No obstante, dichas arquitecturas presentan una serie de inconvenientes y problemas que se deben tener en cuenta. Dichos problemas justifican la definición de una nueva arquitectura basada en entidades administradoras de dominios o SARDs las cuales se comunican y sincronizan mediante el protocolo de aplicación PSSARD.

Gracias a las técnicas de descripción formal se ha conseguido especificar y verificar el comportamiento de este protocolo sin llegar a niveles de implementación. En los tres niveles de abstracción realizados y durante sus fases de verificación se han detectado problemas (deadlocks, secuencias de acciones redundantes, etc) las cuales hubiesen provocado una implementación inconsistente y no optimizada.

Las líneas de trabajo actuales pasan por la implementación de una herramienta que siga las pautas definidas en este artículo.

Referencias

[1] Aiko Pras. *Network Management Architectures*. Tesis. Centre for Telematics and Information Center. University of Twente. Holanda. 1995.

[2] Bolognesi T., Brinksma E. *Introduction to the ISO specification language LOTOS*. Computer Networks ISDN systems, pp. 25-59, 1987.

[3] Mario Marrero, Alvaro Suárez, *A tool for visualizing LOTOS Behavioural Specifications*, 4th FTRTFT, LNCS 1135, pp. 475-478, Sep. 1996.

[4] Mario Marrero, Alvaro Suárez, Elena Carrión and Elsa Macías, *SGLLOT: A visual tool for structural LOTOS specifications*, 5th FTRTFT, LNCS, Lyngby, Sep. 1998.

[5] Elisa Barrena, *Generador de Código LOTOS desde Grafos de Comportamiento (GRCLLOT)*, Proy. Fin Carrera, EUITT-ULPGC, Julio 1998.

[6] S. Pavón, D. Larrabeiti, G. Rabay, "*LOLA: LOTOS Laboratory*", Departamento de Ingeniería Telemática. Universidad Politécnica de Madrid. Febrero-1995.

[7] AATOS Research Group, "*ARA Tools 2.0, A set of tools for analysing the behaviour of concurrent systems*", Electronics Jukka Kemppainen, Agosto-1994.

Dimensionado del Nodo de Acceso con Control Explícito de Tasa DMRCA para Redes de Paquetes*

Xavier Hesselbach Serra, Sebastià Sallent Ribes
Departamento de Matemática Aplicada y Telemática
Universidad Politécnica de Cataluña
C/ Jordi Girona, 1 y 3 – Mòdulo C3 - Campus Nord - 08034 Barcelona
Telf: 93-4015987, Fax: 93-4015981
E-mail: {xavierh, sallent}@mat.upc.es

Abstract

Deployment of multimedia services over packet networks requires certain Quality of Service (QoS) guarantees. Statistical multiplexing in the access node reduces the burstiness and bandwidth utilization but does not control the traffic generated. Therefore, traffic overload can damage the QoS, or available resources are wasted when underload. In order to improve and manage the QoS, a source rate control algorithm is presented. This algorithm is implemented in a intelligent marking DMRCA-based access node, assuring fair rate allocation and low packet loss rate when available bandwidth is reported from the network and sources adapt properly their rate. In this paper the influence of the algorithm parameters on the QoS is studied and dimensioning criteria presented according to simulations for greedy sources. Consequences in the access node delay and required buffer length are derived for specific packet loss rate

1. Introducción

En las actuales redes de paquetes tales como Internet, la red asigna recursos a los usuarios de acuerdo a las disponibilidades del momento, sin ningún tipo de reserva o negociación. Este tipo de servicio se ha dado en llamar best-effort por ofrecer lo mejor que en cada instante es posible obtener sin disponer de reserva alguna. Esta limitación no es impedimento para la transmisión de tráfico de naturaleza multimedia, con la esperanza de que la degradación producida por el retardo y la tasa de pérdidas de los paquetes sea aceptable por el usuario, con el consiguiente bajo coste en la conexión. La literatura ofrece mecanismos para la reserva de recursos basados en la prospección de la red con el fin de establecer un camino donde se asegure un cierto ancho de banda, como por ejemplo el protocolo RSVP [6]. Sin embargo, una vez realizada la reserva, el caudal no empleado queda sin uso.

A medio camino entre best-effort y la reserva de recursos se encuentra lo que se puede denominar “mejor que best-effort”, que intenta asegurar una cierta calidad de servicio siempre que la red ofrezca información permanentemente actualizada del ancho de banda disponible. En este marco, la gestión del tráfico generado y el control de la tasa pueden asegurar cotas de calidad, siempre que las fuentes de tráfico se comporten de acuerdo a los recursos existentes ofrecidos por la red. El papel de regulador puede ser llevado a cabo por un nodo de acceso especializado, responsable de administrar el caudal disponible en la red y controlar el tráfico generado por las fuentes.

Existen diversos algoritmos que desempeñan la labor de control de tasa. Entre ellos, el EPRCA (Enhanced Proportional Rate Control Algorithm) y MMRCA (Max-Min Rate Control Algorithm) [5], que ofrecen solución a la asignación de tasa justa [1] pero que son imperfectos ante conexiones estranguladas. Para resolver este inconveniente surgió la propuesta del DMRCA (Dynamic Max Rate Control Algorithm) [3]. Estos algoritmos presentan en común la idea del marcado inteligente, consistente en la indicación a la fuente de la disponibilidad o carencia de recursos en función del estado de la cola de espera de acceso a la red. Para ello se efectúa un etiquetado o marcado de paquetes de exploración, siguiendo criterios de nivel de congestión y tasa transmitida en dicha fuente. Con este fin se definen dos umbrales (QT y DQT) sobre una cola común de recepción de paquetes procedentes de todas las tributarias, lo cual permite establecer 3 estados: Sin congestión, congestión moderada y alta congestión, tal como muestra la fig.1. En cada uno de ellos, el comportamiento del algoritmo difiere, siendo más permisivo cuanto menor sea el nivel de congestión. Más adelante se abordará con detalle el funcionamiento del algoritmo de DMRCA adaptado a un nodo de acceso.

Este artículo estudia la relación existente entre los valores de los umbrales QT y DQT sobre el retardo, el tamaño de la cola de espera para cada fuente tributaria (dentro de una misma cola compartida) y la probabilidad de pérdida.

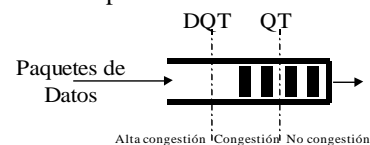


Fig.1. Estados de la cola de espera común del nodo de acceso.

* Este trabajo ha sido parcialmente financiado por CICYT TIC98-0495-C02-01.

Como entorno de aplicación, el estudio se efectuará sobre servicio ABR en redes ATM. El servicio ABR proporciona mecanismos de realimentación hacia las fuentes para informar del ancho de banda disponible y constituye por tanto un marco adecuado para la obtención de resultados reales. Se emplearán fuentes persistentes (greedy) que presentan ávidas condiciones de utilización del enlace.

Esta ponencia ha sido organizada de la siguiente manera: La sección 2 expone el mecanismo básico de notificación al nodo de acceso del caudal disponible en la red. La sección 3 revisa el funcionamiento del algoritmo DMRCA y las mejoras aportadas frente EPRCA y MMRCA, tal como se emplea en los conmutadores de red. La sección 4 presenta la configuración del nodo de acceso, y la 5 el comportamiento de los diversos algoritmos ante fuentes persistentes. En la sección 6 se estudia el retardo y la ocupación de las colas para diversos valores en los umbrales QT y DQT en el nodo de acceso. La sección 7 ofrece los resultados para una fuente no estrangulada en la red, para finalizar con las conclusiones y comentarios finales.

2. Mecanismo de realimentación de red

La existencia de recursos limitados en la red y los requisitos de transmisión de las fuentes orientan la mejora del servicio hacia la definición de un elemento de gestión responsable de maximizar la utilización y de asegurar cotas de calidad. Para servicios en los cuales el retardo es fundamental, las redes sin garantía de entrega son un entorno inhóspito. Para resolver este problema, un procedimiento que reporte la información de caudal disponible existente en la red va a permitir la transmisión bajo ciertos umbrales de calidad a todas aquellas fuentes que sean capaces de adaptarse a esta situación, mediante técnicas de regulación de tasa.

Considérese una conexión en la red. Para llevar a cabo el procedimiento de sondeo de su caudal disponible, se definen paquetes exploradores y de gestión, que serán generados en la fuente y atravesarán la red hasta alcanzar el destino, para regresar con información del estado de congestión y caudal desocupado. Cuando un conmutador o enrutador perteneciente al trayecto de la conexión intercepte uno de dichos paquetes, podrá modificar el contenido de sus campos de acuerdo a los recursos libres en ese dispositivo, si los valores en el dispositivo son más limitantes que los transportados.

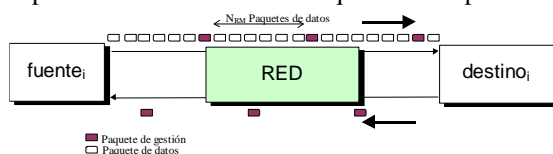


Fig. 2. Realimentación de caudal disponible procedente de la red.

Considerando que los paquetes exploradores cruzan la red de origen a destino y van adquiriendo información de todos y cada uno de los elementos por los cuales van pasando, cuando regresan a la

fuente que los generó contendrán información del caudal más restrictivo disponible. Entonces, la fuente deberá adaptar el tráfico que genere a la información recibida si desea obtener cotas de calidad (retardo y tasa de pérdidas). La fig.2 ilustra el mecanismo de paso de paquetes exploradores, generados de acuerdo a una relación de 1 cada N_{rm} paquetes de datos. En suma, este algoritmo permite revelar el caudal disponible para cada fuente desde el nodo de acceso, de forma independiente del resto de fuentes y con actualización periódica frecuente.

Aunque este procedimiento funciona de forma simple para paquetes de tamaño fijo (caso de las redes ATM), se puede generalizar sin dificultad para paquetes de tamaño variable sin más que considerar la generación de un mensaje explorador cada cierto volumen de datos.

En las redes de banda ancha basadas en tecnología ATM, el servicio ABR es un ejemplo de este mecanismo. Las conexiones ABR denominan celdas RM a los paquetes exploradores y se generan por defecto a $N_{rm} = 32$. Los conmutadores de la red son los responsables de programar los campos de la celda. En particular, los bits de Indicación de Congestión (CI), de No Incremento (NI) y el campo de tasa explícita (ER) pueden ser modificados [2], dependiendo del algoritmo empleado y de las decisiones que considere, en función del nivel de congestión de la red.

Desde la perspectiva de las fuentes en servicio ABR, éstas transmiten a una tasa comprendida entre un valor mínimo MCR y un pico PCR. El valor preciso depende de la congestión interna de la red y viene limitado por el campo ER que se recibiere en la celda RM, después de ser modulado por los conmutadores de acuerdo a los recursos existentes. Los bits CI y NI van a permitir incrementos de tasa (hasta el mínimo entre PCR y ER) o solicitarán un decremento (hasta no menos de MCR). Los detalles relativos al comportamiento de las fuentes y destinos en servicio ABR están especificados en [2].

3. Nodo de acceso con control DMRCA

El Algoritmo de Control de Tasa por Máximo Dinámico es un mecanismo de cálculo aproximado de tasa justa, al igual que lo son el EPRCA y el MMRCA, aunque aporta algunas mejoras y resuelve importantes inconvenientes de los primeros. DMRCA fue propuesto para efectuar un marcado inteligente de los mensajes de gestión en los conmutadores, con el fin de conseguir una repartición justa de recursos (fairness) aún en condiciones de conexiones estranguladas en algún punto de la red, es decir, en caso de que en algún nodo interno la conexión no consiga alcanzar por falta de recursos la tasa justa que le correspondería. En este algoritmo (al igual que en EPRCA y MMRCA), todas las conexiones comparten la misma cola de espera, en la cual se han definido dos

umbrales de decisión, QT y DQT, como se presenta en la figura 1.

Un elemento Gestor monitoriza la tasa máxima (MAX) de todas las conexiones que tributan en el nodo y almacena el número de conexión correspondiente a dicho MAX en MAX-VC, el cual es inicializado cada cierto tiempo. Con el fin de alisar las oscilaciones debidas a los cambios en el valor MAX, el dispositivo efectúa un promediado cuando aparece un nuevo valor máximo, de la siguiente manera:

$$AMAX = (1-\alpha) \cdot AMAX + \alpha \cdot MAX$$

donde α es el factor de promediado. Cuando la longitud de la cola compartida por todas las fuentes supera el umbral QT, el dispositivo realiza un marcado inteligente basado en un umbral de marcado, MT, de valor:

$$MT = AMAX \cdot F_n(\text{longitud de la cola})$$

donde $F_n(\text{longitud de la cola})$ es una función no creciente cuyo rango esta comprendido entre 0 y 1, cuya misión es la de endurecer las condiciones a medida que la cola acumula más unidades.

Cuando la longitud de la cola es inferior a QT, se autoriza el incremento de tasa para todas las fuentes, aunque sólo aquellas cuya conexión no esté estrangulada son las que realmente podrán aumentar el caudal generado, puesto que no se verán limitadas al valor ER. Cuando la longitud de la cola se encuentra entre QT y DQT, la tasa instantánea a la que la fuente está transmitiendo se compara con MT, y el resultado se empleará para selectivamente indicar congestión a aquellas que excedan este umbral. Cuando la longitud de la cola supera a DQT, se solicita reducción de velocidad de transmisión, situando el máximo admitido a $AMAX \cdot MRF$, donde MRF es el Factor de Reducción Mayor.

El sistema se comportará de forma equitativa siempre que AMAX tienda al valor de tasa justa según el criterio max-min, calculado de acuerdo a:

$$Tasa_justa = \frac{C_i - \sum \text{Tasas_conexiones_estranguladas}}{N_i - \sum \text{Conexiones_estranguladas}}$$

El siguiente pseudocódigo representa este algoritmo:

```
// longTotalCola: longitud de la cola
// MT: marking threshold
// i: número de conexión o fuente

// Control de MAX_Timeout
if (MAX > 0)
    MAX_Timeout_DMRCA = (long)(1.0 / MAX * Nrm * 2);
else
    MAX_Timeout_DMRCA = 0;

if (CCR_recibido >= beta * MAX)
    AMAX = (1-MT_DMRCA)*AMAX + MT_DMRCA * MAX;

if (longTotalCola >= DQT) { // ** Congestión severa **
    for (int i=0; i<n_fuentes; i++) {
        CI[i] = true;
        ER[i] = min(MRF * AMAX, ER_celda);
    }
}
```

```
else
    if (longTotalCola >= QT) { // ** Congestión moderada **
        MT = AMAX * Fn(longTotalCola);
        for (int i=0; i<n_fuentes; i++)
            if (CCR[i] >= MT) {
                CI[i] = true;
                ER[i] = min(MT, ER_celda);
            }
            else CI[i] = CI_celda;
    }
    else { // ** No Congestión **
        for (int i=0; i<n_fuentes; i++)
            CI[i] = CI_celda;
```

4. Configuración

La figura fig.3 muestra el diagrama funcional del nodo de acceso, cuyo diseño ha sido implementado para la simulación. Un generador de paquetes, siguiendo el comportamiento de fuente ABR, está sometido a la disciplina del nodo de entrada a la red, que le garantizará unas determinadas cotas de retardo y tasa de pérdida. Por lo tanto, las fuentes deberán adaptar su tasa al perfil indicado por el elemento Gestor, que es el responsable de administrar los recursos y permitir el paso de los flujos de datos. Para ello, el Gestor efectúa dos funciones: Por un lado, el scheduler o planificador, responsable de la disciplina de selección del orden con que las celdas procedentes de las diversas tributarias han de ser transmitidas a la red. Por otro, el Control de Tasa, que administra y modifica los parámetros de las celdas de gestión para comunicar a las fuentes el caudal asignado.

La disciplina de planificación empleada afecta a los parámetros de calidad de servicio obtenidos localmente. La técnica de FIFO garantiza el menor retardo, igual para cualquier fuente sea cual fuera su tasa, así como un tamaño de cola parcial dentro de la cola compartida proporcional al caudal generado por dicha fuente, tal como se observa en las simulaciones realizadas.

En las redes ATM tradicionales, las celdas RM se reciben en la fuente directamente de la red sin atravesar ningún elemento Gestor, con lo cual es la red quien controla directamente el caudal que pueden generar las fuentes. Ello supone una fuerte restricción para los generadores de tráfico en caso de requerir un pico de transmisión, por la falta de elementos en las redes ATM que permitan soportar estos excesos (si no han sido previamente contratados en los parámetros de la conexión), en concreto realizando tareas de alisado de tráfico (recorte de picos mediante introducción de ligeros retardos). Aunque con posterioridad el tráfico a transmitir sea bajo, la existencia de un pico conlleva la pérdida de datos. Por ello, el esquema propuesto incluye el espaciado de tráfico para evitar la pérdida de paquetes.

El elemento Control de Tasa gestiona las celdas RM con opción a realizarlo de acuerdo a tres criterios internos:

1) Inhibido. Bajo este funcionamiento, el estado en el que se encuentra la red (proporcionado por sus conmutadores por medio de las celdas de gestión) pasa de forma transparente hacia las fuentes y son éstas las que en último término deciden la velocidad a la cual transmitir.

2) A tiempo de celda de gestión. En este caso, el Gestor puede modificar los campos de los paquetes de gestión procedentes de la red, teniendo en cuenta criterios locales de congestión en el nodo de acceso a la red. En este caso considera el riesgo de desbordamiento en la cola de espera del Gestor por una generación excesiva de datos en las fuentes y puede solicitar una disminución de caudal emitido, a pesar incluso de que la red autorice incrementos. Con ello, se consigue ejercer el control sobre el retardo y la tasa de pérdidas al tiempo que se evita agravar la situación de congestión en el nodo de acceso, consiguiendo un perfil de generación más próximo a los recursos en red al tiempo que se garantiza una calidad de servicio.

3) A múltiplos de celda de gestión. En el caso en que se opere con fuentes extremadamente sensibles a variaciones de retardo y pérdida de paquetes, es necesario ejercer una gran autoridad sobre ellas de forma que generen con precisión la cantidad de datos transmisibles. Esta situación requiere que las fuentes dispongan de una elevada capacidad de adaptación para cumplir los dictámenes del flujo de celdas RM suplementarias.

El modelo de red considerado contempla tanto estrechamiento del ancho de banda para determinadas conexiones (estrangulamiento) como retardo por transmisión y almacenamiento en nodos intermedios. Con este propósito, el elemento red ha sido modelado como una cola FIFO, que añade una espera igual al retardo ocasionado por la propagación hacia el destino. Asimismo, fija apropiadamente los valores de los campos de las celdas de gestión para que éstas indiquen estrangulamiento de conexión, si fuera el caso. De esta forma, es posible incorporar escenarios tradicionales de test tales como el GFC1 o parking lot por simple ajuste de los campos de las celdas RM.

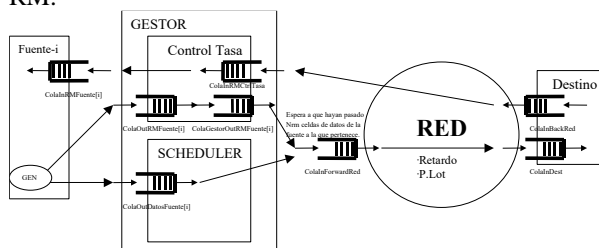


Fig. 3. Modelo de simulación.

Finalmente, el elemento Destino actúa como sumidero de celdas de datos y retorna las de gestión.

Con el fin de evaluar las prestaciones del nodo de acceso, las simulaciones llevadas a cabo se han efectuado bajo las siguientes condiciones:

- 5 fuentes con parámetros $icr=mcr=0$, $pcr=1$, $nrm=32$, $rdf=1/32$, $rif=1/128$. Considerando que 1 es el máximo caudal del enlace, la tasa a la cual puede transmitir una fuente no está limitada más que al propio caudal disponible del enlace y al estado de congestión del nodo de acceso.

- Las fuentes 1 2 y 3 se definen con conexiones estranguladas en la red por lo que no disponen de la tasa justa que otorgaría el criterio de max-min. Los valores considerados en la simulación para ellas son $\{0,05; 0,05; 0,1\}$. En consecuencia, el ancho de banda justo para las fuentes 4 y 5 bajo max-min [1], [2] resulta ser, de acuerdo a la expresión de la tasa justa:

$$Tasa_justa(\#4,\#5)=[1-(0,05+0,05+0,1)]/(5-3)=0.4$$

- La red no introduce retardo por propagación, por lo que únicamente existe la demora por retorno de celdas de gestión en destino, que es de 1 tiempo de celda. Las consecuencias a este efecto requieren un estudio más minucioso.

- Disciplina de planificación de servicio: FIFO

Debe hacerse notar que para el tráfico generado por la fuente se considera tanto el propio de datos como el de celdas RM, y así se refleja en el ACR de las celdas de gestión.

5. Comportamiento del algoritmo

En esta sección se va a estudiar el comportamiento del sistema bajo fuentes persistentes en un entorno ATM bajo servicio ABR con criterio max-min para la asignación de recursos, comparando EPRCA y MMRCA frente a DMRCA, para umbrales $QT = 50$ y $DQT = 200$. En la próxima sección se modifican estos umbrales para evaluación del efecto producido sobre el transitorio, el retardo, el tamaño de las colas y la probabilidad de pérdida.

La duración de las simulaciones es de 10^8 celdas en el enlace, de donde se obtienen funciones densidad de retardo y de ocupación de las cuales extraer cualquier parámetro estadístico.

En las siguientes gráficas, el término colas parciales hace referencia a la porción de cola compartida que emplea cada una de las fuentes. La longitud de la cola compartida se obtiene consecuentemente del agregado de colas parciales.

Se presentan a continuación las características de los transitorios iniciales (en los cuales interesan mínimas oscilaciones y tendencia a utilización completa del ancho del enlace), la evolución de la longitud de las colas, las densidades de probabilidad de ocupación de las colas parciales y de los retardos para cada una de las fuentes, y la probabilidad de pérdida en cada cola parcial asociada a una fuente.

5.1. EPRCA

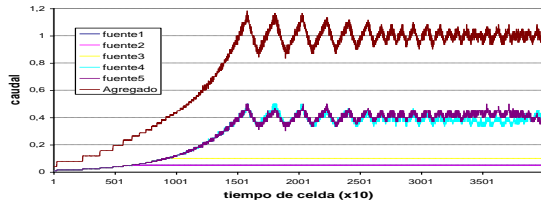


Fig. 4. Evolución del caudal total y de cada fuente

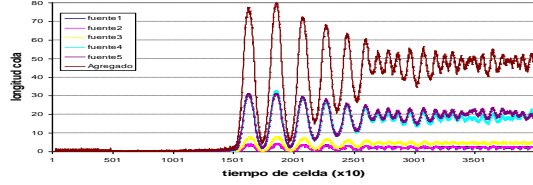


Fig. 5. Evolución de la cola (total y parciales)

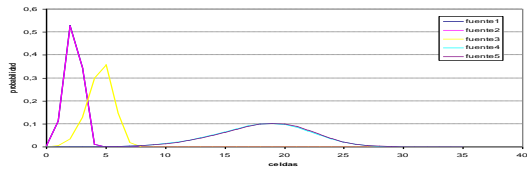


Fig. 6. Densidad probabilidad Ocupación Colas Parciales

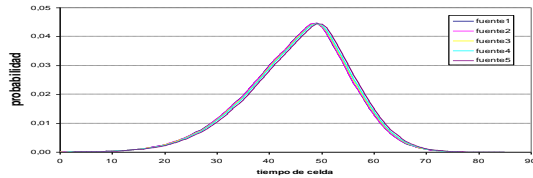


Fig. 7. Retardo acumulado colas parciales

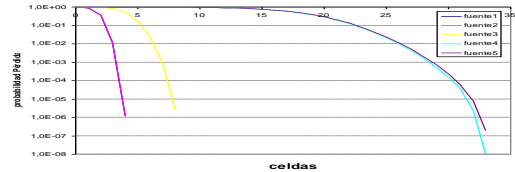


Fig. 8. Probabilidad de pérdida para cola parcial limitada.

5.2. MMRCA

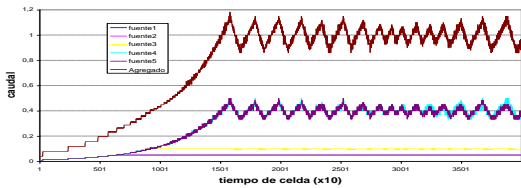


Fig. 9. Evolución del caudal total y de cada fuente

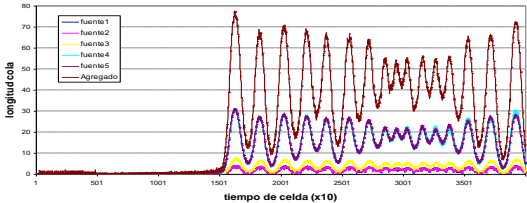


Fig. 10. Evolución de la cola (total y parciales)

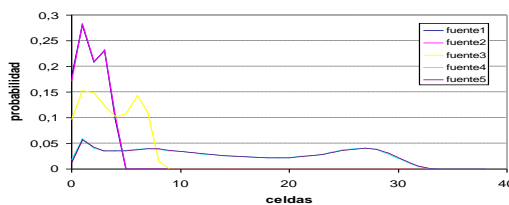


Fig. 11. Densidad probabilidad Ocupación Colas Parciales

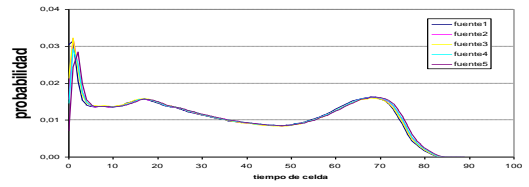


Fig. 12. Retardo acumulado colas parciales

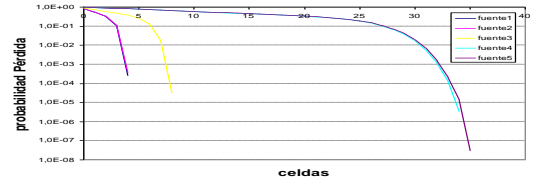


Fig. 13. Probabilidad de pérdida para cola parcial limitada.

5.3. DMRCA

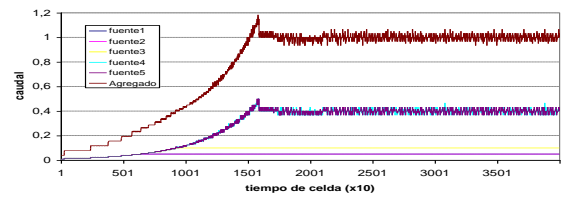


Fig. 14. Evolución del caudal total y de cada fuente

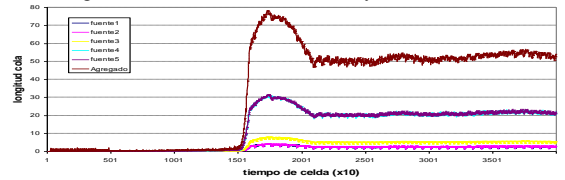


Fig. 15. Evolución de la cola (total y parciales)

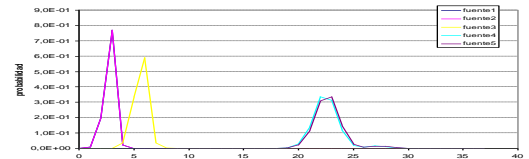


Fig. 16. Densidad probabilidad Ocupación Colas Parciales

	media	desv. estándar	máximo
fuelle1	2,8030	0,4645	5
fuelle2	2,8044	0,4639	5
fuelle3	5,6348	0,6303	9
fuelle4	22,5983	1,4533	35
fuelle5	22,7532	1,4599	36

Tabla 1. Estadísticas de la ocupación parcial de cola.

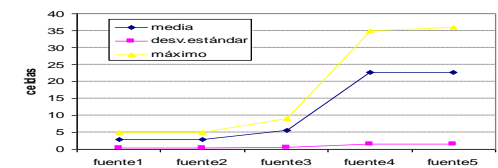


Fig. 17. Estadísticas de la ocupación parcial de cola

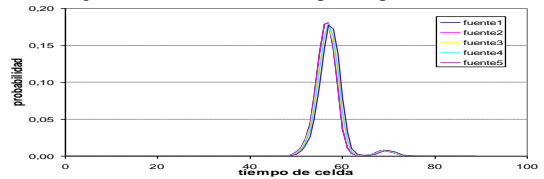


Fig. 18. Retardo acumulado colas parciales

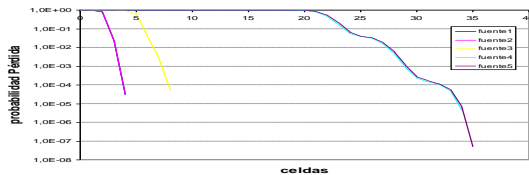


Fig. 19 Probabilidad de pérdida para cola parcial limitada.

	media	desv.estándar	máximo
fuelle1	56,8144	3,3327	87
fuelle2	56,8430	3,3287	87
fuelle3	57,1119	3,3684	88
fuelle4	57,2676	3,3250	88
fuelle5	57,6671	3,3254	88

Tabla 2. Estadísticas del retardo en cola.

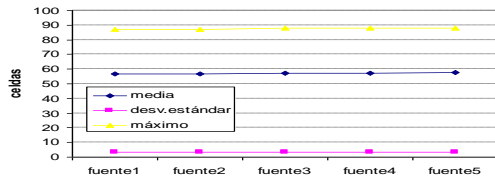


Fig. 20. Estadísticas del retardo en cola

5.4. DMRCA en el nodo de acceso

La simple inspección de las trazas de tasa permite descubrir que los niveles de oscilación son sensiblemente menores en el algoritmo de DMRCA que en EPRCA o MMRCA, debido al comportamiento de la evolución del tamaño de la cola. La oscilación producida viene motivada por el tipo de criterio definido en cada algoritmo para realizar la reducción de tasa: Mientras EPRCA y MMRCA consideran los menores niveles de caudal transmitido como referencia [3], [5], DMRCA toma el valor alisado de los máximos, lo cual ante conexiones estranguladas, asegura que las fuentes a las que se les ha asignado su tasa justa no se vean perjudicadas por existir una referencia tomada de un valor de conexión estrangulada. El mejor comportamiento se pone de manifiesto tanto por la forma del transitorio (trazas de tasa y tamaño de cola) como por el retardo y tamaños de cola durante el régimen permanente. Especialmente, el retardo mejora en cuanto a la desviación que presenta, lo cual es fundamental en la transmisión de tráfico multimedia. En parámetros de ATM, se trata de una desviación de retardo de celda (CDV) pequeña (del orden del 5,8% de la media).

El tráfico total generado por las fuentes puede llegar a superar instantáneamente el 100% del enlace con el objetivo de alcanzar su utilización máxima. Ello es posible gracias a la existencia de la cola compartida del nodo de acceso.

En cuanto al fenómeno de la oscilación, cuando la cola compartida se encuentra vacía (por debajo de QT) se autorizan aumentos de tasa. Si el algoritmo no es suficientemente robusto, puede no detener ese incremento a tiempo y llenar las colas, de forma que en ese momento se solicitará una disminución inmediata de tasa, lo cual favorece la existencia de oscilación. Este efecto puede observarse en EPRCA y MMRCA, pero queda resuelto en DMRCA.

En suma, DMRCA es apropiado como base para el mecanismo de control del nodo de acceso.

6. DMRCA: Efecto sobre retardo y ocupación de QT y DQT

La variación de los umbrales QT y DQT afecta al tamaño de las colas y al retardo, así como a las variaciones de tasa que se exigen sobre la fuente (incidencia sobre los campos de los paquetes de gestión). Esto es, ante umbrales más grandes el alisado sobre la fuente es superior por el mayor margen de almacenamiento existente.

Esta sección cuantifica este efecto modificando los valores de estos parámetros. QT variará su valor para DQT=200 y DQT=50.

6.1. Efecto para DQT = 200, QT = 0

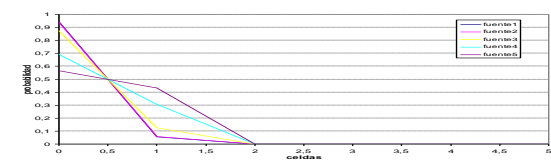


Fig. 21. Densidad probabilidad Ocupación Colas Parciales

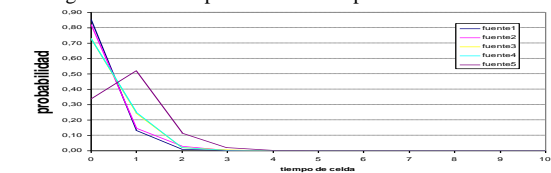


Fig. 22. Retardo acumulado colas parciales

6.2. Efecto para DQT = 200, QT = 10

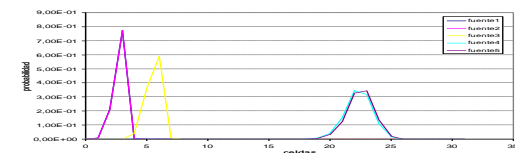


Fig. 23. Densidad probabilidad Ocupación Colas Parciales

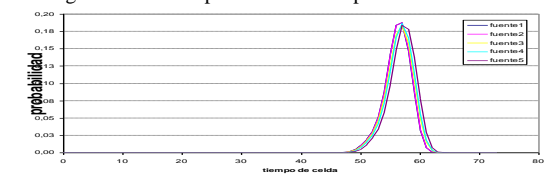


Fig. 24. Retardo acumulado colas parciales

6.3. Efecto para DQT = 200, QT = 150

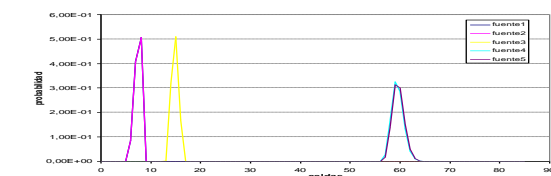


Fig. 25. Densidad probabilidad Ocupación Colas Parciales

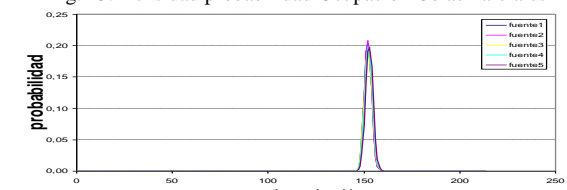


Fig. 26. Retardo acumulado colas parciales

6.4. Efecto para DQT = 200, QT = 175

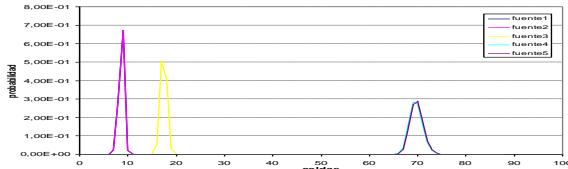


Fig. 27. Densidad probabilidad Ocupación Colas Parciales

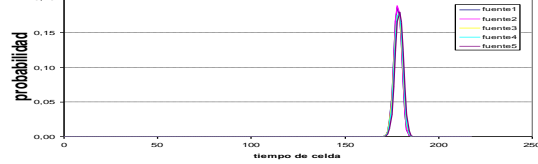


Fig. 28. Retardo acumulado colas parciales

6.5. Efecto para DQT = 200, QT = 200

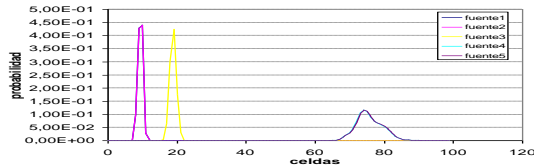


Fig. 29. Densidad probabilidad Ocupación Colas Parciales

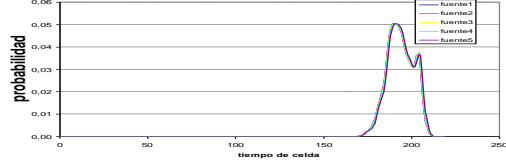


Fig. 30. Retardo acumulado colas parciales

6.6. Efecto para DQT = 50, QT = 0

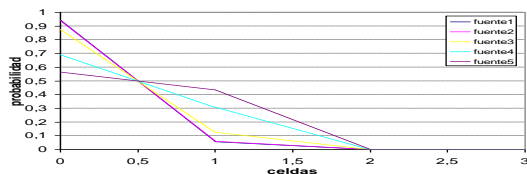


Fig. 31. Densidad probabilidad Ocupación Colas Parciales

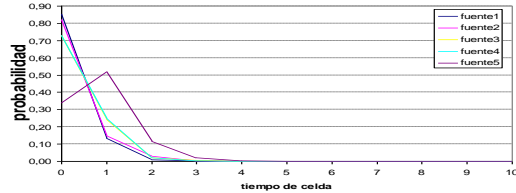


Fig. 32. Retardo acumulado colas parciales

6.7. Efecto para DQT = 50, QT = 10

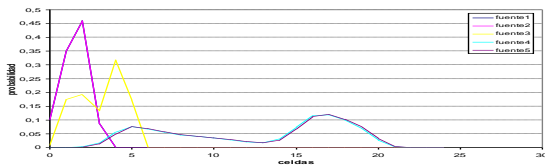


Fig. 33. Densidad probabilidad Ocupación Colas Parciales

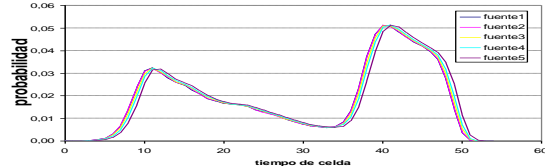


Fig. 34. Retardo acumulado colas parciales

6.8. Efecto para DQT = 50, QT = 25

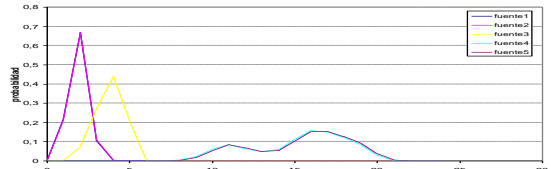


Fig. 35. Densidad probabilidad Ocupación Colas Parciales

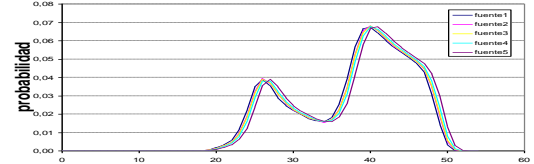


Fig. 36. Retardo acumulado colas parciales

7. Retardo y ocupación de colas

Se presentan a continuación los resultados obtenidos para una de las fuentes con conexión no estrangulada (fuente 4 ó 5 son equivalentes) que por no estarlo son las que entregan mayor caudal y por tanto se encuentran en la situación más desfavorable en cuanto a tamaño de cola parcial y para las que el algoritmo de DMRCA actúa más enérgicamente. Nótese que una conexión con caudal estrangulado no puede ser causa de congestión, puesto que el ancho de banda que consume puede crecer únicamente hasta el valor de estrangulamiento.

Las tablas 3 y 4 resumen el comportamiento de DMRCA bajo fuentes persistentes, para diferentes valores de los umbrales QT y DQT.

QT	0	150	175	200
Medio	0,8287	152,7710	178,9446	194,2146
Desv.Std.	0,7363	2,2330	2,4695	7,6288
Máximo	9	213	217	218

Tabla 3. Retardo con DQT=200

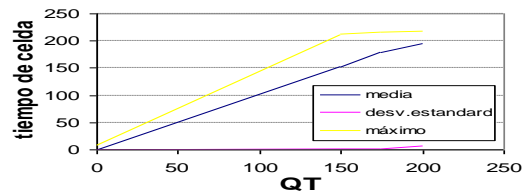


Fig. 37. Retardo con DQT=200

QT	0	150	175	200
Medio	0,4357	59,6357	69,7901	75,7758
Desv.Std.	0,4974	1,4257	1,6348	4,0280
Máximo	2	85	87	98

Tabla 4. Ocupación de cola parcial con DQT=200

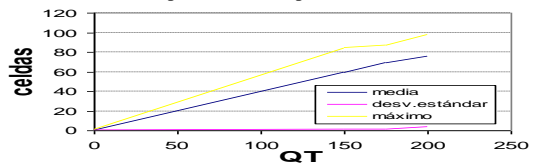


Fig. 38. Ocupación de cola parcial con DQT=200

QT	0	10	25
Medio	0,8287	31,5075	38,5602
Desv. Std.	0,7363	13,6045	7,3921
Máximo	9	54	54

Tabla 5. Retardo con DQT=50

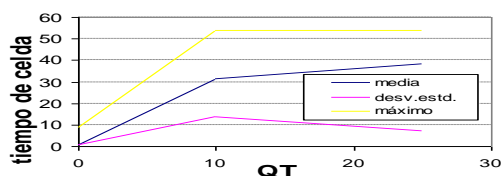


Fig. 39. Retardo con DQT=50

QT	0	10	25
Medio	0,4357	12,6084	15,3431
Desv.Std.	0,4974	5,3216	2,9571
Máximo	2	23	23

Tabla 6. Ocupación de cola parcial con DQT=50

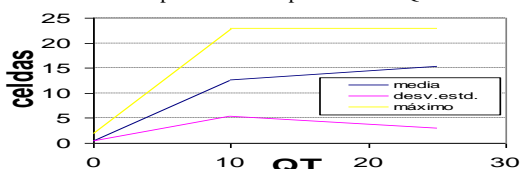


Fig. 40. Ocupación de cola parcial con DQT=50

Para las tablas con valores de DQT mayores a N_{rm} , se concluye que el responsable del retardo medio generado sobre los paquetes es QT mientras que DQT lo es del máximo. La misma influencia para la ocupación media y máxima de las colas. Asimismo, el empleo de colas de pequeño tamaño (QT y DQT pequeños) pone de relieve la importancia de la cadencia (valor N_{rm}) de generación de las celdas de gestión, que son las que controlan la velocidad con que la fuente se adapta. Como se observa, cuando la cola presenta QT y DQT mayores a N_{rm} (Figs. 37 y 38), el comportamiento del retardo y ocupación es proporcional (lineal) a QT y limitados superiormente por DQT. Sin embargo, si el valor de N_{rm} es comparable a DQT (figs. 39 y 40), el sistema alcanza rápidamente una cota superior, debido al gran control que se ejerce sobre la fuente.

8. Conclusiones

En este trabajo se ha efectuado una evaluación del impacto sobre el retardo y el tamaño requerido de cola de un nodo específico de acceso a la red basado en el mecanismo DMRCA adaptado para realizar funciones de control de tasa bajo restricciones de calidad de servicio. Se ha demostrado su mejor funcionamiento en este punto de la red frente a EPRCA y MMRCA, en cuanto a oscilaciones, niveles de retardo, tamaño de cola necesaria y probabilidad de pérdida de paquetes. Estos parámetros han sido cuantificados para los umbrales QT y DQT, que controlan el compromiso entre calidad del alisado y capacidad de adaptación exigida a las fuentes y ofrecen un criterio para el dimensionado del nodo de acceso basándose en las gráficas presentadas.

Dentro de las líneas continuistas cabe reseñar el estudio del impacto sobre el control de tasa en el nodo de acceso en función del retardo en la entrega de información de caudal disponible, el efecto de apertura y cierre de conexiones durante el funcionamiento del algoritmo y el análisis del

comportamiento ante otros tipos de fuentes tales como On-Off y generadores multimedia reales como por ejemplo vídeo MPEG con calidad variable.

Referencias

- [1] A.Arulambalam, X.Chen. Allocating Fair Rates for Available Bit Rate Service in ATM networks. IEEE Communications. Vol.34, No.11, pp.92-100. Noviembre 1996.
- [2] ATM Forum. Traffic Management Specification. Version 4.0. Abril 1996.
- [3] F.M.Chiussi, Y.Xia, V.P.Kumar. Dynamic Max Rate Control Algorithm for Available Bit Rate Service in ATM networks. GLOBECOM'96, pp.2108-2117. Londres. 1996.
- [4] R.Jain, S.Kalyanaraman, S.Fahmy, R.Goyal. Source Behavior for ATM ABR Traffic Management: An explanation. IEEE Communications. Vol.34, No.11, pp.50-57. Noviembre 1996.
- [5] S.Muddu, F.M.Chiussi, C.Tryfonas, V.P.Kumar. Max-Min Rate Control Algorithm for Available Bit Rate Service in ATM networks. ICC'96, pp.412-418. Dallas. 1996.
- [6] L.Zhang, S.Deering, D.Estrin, S.Shenker, D.Zappala. RSVP: A new Resource ReSerVation Protocol. IEEE Network, Vol.7, No.5, pp.8-18, Septiembre 1993.
- [7] B.Zheng, M.Atiquzzaman.. Traffic Management of Multimedia over ATM Networks. IEEE Communications. Vol.37, No.1, pp.33-38. Enero 1999.

Capítulo 4B: Aplicaciones de Teleeducación, Telemedicina y Diseño de Interfaces

Teleconferencia y Trabajo Cooperativo en Entornos Universitarios.

E. Pastor, J. Carracedo, D. Fernandez, J. A. Sanchez (U. Politécnica de Madrid)

Experiencias en el uso del lenguaje Java para la enseñanza de Ingeniería Telemática.

J.C. Dueñas, G. Huecas, S. Pavón, T. Robles, J. Salvachúa (U. Politécnica de Madrid)

Acceso a bases de datos de imágenes de radiología mediante ActiveX scripting.

J. Alvarez, C. del Barrio, M. de Cabo, E. Fernandez, G. Garcia y M. Rodriguez. (U. Valladolid)

Sistema de videoconferencia aplicado en telepatología.

N. Pérez, M. de Cabo, Y.A. Dimitriadis, E. Fernández. (U. Valladolid)

Generación de programas gráficos en X Window. Herramienta xgui.

C.M. Ramírez-Casañas, J.R. Ansó Magaña (U. Las Palmas)

Interfaz de Usuario basado en Mundos VRML para un Museo Virtual Tridimensional en Internet.

J.F. Díez Higuera, F.J. Díaz Pernas (U. Valladolid)

Teleconferencia y trabajo cooperativo en entornos universitarios

Encarna Pastor¹, Justo Carracedo², David Fernández¹, José A. Sánchez²

¹Dpto. Ingeniería de Sistemas Telemáticos, Universidad Politécnica de Madrid
ETSI Telecomunicación, Ciudad Universitaria, 28040 MADRID

Telf: 91-5495700, Fax: 91-5432077

E-mail: {encarna,david}@dit.upm.es

²Dpto. Ingeniería y Arquitecturas Telemáticas, Universidad Politécnica de Madrid
EUIT Telecomunicación, Campus Sur, 28031 MADRID

Telf: 91-3367820, Fax: 91-3319229

E-mail: {jcarracedo,jsanchez}@diatel.upm.es

Abstract

Teleconference is one of the support technologies for tele-education activities that is becoming popular, mainly in university environments. However, when deploying tele-education services based on teleconference, many issues are still unexplored and some problems remain unsolved. At many times, the existing products in the market show lack of robustness, they are only fitted to a specific networking technology or even they do not satisfy the special needs of applications. On the other hand, the use of teleconference systems requires people able to adapt their behaviour to the new media features. This paper describes the problems found and the solutions adopted in the deployment of teleconference systems over different networking technologies as result report of a r&d project carried out by two departments of the Universidad Politécnica de Madrid.

1. Introducción

Muchas universidades y centros educativos están invirtiendo una gran cantidad de recursos en equipamiento que facilite actividades de *teleeducación*. El número, variedad y escala de experiencias, ofertas y servicios que se reflejan en publicaciones recientes o que se encuentran en Internet, muestra que hay una tremenda actividad tanto por parte de grupos de investigación como de instituciones y empresas que van adquiriendo compromisos en la línea de uso de soportes telemáticos en educación.

Así, estamos siendo testigos de la proliferación de universidades virtuales, campus virtuales, aulas y cursos virtuales que, en definitiva, con tecnologías y modelos muy distintos, proponen actividades educativas que podrán ser seguidas por profesores y estudiantes sin que sea imprescindible su presencia física en el lugar en que se impartan o desarrollen.

Son muchos los factores que influyen en el auge de la teleeducación. Por ejemplo, cabe destacar:

- La necesidad de formación continua de profesionales, que requieren una formación flexible, en horario cómodo, bien desde su domicilio o su lugar de trabajo.
- Los estudiantes a tiempo parcial, cuyas vidas profesionales están caracterizadas por una carga fuerte de trabajo y constantes viajes.
- La masificación de las universidades presenciales, que puede aliviarse ofreciendo cursos a distancia, sin necesidad de acudir al campus.
- Las tendencias pedagógicas, que apuntan hacia un modelo de aprendizaje centrado más en el trabajo personal del alumno y menos en la mera

transmisión de conocimientos por parte del profesor.

A todo ello habría que añadir la gran ventaja que podría representar el que estudiantes de una determinada universidad asistan de forma remota a ciertas clases que se impartan en otros centros.

Una de las tecnologías de apoyo a la teleeducación que se va extendiendo con bastante rapidez, y sobre todo en el entorno universitario, es la *teleconferencia*.

Teleconferencia es el nombre genérico que se da al hecho de utilizar los medios de telecomunicación para llevar a cabo encuentros y conversaciones entre individuos que están físicamente alejados. En este sentido se podría decir que la teleconferencia no es un concepto nuevo: empieza con el teléfono. La teleconferencia puede realizarse a través de distintos medios y utilizando distintas tecnologías de comunicación y así el término genérico se suele especializar en: audioconferencia, videoconferencia (audio-vídeo) y conferencia de datos (espacios de trabajo compartido).

A pesar de que los sistemas de teleconferencia son ya bastantes populares, todavía hay muchos aspectos por explorar y problemas por resolver cuando se trata de implantar un servicio. Hay muchos productos en el mercado, sin embargo a veces no están suficientemente depurados, o están adaptados a una tecnología de red concreta o no responden a las necesidades especiales de una aplicación y hay que recurrir a integraciones y desarrollos específicos. Por otra parte, estos desarrollos específicos permiten a la vez investigar y experimentar sobre nuevos servicios interactivos y nuevas plataformas.

Este artículo describe los problemas encontrados y soluciones adoptadas en la implantación de sistemas de teleconferencia sobre distintas infraestructuras de comunicación, como resultado del proyecto T+T en el que han participado dos departamentos de la Universidad Politécnica de Madrid.

Se describe, en primer lugar, el contexto del proyecto y sus objetivos. A continuación se detalla la infraestructura de red y el equipamiento elegido. El artículo se centra después en la descripción del modelo de aula virtual y los perfiles o escenarios de comunicación utilizados en las actividades educativas y académicas realizadas. Finalmente, el proyecto contribuyó a la dotación de unas aulas para actividades de teleeducación y trabajo en grupo y este artículo también dibujará algunas ideas sobre el diseño de este tipo de aulas.

2. El marco de trabajo

El proyecto T+T ha sido una experiencia piloto que ha utilizado la infraestructura de comunicaciones ATM de la Universidad Politécnica de Madrid para la organización de actividades conjuntas entre dos centros de la Universidad. Ha sido financiado por la CICYT (Programa Nacional de Aplicaciones y Servicios Telemáticos, 1997-1998).

En concreto, la experiencia ha involucrado a dos Departamentos de la UPM ubicados en campus distintos: el Departamento de Ingeniería de Sistemas Telemáticos de la ETSI de Telecomunicación y el Departamento de Ingeniería y Arquitecturas Telemáticas de la EUIT de Telecomunicación.

El objetivo del proyecto ha sido llevar a cabo experimentos con nuevos servicios que faciliten actividades en grupo o reuniones sin tener que desplazarse del lugar de trabajo, como por ejemplo: organización de teleseminarios sobre temas de interés, coordinación y trabajo en tele-reuniones, participación en charlas o conferencias impartidas en el centro remoto, teleclase en cursos de doctorado o postgrado, etc.

Se trataba también de facilitar la colaboración entre dos departamentos que han seguido una larga trayectoria de colaboración tanto en aspectos docentes como de I+D. El resultado de esta experiencia piloto ha potenciado esta colaboración, posibilitando la realización de actividades que hoy en día son menos factibles debido a la separación geográfica. Estos resultados creemos que pueden ser de utilidad para toda la comunidad universitaria y en particular para el área de Ingeniería Telemática.

Las actividades que se han realizado durante el curso 97/98 han incluido:

- Seminarios de cursos de doctorado del Programa de Doctorado del DIT. En estos cursos hay alumnos de la EUITT matriculados

que han podido asistir y participar en los seminarios de forma remota.

- Asistencia (remota) a lecturas de Proyectos Fin de Carrera.
- Reuniones técnicas y de coordinación del propio proyecto. Trabajo en grupo.
- Sesiones de demostración para alumnos de diversos cursos de ambas Escuelas.

Estas actividades han continuado después de terminar el proyecto, y los teleseminarios y el trabajo cooperativo entre profesores son actividades ya integradas en el día a día académico.

Durante las primeras fases del proyecto se abordó el estudio y definición de las soluciones técnicas a implantar. Se contaba en cada Escuela con un aula que el proyecto contribuyó a dotar, y que en la actualidad se utilizan como salas de videoconferencia y aulas de teleeducación.

3. Infraestructura de red y equipamiento

Se buscaba una solución flexible que permitiese, no sólo el aprovechamiento de la infraestructura proporcionada por la red ATM de la Universidad Politécnica, sino la creación de unas salas de videoconferencia polivalentes que pudieran ser utilizadas para conectarse con otros centros, mediante RDSI o directamente a través de Internet, y susceptibles de ser utilizadas en escenarios sencillos, como una videoconferencia personal, o en otros ciertamente más complejos, como puede ser un teleseminario impartido por varios profesores a varias salas remotas.

En todo momento se trató de buscar soluciones basadas en estándares internacionales ampliamente aceptados. Con este objetivo, se analizaron las distintas normas existentes en aquel momento, así como los productos disponibles en el mercado.

3.1. Estándares de videoconferencia

Cuando se inició el proyecto T+T en 1997, el único estándar ampliamente difundido, aceptado y con productos disponibles en el mercado lo constituía la norma H.320. Esta norma define un conjunto de protocolos a utilizar para la realización de videoconferencia sobre RDSI de banda estrecha (redes de conmutación de circuitos).

En esas fechas se estaba terminando de definir el conjunto de normas H.323 [1], que normalizan la realización de videoconferencia sobre redes de conmutación de paquetes, cuya aplicación principal es la videoconferencia sobre redes IP. Existían algunas implementaciones preliminares de los protocolos, pero ningún producto estable en el mercado.

Adicionalmente, se estaban definiendo dos normas para la realización de videoconferencia sobre redes ATM. La primera de ellas, H.321, es una mera

adaptación de la norma H.320 para su funcionamiento sobre redes ATM mediante emulación de circuitos; y la segunda, H.310, es la que realmente define unos procedimientos nuevos adaptados a las características y posibilidades de las redes ATM [2], [3]. Al igual que sucedía con H.323, no existían en el mercado productos estables acordes con estas normas.

En cuanto a la calidad de vídeo proporcionada por los estándares anteriores, hay que reseñar que sobre el papel todos permiten alcanzar la calidad de vídeo requerida en el proyecto T+T.

3.2. Infraestructura de red

La implantación de la infraestructura ATM de la UPM se llevó a cabo a lo largo de 1.997. Como muestra la Fig. 1, la red está basada en *routers* Cisco 4700 y 7505, conmutadores de acceso Cisco Catalyst 5000 y 3000, y conmutadores ATM Cisco LS 1010.

En el rectorado de la UPM existe un conmutador central LS 1010, al que se conectan mediante enlaces de fibra óptica a 155 Mbps equipos similares, situados en cada uno de los centros de la UPM (en el caso de que existan dos ó más centros geográficamente próximos, como en el Campus Sur de la UPM, donde se localiza la EUITT, se ha ubicado un único conmutador ATM). A cada conmutador ATM se conecta un equipo de acceso Catalyst 3000, que dispone de puertos Ethernet a 10 Mbps, lo que permite la conexión de las redes de área local existentes previamente en los Centros.

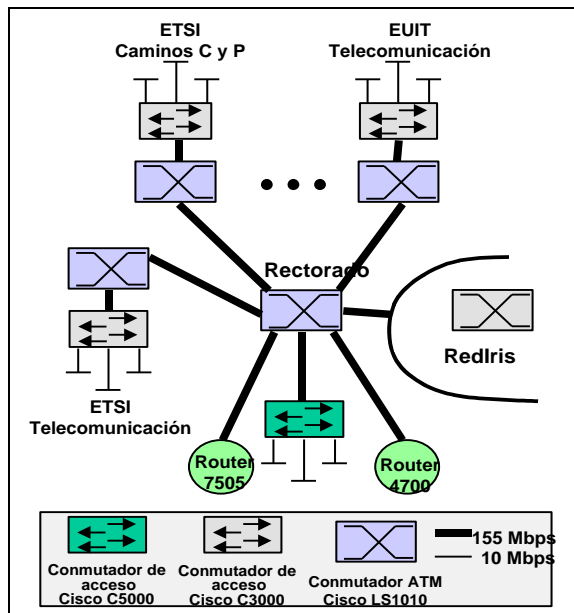


Fig. 1. Red ATM de la UPM

Desde el punto de vista lógico, se han creado una serie de redes virtuales que dan soporte a los distintos servicios y unidades organizativas de la UPM: gestión académica, bibliotecas, gestión económica, dirección, departamentos y alumnos. Estas redes están aún en fase de implantación.

Durante las primeras fases del proyecto T+T se realizaron experimentos de videoconferencia utilizando los accesos Ethernet a la red de la UPM. Pero, aunque se utilizaron tasas de vídeo bajas con el objeto de no interferir con el tráfico de producción, se comprobó que la calidad era muy variable en el tiempo, lo cual provocaba cortes frecuentes en el audio y, en definitiva, la imposibilidad de su uso práctico.

La última fase del proyecto se realizó mediante un acceso directo a la red ATM. Para ello, se solicitó el establecimiento de una conexión física desde las redes ATM locales de cada centro a los conmutadores de la UPM y el establecimiento de un circuito ATM entre ellas (PVC) de 5 Mbps.

Tal como aparece representado en la Fig. 2, los equipos de videoconferencia se conectaron mediante placas ATM a 25 Mbps a los conmutadores de grupo y se configuraron para utilizar IP sobre ATM ("Classical IP over ATM"). Se utilizó, además, un router CISCO 7000 para facilitar el acceso al resto de redes de los centros y a Internet.

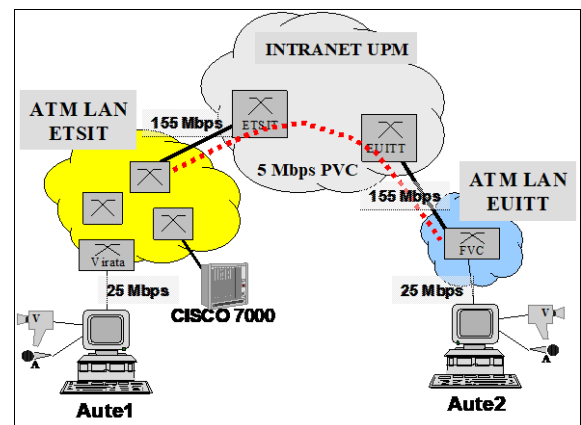


Fig. 2. Escenario de Red ATM

3.3. Solución Adoptada

La ausencia de equipos capaces de cubrir las necesidades de todos los escenarios planteados (ATM, RDSI e Internet) y un cierto retraso en el despliegue de la red ATM de la Universidad, nos llevó finalmente a una solución consistente en la utilización de dos equipos distintos (ver Fig. 3):

- Un equipo de videoconferencia de sala compatible con el estándar H.320.
- Un equipo de videoconferencia sobre ATM basado en un ordenador personal y aplicaciones de videoconferencia propietarias.

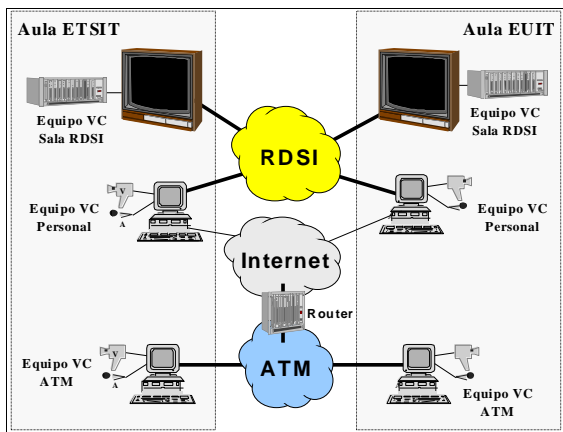


Fig. 3. Equipos de videoconferencia

El primero de los equipos, un Venue 2000 de la casa PictureTel, se conecta a un acceso básico RDSI y permite realizar videoconferencias a 128 kbps. Las principales razones de la elección de un equipo de este tipo fueron la buena calidad de audio y vídeo y, sobre todo, sus posibilidades de expansión. Aunque en el momento de su compra no estaban disponibles, estaban anunciadas futuras soluciones para su conexión a redes ATM, siguiendo el estándar H.321.

El segundo equipo de videoconferencia se basó en una plataforma PC dotada de las placas auxiliares necesarias. En particular:

- Placa ATM a 25 Mbps (FORE) para su conexión al conmutador local de la red de cada centro.
- Placa codificadora/decodificadora de vídeo de altas prestaciones (BITFIELD), capaz de enviar vídeo en tasas cercanas a los 2 Mbps. Esta placa incluye también el hardware necesario para la gestión del audio.
- Placa auxiliar de "overlay" de vídeo para superponer la imagen de vídeo en la pantalla del ordenador.

Las aplicaciones de videoconferencia utilizadas en este equipo, fueron originalmente desarrolladas en un proyecto anterior [4], [5] y posteriormente refinadas y mejoradas en el proyecto LEVERAGE del programa ACTS de la CE [6].

Dichas aplicaciones permiten el envío y recepción de audio y vídeo sobre IP en escenarios multipunto, aunque en este caso solamente se utilizarán en configuración punto a punto. La encapsulación utilizada y los protocolos de control que implementan las aplicaciones, responden a soluciones propietarias, debido a la ausencia de estándares estables en el momento de su desarrollo.

Finalmente, mencionar que se adquirió un tercer equipo de videoconferencia personal basado en un PC con una placa de videoconferencia (PictureTel Live200) para facilitar la realización de videoconferencias personales en las que se utilizaban aplicaciones compartidas.

La referencia [7] incluye información técnica y productos existentes de videoconferencia.

3.4. Elementos audiovisuales externos

Un aspecto clave a la hora de diseñar e implantar aulas de teleeducación son los elementos audiovisuales externos a los propios equipos de videoconferencia. En nuestro caso, se decidió dotar a las aulas de:

- Cámaras motorizadas para enfocar al profesor y a la sala.
- Un proyector de vídeo capaz de proyectar la pantalla de los ordenadores personales y del equipo de sala.
- Una cámara de documentos para proyectar opacos.
- Micrófonos de mesa e inalámbricos y otros equipos de audio.
- Una matriz de conmutación de vídeo para facilitar la selección de las distintas fuentes de vídeo.
- Una iluminación adecuada que garantizase la calidad de las imágenes enviadas.

Contrariamente a lo que se piensa, son estos equipos auxiliares los que definen en muchos casos la calidad objetiva y subjetiva que perciben los usuarios finales. De nada sirve, por ejemplo, transmitir vídeo de alta calidad si en la etapa final de su proceso, esto es, cuando se proyecta en la sala, la calidad se degrada debido a un proyector poco luminoso o de baja resolución.

Además, el control y la coordinación de todos estos elementos es una tarea compleja y que requiere a menudo una persona dedicada a su manejo. Dentro del proyecto T+T hemos investigado formas de facilitar y automatizar su control. En un apartado posterior se aborda este aspecto con más detalle.

4. Conferencia de datos. Colaboración

La videoconferencia es una herramienta muy efectiva para la comunicación entre un grupo de participantes remotos. Si el grupo necesita además trabajar sobre documentos y manejar información en común, tendrá que hacer uso de la conferencia de datos o espacios de trabajo compartido.

El estándar T.120 de ITU [8], aceptado ya por la industria, define un modelo de conferencia de datos y especifica cómo establecer y gestionar conferencias síncronas entre dos o más participantes, posiblemente conectados a través de redes distintas

Entre las aplicaciones estándar de conferencia de datos que suelen acompañar a la videoconferencia se pueden citar: la pizarra electrónica, la compartición de escritorio, la conversación (*chat*) y la transferencia de ficheros.

La pizarra electrónica emula una superficie común de trabajo que se visualiza de forma conjunta por

todos los participantes, y en ella se pueden realizar anotaciones de texto y dibujar figuras sencillas. Esta aplicación permite añadir interactividad a las sesiones de videoconferencia, porque sobre la ventana asociada a la pizarra normalmente cualquiera puede escribir o dibujar, aunque también se puede establecer una política de control de turnos. Suele ir acompañada de un elemento apuntador (*telepuntero*).

La compartición de escritorio es una extensión del concepto de pizarra. Uno de los participantes en la videoconferencia asume el papel de *master* (que puede ir rotando a lo largo del desarrollo de la sesión), y la ventana de la aplicación que ejecute se visualizará por el resto. De esta manera se pueden compartir documentos de trabajo (gráficos, ficheros de texto, hojas de cálculo, páginas HTML, etc.).

Por último, la conversación textual es útil para el envío y recepción de notas y comentarios privados entre los participantes. También puede utilizarse como apoyo a la comunicación si se producen fallos en la transmisión de vídeo y audio (algo que ocurre con cierta frecuencia).

Además de utilizar las aplicaciones de conferencia de datos que acompañaban a los equipos de videoconferencia, en el proyecto se realizaron experimentos con NetMeeting, que es compatible con el estándar T.120 y los resultados fueron bastantes satisfactorios.

5. Aula virtual

Con la infraestructura de red, el equipamiento y las aplicaciones de conferencia de datos mencionadas en los apartados anteriores, hemos llevado a cabo actividades educativas y de trabajo en grupo utilizando distintos *perfiles* de comunicación.

El modelo adoptado para todos los perfiles ha sido el modelo de *aula virtual*¹. En este modelo se trata de reproducir una situación presencial mediante el uso de videoconferencia y aplicaciones de conferencia de datos. La comunicación síncrona, la posibilidad de interacción en tiempo real y la presencia simultánea de los participantes, son los rasgos esenciales del modelo de aula virtual [9].

Los perfiles de comunicación se caracterizan más por la actitud de los propios participantes, los roles que desempeñan y la actividad que se lleva a cabo, que por la tecnología involucrada:

- Teleclase. Grupo de participantes numeroso. Un profesor o conferenciante expone. Los asistentes interaccionan y participan en turno de preguntas.

- Teleseminario. Los participantes son más activos. No sólo preguntan sino también opinan.
- Telereunión. El grupo es pequeño, se ha marcado unos objetivos comunes, hay mucha interacción y se trabaja sobre documentos.

Las ventajas del modelo de aula virtual son bastante evidentes: permite contactos visuales entre participantes, acceso a expertos en ubicaciones geográficas distantes, facilita el trabajo en grupo, motiva, etc.

Pero también tiene sus limitaciones, sobre todo en el perfil de teleclase: requiere una estrategia educativa y cambio de comportamiento del profesor en el aula. Se trata de evitar la mera transmisión de conocimiento a una audiencia pasiva, para provocar una actitud participativa centrada en una mayor interacción entre todos los actores.

6. El entorno físico y el comportamiento de los actores

Evidentemente, lo esencial de una videoconferencia es su contenido: la información visual y auditiva que intercambian entre sí los participantes. No obstante, las condiciones físicas en que se desarrolle pueden ser determinantes para el éxito o fracaso de la comunicación. Si en una reunión o seminario convencional, la claridad del sonido y de las imágenes que se presenten son importantes de cara a su adecuada ejecución, en el caso de la videoconferencia, donde los recursos técnicos en los que se apoya son limitados y determinantes, esta adecuación es fundamental para la consecución de los objetivos marcados.

Para que estos objetivos se alcancen, habrá que tener en cuenta:

- Qué tipo de reunión o de conferencia quiere realizarse.
- De qué recursos tecnológicos y ambientales se dispone.
- Cómo sacar provecho de estos recursos.

6.1. Dimensiones y configuración de la sala

El tamaño y la forma de la sala deberán adaptarse al uso propuesto para la misma. Salas más amplias tendrán más problemas a la hora de establecer una buena acústica y de prever la movilidad del equipo según los diferentes escenarios educativos antes reseñados. Por contra, permitirán mayor comodidad en los participantes cuando se requiera una audiencia numerosa a una determinada conferencia.

Para poder abarcar las distintas configuraciones, la clave es la flexibilidad y la movilidad del equipo y accesorios: mesas de asistentes, cableado, ubicación de las cámaras, colocación de los conectores, etc. Esta configuración flexible es más factible si la sala tiene un tamaño reducido (caso de la ubicada en la EUITT) pero resulta más difícil de conseguir cuando la sala tiene mayores dimensiones.

¹ El término *virtual* se ha mistificado y se usa de manera indiscriminada para calificar cualquier situación educativa que implique distancia. Aquí utilizamos una definición más restrictiva. Ver también [9].

Otro aspecto importante es el relativo a la iluminación. Debe ser homogénea y fría y no mezclar distintos tipos de luces. Es importante tener en cuenta que el balance (tanto de luminancia como de crominancia) lo hace la propia cámara, por lo que se ha de procurar establecer las condiciones óptimas para que la cámara realice un adecuado ajuste.

Por ello, es importante mantener la misma temperatura color de la luz y una difusión homogénea de la misma. Los fluorescentes producen estas características (existen fluorescentes profesionales adaptados a los requisitos de la videoconferencia). En general hay que evitar la aparición de sombras en la cara de los participantes, lo que se consigue estando la sala profusamente iluminada de forma que la luz sea difusa. Hay que prestar atención a la luz solar exterior porque puede producir desequilibrios indeseados en el balance de las cámaras.

Por esta misma razón, es importante tener en cuenta el color del fondo y del mobiliario para que produzca buen contraste con las caras, resaltando la fisonomía. Colores demasiado claros (tipo crema), producen confusión entre la piel y el fondo. Un color interesante es el azul TV (el azul básico de la señal de vídeo). Para conferencias de grupos amplios este color resulta demasiado oscuro para el encuadre general, siendo muy adecuado para enfoques cortos.

Una solución puede estar en utilizar un color más claro, que resolvería ambas situaciones, o usar algún tipo de forillo o cortina para los encuadres más cerrados del conferenciante principal.

En videoconferencia se obtiene una calidad de imagen aceptable usando anchos de banda muy restrictivos gracias a los algoritmos de codificación que consiguen que entre píxeles contiguos con una información muy parecida (caso de un fondo liso y uniforme) la información que es necesaria transmitir sea mínima. Por ello es totalmente desaconsejable introducir fondos con dibujos y representaciones.

6.2. El sonido

Es importante tener una buena acústica, sin reverberaciones ni ecos, para que sea inteligible lo que dicen los intervinientes en la videoconferencia. Existen varios tipos de micrófonos utilizables según la situación. Cabe destacar:

- Micro incorporado a la cámara. Poco interesante cuando el hablante está alejado o la cámara enfoca distintos encuadres.
- Micro de ambiente omnidireccional. Añaden espontaneidad en las exposiciones pero también ruido. Exige a los asistentes ser más comedidos que en una conferencia normal donde los diálogos en voz baja son frecuentes.
- Inalámbricos unidireccionales. Interesante para facilitar que alguno de los participantes pueda

intervenir en un momento dado. Es muy interesante para turnos de preguntas.

- Micro de corbata. Es un micro inalámbrico que permite al conferenciante gran libertad de movimientos aunque es necesario tener cuidado con el ruido, los roces y la mala colocación.

6.3. Panel de control

Un reto importante, no fácil de resolver satisfactoriamente, es conseguir que desde un punto de vista psicológico, una sesión de videoconferencia se aproxime lo más posible a un encuentro presencial. Para ello es importante que cuando algún interviniente se dirija a un público remoto éste "vea" que lo está mirando, al igual que ocurre en un encuentro convencional.

Esta situación puede conseguirse si se dispone de diversas cámaras, situadas en emplazamientos adecuados, de forma que lo que se transmite en cada momento se corresponda con lo más conveniente para facilitar la cooperación grupal. La disposición de las cámaras dependerá del escenario de comunicación que se está tratando de reproducir

Si, por ejemplo, se trata de un escenario en el que una persona está impartiendo una conferencia a dos grupos, uno local y otro remoto y está usando unos gráficos explicativos, habrá que decidir en que momento y dónde el público que asiste presencialmente y el propio conferenciante ven estos gráficos o ven al público remoto.

Para posibilitar estas *vistas* distintas es necesario disponer de una matriz de conmutación de vídeo provista de varias entradas y salidas, y de un mecanismo que permita seleccionar cómodamente la fuente de vídeo y los movimientos de cámara PTZ (panorámica, inclinación (*tilt*) y zoom). Además, deberá ser posible gestionar la apertura o cierre de los distintos micrófonos de que se dispone. Dentro del proyecto T+T se ha desarrollado una pequeña aplicación que permite manejar desde un PC, a través de una interfaz RS-232, la matriz de conmutación y los movimientos de la cámara.

Cuando se trata de una telereunión a la que asisten un número muy reducido de personas, el gobierno de las cámaras y la elección del encuadre puede realizarlo directamente uno de los asistentes sin por ello disminuir el grado de atención y de aprovechamiento del encuentro. No obstante, en escenarios más complejos será necesario que sea una persona con cierta experiencia la que maneje el panel de control para conseguir un buen resultado.

Otra función de control que puede ser interesante realizar es el reparto del ancho de banda disponible (cuando ello es posible) entre dos fuentes de vídeo que están siendo transmitidas de forma combinada. Puede tratarse de una aplicación cooperativa tipo NetMeeting a la que se añada un recuadro con la imagen del conferenciante.

6.4. Nuevos hábitos de comportamiento

Es necesario percatarse de que la nueva situación de reunión virtual que se produce en el ámbito de la videoconferencia no es igual a aquella presencial a la que estamos acostumbrados y que, por tanto, será necesario adaptar nuestros comportamientos para sacar el mayor provecho posible de los recursos tecnológicos, al tiempo que evitemos aquellas circunstancias que producen efectos indeseados.

Para empezar, tenemos que revisar el vestuario para que produzca un adecuado contraste con el fondo y con el rostro, pero evitando los blancos o los colores muy chillones. Si a alguien le gusta usar ropa de rayas o con dibujos, el día que participe en una videoconferencia tiene que optar por una solución más sobria. Como se ha dicho antes, el cambio de color y luz entre píxeles contiguos de un mismo cuadro repercute en un incremento de la información necesaria a transmitir. Si además existe movimiento (diferencia entre los píxeles equivalentes de cuadros consecutivos) la situación se complica aún más. Un conferenciante inquieto armado de una camisa de florecitas generará un aumento superfluo de tráfico que repercutirá en una menor calidad de la comunicación.

En el estado actual de la tecnología los retardos y la falta de sincronía entre la voz y la imagen son claramente perceptibles. Por ello es necesario evitar movimientos innecesarios, guardar pausas en los turnos de réplica y evitar las interrupciones.

7. Conclusiones

La puesta a punto del entorno de trabajo descrito fue laboriosa, debido a la multitud de elementos hardware y software empleados. Hay que destacar que, aún utilizando componentes de calidad, la mayoría procedía de diversos fabricantes, lo que añadía problemas de incompatibilidad entre los mismos, sobre todo en lo referente a la conexión entre el conmutador y las tarjetas ATM, lo que reveló la inmadurez de las implementaciones ATM durante el desarrollo del proyecto. Se confirmó, una vez más, que el entorno Windows 95/NT era poco robusto y nada fiable.

En resumen, la diversidad de tecnologías y componentes implicados en una sesión de videoconferencia multimedia, las incompatibilidades entre productos de distintos fabricantes y la falta de madurez de algunas implementaciones (particularmente en lo que respecta a ATM), son factores que pueden limitar las prestaciones y que hay que analizar a la hora de planificar la adquisición de equipamiento.

Al tiempo que es necesario trabajar en el perfeccionamiento de las soluciones tecnológicas que posibilitan la teleconferencia, será necesario que todos los que están interesados en el uso de esta herramienta, investiguen en la metodología más

adecuada para su uso, lo que representa entre otras cosas adaptar sus hábitos (tanto docentes como de otro tipo) a las características requeridas por este nuevo medio.

Agradecimientos

El Proyecto T+T ha sido financiado en parte por la Comisión Interministerial de Ciencia y Tecnología, a través de la Acción Especial TEL96-2589-E.

Referencias

- [1] S.Okubo et al. *ITU-T Standardization of Audiovisual Communication Systems in ATM and LAN environments*. IEEE Journal, August 97.
- [2] ITU-T Recommendation H.310. *Broadband Audiovisual Communication and Terminals*. December 1996.
- [3] L.Bellido. "A Survey of Standards for Videoconferencing over Broadband Networks". *First LEVERAGE Conference on Broadband Communications in Education and Training*. Cambridge, January 1998. Disponible en: <http://www.dit.upm.es/~leverage/conf1/>
- [4] E. Pastor, D. Fernández and L. Bellido. "Cooperative Learning Over Broadband Networks". *6th Joint European Networking Conference (JENC6)*, Tel-Aviv, Mayo 1995.
- [5] L. Bellido, D. Fernández, E. Pastor. "Architectural Issues for Multimedia Cooperative Systems". *3rd International Workshop on Protocols for Multimedia Systems (PROMS'96)*, Madrid, Octubre 1996.
- [6] Proyecto LEVERAGE: <http://www.dit.upm.es/~leverage>
- [7] SAVIE Initiative (Support Action to facilitate the use of Videoconferencing in Education): <http://www.savie.com>
- [8] ITU-T Recommendation T.120: *Data protocols for multimedia conferencing*. July 1996.
- [9] Paquette, G. "Virtual Learning Centers for XXth Century Organizations". En *The Virtual Campus. Trends for Higher Education and Training*. Verdejo, V., Davies, G. (eds), Chapman & Hall, 1998.

Experiencias en el uso del lenguaje Java para la enseñanza de Ingeniería Telemática

Juan C. Dueñas, Gabriel Huecas, Santiago Pavón, Tomás Robles, Joaquín Salvachúa
Departamento de Ingeniería Telemática Universidad Politécnica de Madrid
Ciudad Universitaria sn, 28040 Madrid
E-mail: {jcduenas, gabriel, santiago, trobles, jsr}@dit.upm.es

Abstract

Telematics Engineering plays a relevant role within the disciplines related to Telecommunications. In the past, this knowledge area has been focused towards network management and planning, but with the emergence of new technologies such as Internet and techniques from the field of client/server computing, the focus is now moving towards the development and management of services, built as distributed applications. This fact imposes more responsibility on the efforts devoted to the programming related courses in the telematic engineering syllabus. In this article, the authors review one telecommunication curricula and its contents in terms of telematics related courses, present the role of the programming language chosen for those courses, propose the usage of the Java programming language in this environment, and present the result of the experience of its application in the Telematics Department of the ETSI Telecomunicación of Universidad Politécnica de Madrid.

1. El área de Ingeniería Telemática

Tradicionalmente, la ingeniería telemática se define como el conjunto de técnicas y conocimientos necesarios para el análisis, diseño y realización de sistemas de comunicación de tipo digital. El término "telemática" refleja la componente de desarrollo de sistemas de telecomunicación, y también la relevancia de las técnicas informáticas para lograrlo.

El área incluye conocimientos sobre:

- Organización, gestión, planificación, mantenimiento y explotación de redes.
- Concepción, descripción funcional y operatividad de servicios de valor añadido.
- Técnicas de conectividad entre usuarios (conmutación de circuitos, paquetes e híbrida) y de compartición en canales de acceso múltiple.
- Lógica de interconexión de las redes de acceso, para garantía de la calidad final mediante la codificación, detección y corrección de errores, protocolos y señalización.
- Medida, modelado, simulación y análisis de la actividad o tráfico.
- Control de acceso, tarificación, seguridad, privacidad y autenticación para regular el acceso a informaciones y servicios y para certificar el origen, contenido y destino de la información.
- Descomposición, especificación, diseño, validación, desarrollo e instalación del hardware y software de comunicaciones.

El impacto industrial del área se refleja en que ocupa directa o indirectamente a la mitad de ingenieros de telecomunicación en tareas técnicas, como se desglosa en [1]:

1. 34% de ingenieros en tareas de ingeniería del software,
2. 28% en tareas de ingeniería electrónica,

3. 20% en tareas relacionadas con radiofrecuencia,
4. y 16% en tareas puramente telemáticas.

En los últimos tiempos se detecta un desplazamiento del ámbito de interés del área de conocimiento desde la creación de sistemas físicos hacia el desarrollo de nuevos servicios a partir de la infraestructura ya establecida, donde el papel predominante es la explotación de las redes existentes y la creación de servicios sobre ellas, en forma de aplicaciones distribuidas. Una vez completo el proceso de liberalización de las telecomunicaciones, el grueso principal del trabajo de los ingenieros con formación telemática consistirá en el desarrollo de estos servicios distribuidos.

Internet es un claro ejemplo de esta evolución: hace algunos años se estudiaba sólo desde el punto de vista de la interconexión, pero con la creación de nuevas aplicaciones como los navegadores de Web, sus posibilidades de explotación han abierto nuevos campos de estudio, como el comercio electrónico, basados en el enfoque de servicios.

En este nuevo escenario, es preciso reconsiderar la formación en informática incluida en las asignaturas de un currículo en ingeniería telemática. El Departamento de Ingeniería Telemática de la Universidad Politécnica de Madrid está realizando un esfuerzo de modernización de los contenidos de sus asignaturas, al hilo de la implantación de los nuevos planes de estudio, y tratando de reducir la dispersión existente, por ejemplo, en cuanto a lenguajes de programación.

Entre otros cambios, estamos evaluando el papel de los lenguajes de programación en la formación del ingeniero telemático, y más específicamente, estamos incorporando el lenguaje Java en la docencia. Este artículo proporciona algunos resultados preliminares de esta experiencia.

2. La enseñanza de Ingeniería Telemática

El título oficial de ingeniero de telecomunicación y las directrices generales propias de los planes de estudio asociados a él se regulan en el Real Decreto 1421/1991, de 30 de agosto [2]. Las distintas universidades que ofrecen la titulación particularizan este plan general de estudios.

En las directrices del título oficial aparecen varias materias troncales para el primer y segundo ciclo de la titulación. De ellas, las más directamente relacionadas con la ingeniería telemática y objeto de docencia por parte de departamentos de esta área son: Arquitectura de redes, sistemas y servicios; Fundamentos de los ordenadores; Fundamentos de la programación; Arquitectura de ordenadores y Redes, sistemas y servicios de comunicaciones.

Centrándonos en la Universidad Politécnica de Madrid, los estudios se articulan en dos ciclos de tres y dos cursos académicos, respectivamente, sin titulación intermedia. En cuanto a las asignaturas obligatorias de universidad, destaca la presencia de un número considerable de créditos de primer ciclo, cuyo fin es ampliar la formación común de todos los estudiantes sobre materias, incluyendo también un gran número de horas destinadas a las enseñanzas prácticas de laboratorio. En este sentido, es de señalar que la programación de ordenadores pasa a considerarse una técnica fundamental, por lo que se le asignan 6 créditos troncales en el primer ciclo.

La oferta de materias optativas de segundo ciclo se organiza en tres áreas de especialidad: Comunicaciones, Electrónica y Telemática. En cada una de ellas, algunas asignaturas se organizan a su vez en intensificaciones, que para el área de ingeniería telemática en la UPM son: Redes y servicios de comunicación y Sistemas informáticos, cada una con 16 créditos.

El área de conocimiento de "ingeniería telemática" tiene una presencia importante en este plan de estudios: alrededor del 42% del total de asignaturas del plan, un tercio de las asignaturas troncales y obligatorias de universidad del primer ciclo, y cerca de la mitad de las asignaturas troncales y obligatorias de universidad del segundo ciclo tienen vinculación con ella.

Tomando como elemento de relación los conocimientos del alumno, y centrándonos en los aspectos de uso de un lenguaje de programación podemos establecer una secuencia de asignaturas relacionadas, mostrada en la tabla 1. Todas las asignaturas de esta secuencia tienen como conocimientos previos necesarios los que se imparten en la asignatura "fundamentos de programación".

Tabla 1: asignaturas afines.

	Asignatura	Curso Cuatr.	Tipo
1	Fundamentos de programación" (FPROG)	1/1	Troncal
2	"Laboratorio de programación" (LPROG)	1/2	Troncal
3	"Fundamentos de los ordenadores" (FORD)	2/2	Troncal
4	"Arquitectura de los ordenadores" (AORD)	4/1	Troncal
5	"Laboratorio de programación de sistemas" (LPRS)	4/2	Troncal
6	"Ingeniería del software"	4/2	Obligatoria telemática
7	"Software de comunicaciones"	4/2	Obligatoria telemática
8	"Laboratorio de software de comunicaciones"	5/1	Obligatoria telemática

De lo anteriormente expuesto se observa la gran presencia de las asignaturas relacionadas con la programación y los ordenadores en el plan y su relevancia de cara a la especialidad telemática, seguida por cerca de la mitad del alumnado, y la evolución histórica de los contenidos en la titulación, orientados hace algunos años hacia temas de electrónica, y actualmente desplazándose hacia temas relacionados con los ordenadores y el software.

Aunque no existen pre-requisitos legales en el plan de estudios, la presencia de pre-requisitos de conocimiento convierten la asignatura de programación en pre-requisito para todas las de telemática del currículo.

2.1. La enseñanza telemática en otros entornos.

En EE.UU. los temas relacionados con la telecomunicación y la telemática se imparten por departamentos de "Computer Science" o "Electrical Engineering", que implementan en programas concretos los currículos propuestos por asociaciones profesionales como ACM o IEEE. La "Association for Computer Machinery" (ACM) y la "Computer Society" del Instituto de Ingenieros Eléctricos y Electrónicos (CS/IEEE) realizaron durante 1988 y 1989 un esfuerzo común para elaborar recomendaciones curriculares en el área de informática y afines, que dio lugar en 1991 al currículo propuesto conjuntamente por ambas instituciones [3], cuyas características más importantes frente a propuestas anteriores son el equilibrio entre la enseñanza de software y hardware, el gran peso de las enseñanzas básicas y la organización en varias áreas fundamentales como algoritmos y estructuras de datos, arquitecturas, sistemas operativos, lenguajes de programación, e ingeniería de software.

La relevancia de esta propuesta con respecto al plan de estudios UPM es relativa: la de la UPM es una titulación en ingeniería de telecomunicación y tiene aquí un aspecto finalista más que científico. Por otra parte, el paso de los años ha revelado algunas deficiencias, como se muestra en [4]. Sin embargo, la propuesta mencionada no solamente es la más solvente y reconocida internacionalmente, sino que contiene elementos de gran valor para nuestra ingeniería, entre otros:

- la potenciación de los conocimientos sobre ingeniería del software, debido a su impacto en el desarrollo tecnológico de los equipos de telecomunicaciones,
- la consideración de la programación como técnica fundamental a lo largo del currículo y su papel básico en la vida profesional,
- y el papel fundamental de los laboratorios en la formación de los ingenieros.

La introducción de docencia en aspectos telemáticos en los planes de estudios de ingeniería e informática en la universidad de EE.UU. es continua [5]. Un esfuerzo en la misma dirección (en Canadá) es el mostrado en [6], en el que se propone un plan de estudios muy parecido al vigente para la ingeniería de telecomunicación en España en sus aspectos telemáticos, con gran presencia de asignaturas relacionadas con la programación de ordenadores desde el comienzo de los estudios y a lo largo de todo el plan.

El interés por incorporar temas de telemática se extiende a otras titulaciones tradicionalmente alejadas. [7] propone un modelo de currículo para titulaciones en sistemas de información, promovido por ACM, AIS y AITP, que aúna el interés de los temas puramente informáticos (lenguajes de programación, programación y uso de sistemas operativos) con el mostrado por temas de telecomunicación. Esto muestra un cambio en la orientación desde las disciplinas científicas hacia la ingeniería de aplicaciones.

3. Programación e Ingeniería Telemática

La aparición del currículo ACM supuso también un consenso de hecho en el uso de lenguajes de programación imperativos de alto nivel para la enseñanza de disciplinas relacionadas con las tecnologías de la información, porque facilitaban la transmisión de conceptos y prácticas estructuradas de programación. Este acuerdo ha desaparecido hoy en día, para dar lugar a posturas divergentes sobre el papel de la enseñanza de la programación en estas disciplinas:

- una opción propone el equilibrio entre la enseñanza de los conceptos de programación y el uso de lenguajes ampliamente extendidos, para lo cual utilizan lenguajes didácticos (Pascal principalmente),
- otros docentes enseñan a los estudiantes a programar en alguno de los lenguajes de mayor

uso industrial, de cara a su formación para el futuro mercado de trabajo (C o C++),

- cuando el objeto fundamental son los conceptos y técnicas, el lenguaje es solamente un vehículo de expresión (se utilizan los lenguajes que mejor reflejen cada uno de los conceptos concretos),
- y enfoques matemáticos con lenguajes declarativos o funcionales.

En la titulación de ingeniería de telecomunicación, la programación de ordenadores tiene un aspecto finalista: el objetivo es aprender a programar para construir un sistema de comunicaciones o que las use. Además, debido a la relativa escasez de créditos para el aprendizaje de la programación, consideramos que en ésta área de conocimiento el enfoque más adecuado es una mezcla de los dos primeros: debe ser un lenguaje lo más cercano posible a la práctica profesional, pero no tanto como para hacer difícil la comprensión de los conceptos básicos de programación.

En cualquier caso, la práctica parece ir en la dirección expresada: aunque lo más importante sea la enseñanza de los conceptos, éstos deben expresarse en un lenguaje de programación, del cual debe enseñarse su sintaxis y semántica. Por otra parte, la pretensión de enseñar varios lenguajes de programación al tiempo es impracticable.

Respecto al uso de lenguajes imperativos, es preciso realizar una puntualización. El mismo currículo ACM en el año 1991 propone la enseñanza prioritaria de este tipo de lenguajes, para en un momento más avanzado de la formación revisar otras técnicas. Está demostrado que la mayoría de los sistemas que antes se realizaban con esos lenguajes imperativos pueden realizarse usando lenguajes derivados de aquellos, pero con una característica añadida: la orientación a objetos.

Estas técnicas tuvieron su empuje inicial en la necesidad industrial de reducir los esfuerzos en el desarrollo de programas, bien mediante la reutilización promovida por la herencia, bien mediante la mejora de la mantenibilidad que permite el polimorfismo. Durante años, la falta de atención que los centros de formación han mostrado a estas técnicas, ha exigido a las empresas fuertes desembolsos en formación de los recién titulados, lo que ha provocado el florecimiento de una industria docente al margen de la oficial en la que se enseñan, pobremente, estas técnicas.

Cierto es, por otra parte, que la falta de acuerdo sobre su bondad ha sido un impedimento para su adopción generalizada, pero se puede afirmar que ya se han impuesto en la industria relacionada con la creación de software, incluido el de telecomunicación. Comienzan a aparecer estándares internacionales sobre su uso en las fases de diseño, una vez que las empresas ya tienen sus

normas "de facto" para las fases de diseño e implementación.

Por nuestra parte, creemos necesario añadir la orientación a objetos en un plano de igualdad con los principios algorítmicos básicos. Ambas técnicas son la manifestación de los conceptos recurrentes en la programación: la abstracción y la ocultación de información. Desde el punto de vista práctico, los lenguajes de programación orientados a objeto más habituales permiten en cierta medida seguir usando los principios de la programación estructurada, que al margen de lo que algunos puristas afirman, pueden formar parte de las primeras técnicas que se enseñan.

Quisiéramos resaltar un concepto fundamental que pensamos deberá hacerse cada vez más presente en cualquier aproximación a la docencia de este tema, y en especial en aquellas titulaciones como la nuestra en las que la programación es una herramienta más que un fin en sí misma: la orientación de la asignatura hacia los métodos ingenieriles.

La programación de ordenadores debe entenderse en un marco más amplio que la realización de programas que puedan ser compilados y ejecutados eficientemente en una máquina. Esta actividad debe comprenderse dentro del proceso de ingeniería que consiste en tomar decisiones y comunicarlas a otros, para lo cual se requiere un cambio de paradigma en la enseñanza de la programación, que incorpore de forma esencial los aspectos de comunicación humana:

- el razonamiento sobre las decisiones de diseño e implementación, soportado por medidas cuantitativas y cualitativas,
- y la cooperación entre agentes del desarrollo, que se concreta en los aspectos formales del proceso de ingeniería (estándares de diseño y codificación) y en la consideración de la mantenibilidad y la legibilidad como fines tan importantes como la eficiencia.

4. Elección del lenguaje de programación

Vistos los diferentes aspectos bajo los que se puede enseñar la programación de ordenadores, la siguiente pregunta es: ¿Cuál lenguaje de programación enseñar?. No se trata sólo de mostrar el lenguaje, porque deben incluirse elementos de teoría de programación, pero tampoco debe darse un enfoque puramente teórico (más propio de otras titulaciones en las que la programación es un fin en sí mismo), puesto que otras asignaturas basan su desarrollo en el conocimiento práctico de la programación en algún lenguaje. Además, dado el escaso número de créditos asignados a esta tarea, es muy posible que los alumnos sólo puedan llegar a conocer y utilizar uno, a lo sumo dos lenguajes de programación distintos en todos sus estudios.

Una guía de los elementos significativos para la elección del lenguaje puede encontrarse en [8]:

- existencia de estándar del lenguaje,
- extensión y significación del uso del lenguaje en el mundo profesional,
- facilidad del lenguaje para la construcción de interfaces de usuario,
- pendiente de la curva de aprendizaje,
- coste para el alumno de obtener un entorno fiable de desarrollo,
- posibilidades de uso del lenguaje para su uso en cursos posteriores,
- credibilidad del lenguaje entre los estudiantes,
- seguridad del lenguaje,
- disponibilidad de herramientas para depuración,
- estabilidad de la especificación del lenguaje,
- facilidad para la extensión,
- disponibilidad y calidad de los libros de texto,
- buen soporte pedagógico que facilite las buenas prácticas de programación.

Aunque la lista presenta elementos de decisión que en un primer momento pueden parecer poco adecuados (credibilidad entre los estudiantes, por ejemplo), tras un análisis más profundo, encontramos que incluso esos elementos son importantes, porque influyen en la motivación de los alumnos por la asignatura.

La situación actual sobre el lenguaje más comúnmente utilizado para la introducción a la programación refleja un escenario en evolución: la mayoría de las universidades y escuelas técnicas de los estudios [9] [10] [11] siguen utilizando Pascal como primer lenguaje; sin embargo, una gran parte de ellos han detectado la necesidad de evolucionar hacia un lenguaje que incorpore construcciones más modernas. Hasta hace pocos años, Ada parecía el lenguaje más adecuado para sustituir a Pascal, y las escuelas que realizaron ese cambio informan de resultados muy satisfactorios. Las mayores dificultades en la utilización de Ada surgen de su poca credibilidad entre los estudiantes, de la ausencia de bibliotecas del lenguaje disponibles para los alumnos y de la falta de repercusión en el mundo profesional, lo que obligaba a algunas de las escuelas a incorporar otros lenguajes (C o C++) en cursos posteriores.

Con la aparición de Java [12], la situación presenta visos de cambio [13]. Prácticamente todas las escuelas que estaban utilizando C o C++ como primer lenguaje, y muchas las que todavía usan Pascal han manifestado su interés en estudiar las posibilidades de Java.

Java es un lenguaje de programación orientado a objetos para la creación de aplicaciones, cuya orientación inicial era la de realizar interfaces de usuario y programas para Internet. En pocos años, el lenguaje ha crecido extraordinariamente, así como el número de entornos de desarrollo y bibliotecas; se ha extendido su utilización para programación en general (excepción hecha de la programación de

sistemas), e incluso se ha propuesto para su estandarización internacional. Las características esenciales del lenguaje ya están bien establecidas, y aunque la batalla comercial para su implantación todavía no ha terminado, comienza a considerarse como un lenguaje muy adecuado [14]. El interés sobre este lenguaje ha alcanzado también a la universidad. Encontramos varias referencias sobre talleres y congresos sobre su uso docente [15], cuyas conclusiones nos han movido a su elección:

- es un lenguaje sintácticamente sencillo,
- introduce el paradigma de orientación a objetos desde el principio,
- promociona el razonamiento sobre el uso, más que sobre la implementación; lo que se refleja en la enorme cantidad de bibliotecas disponibles para casi cualquier área de aplicación,
- existen varios entornos de programación fáciles de utilizar, baratos y de calidad profesional,
- su credibilidad entre los docentes y los profesionales es alta, al haber sido concebido como una evolución de lenguajes conocidos,
- utiliza la sintaxis que se ha estandarizado de hecho en la programación: aquella derivada del lenguaje C. No es la óptima y en algunos aspectos es mala, pero es la que los ingenieros se van a encontrar en el mundo real.
- se adecua perfectamente a los contenidos que los ingenieros de telecomunicación van a encontrarse durante el resto de sus estudios (uso de Internet, programación concurrente, etc.).

Pero teniendo en mente la ingeniería telemática y los nuevos retos a los que debe hacer frente, mencionados en la primera sección, Java toma una mayor fuerza en los estudios de telecomunicación, porque:

- es un lenguaje concebido para el desarrollo de sistemas distribuidos, con facilidades como la reflexión, serialización, y soporte de interfaces incluidas en la definición del propio lenguaje (no como bibliotecas), y
- contiene, como parte del lenguaje, elementos para la representación y gestión de la concurrencia, por lo que no es necesario bajar al nivel del sistema operativo.

No queremos pasar por alto el hecho de que todavía no existe un consenso generalizado sobre el uso de Java como primer lenguaje [16], aunque pensamos que las ventajas objetivas que presenta en este momento superan a los inconvenientes.

5. Experiencias de uso docente

Como ya se ha mencionado, el Departamento de Ingeniería de Sistemas Telemáticos de la UPM está utilizando el lenguaje Java para la docencia de las asignaturas del área con contenidos en programación de ordenadores. Esta experiencia es por el momento limitada (en algunas asignaturas se trata de la primera o segunda impartición). A continuación incluimos la evaluación subjetiva por parte de los profesores del papel que Java juega como lenguaje

de programación en las asignaturas "Fundamentos de programación", "Laboratorio de programación" y "Laboratorio de software de comunicaciones".

5.1. Fundamentos de programación

La asignatura es troncal y se imparte en el primer cuatrimestre del primer curso, lo que supone que no hay pre-requisitos en su impartición, por lo que deberán tenerse en cuenta las características de los estudiantes recién ingresados en la universidad, y que las técnicas que se enseñan en la asignatura deberán ser útiles en cualquiera de las actividades que como ingenieros desarrollen en el futuro. El hecho de que es la primera asignatura que muchos de los alumnos cursan sobre temas informáticos la hace especialmente singular en cuanto a su tratamiento e impartición.

Los objetivos de la asignatura son aprender a elegir y diseñar las estructuras de datos y algoritmos necesarios para modelar un problema y su solución, conocer un lenguaje concreto de programación (Java) y aprender a estructurar los programas mediante construcciones que faciliten la comprensión y adaptación posterior.

El temario de la asignatura (pueden encontrarse en <http://www.dit.upm.es>) consiste en una secuencia que muestra las técnicas de programación en orden de dificultad y novedad creciente. Esto es, los primeros bloques describen técnicas que pueden utilizarse para la realización de programas muy pequeños en cuanto a tamaño, tiempo de vida o número de usuarios; a medida que se avanza en el temario, se describen técnicas más avanzadas, útiles para desarrollos con equipos más numerosos. Algunos elementos novedosos del temario son:

- la presentación incremental de construcciones para el control de la complejidad, mediante las abstracciones "función", "clase" y "módulo".
- la introducción de las excepciones entre las sentencias básicas de control, que se justifica por la necesidad de abordar la gestión de errores de forma limpia y mantenible.
- la presentación de aspectos de orientación a objetos en un punto temprano del programa.
- la desaparición del concepto de gestión explícita de memoria. En lugar de ello, se presenta el concepto de referencia, que en el lenguaje de programación que utilizamos, soluciona los problemas de gestión explícita, demasiado complejos para los alumnos de la asignatura.
- la propuesta de ejercicios de realización obligatoria en el laboratorio hasta obtener programas que funcionen. Esta estrategia docente ha dado buenos resultados en el pasado, y es un refuerzo positivo para los alumnos.

En años anteriores, la asignatura tomó como lenguaje de programación TurboPascal. Las mejoras más evidentes logradas al adoptar Java han sido:

- la capacidad del lenguaje para ilustrar los principales conceptos que pretendíamos introducir en el curso,
- el hecho de ser un lenguaje orientado a objetos frente a la orientación a objetos de TurboPascal, que es mixto,
- la mayor proyección industrial de Java,
- la existencia de entornos de libre distribución que permiten que los alumnos realicen sus ejercicios y prácticas en casa, y
- la popularidad del lenguaje, aspecto muy importante para incrementar la motivación de los alumnos.

También se plantearon argumentos en contra:

- Al no tener Java gestión explícita de memoria por parte del programador, puede surgir una laguna en la formación práctica de los alumnos. Contra este argumento concluimos que en un curso tan básico siempre habíamos observado que la auténtica gestión de memoria siempre quedaba fuera del temario, o era tratada de forma muy superficial. Sí es posible construir estructuras dinámicas convencionales (pilas, árboles, etc.), dejando para cursos posteriores la explicación más en detalle de la gestión de memoria.
- La inexistencia de libros adecuados como libros de texto. Tras varios meses de evaluación resolvimos llegar a un equilibrio entre extensión y calidad en cuanto al texto, y hemos utilizado [17], que cumple los requisitos mínimos de temarios, secuenciación, profundidad y cantidad y calidad de ejemplos, como para constituirse en el libro de referencia básico.
- El entorno de laboratorio disponible está equipado con máquinas PC con sistema operativo Linux. No hemos encontrado entornos de desarrollo gratuitos y con la calidad y velocidad adecuada. Haciendo de la necesidad virtud, pensamos que sería tan útil o más para los alumnos editar sus programas con cualquier editor (vi, emacs, etc.) y compilarlos con las facilidades mínimas (Java Development Kit).

La asignatura "Fundamentos de programación" finalizó su primera experiencia de uso de Java en Febrero de 1999. Las conclusiones más notables de esta experiencia se pueden resumir en los siguientes puntos:

- El grado de dificultad en cuanto a la introducción de los conceptos básicos de la programación estructurada (secuencia, condición e iteración) y estructuras no dinámicas de datos es similar al de otros lenguajes (Ada, Pascal). Incluso la introducción de las excepciones y su manejo ha resultado natural y sin mayor complejidad de aprendizaje.
- Las dificultades más importantes surgen por la necesidad de enseñar principios de diseño, puesto que las construcciones del lenguaje reflejan estos principios de forma

más directa. Esta temprana aparición del diseño en la codificación ha tenido una incidencia de difícil evaluación, pues es fácil mezclar detalles de organización de código con decisiones de implementación.

- La complejidad de la jerarquía de clases dedicada a la entrada/salida en Java obliga a los profesores a proporcionar a los alumnos nuevas clases fuera de las estándar, que oculten esos detalles.
- La opinión generalizada de los alumnos es positiva, especialmente entre los alumnos repetidores, que encuentran el lenguaje más sencillo.
- De la misma forma, la opinión general de los profesores que imparten la asignatura es positiva. Es destacable que, siendo el primer curso completo con el nuevo lenguaje, todos los profesores se sienten muy satisfechos sobre como se ha desarrollado, y del grado de cumplimiento del temario (lo que suele ser infrecuente en el primer año después de un cambio tan radical en la asignatura).
- Si las notas pueden ser una medida de la calidad de la docencia, en la convocatoria de febrero del curso pasado aprobaron el 50% de los presentados, mientras que en la de febrero de este curso fueron el 60%. Por otra parte, mientras que en anteriores exámenes de febrero el número de presentados se mantenía en el 75% de alumnos, en esta convocatoria se ha rozado el 90%.

5.2. Laboratorio de programación

El Laboratorio de Programación se imparte en el segundo cuatrimestre del primer curso. Complementa a la asignatura de Fundamentos de Programación, aportando la experiencia práctica asociada a los conceptos teóricos presentados en esta. Consta de 4,5 créditos dedicados íntegramente a la realización de prácticas de laboratorio, suponiéndose que los alumnos ya han adquirido los conocimientos necesarios del lenguaje en el que van a trabajar.

Tras cursar esta asignatura, se pretende que los alumnos sepan manejar un entorno de programación concreto, puedan escribir, ejecutar y probar programas diseñados por ellos mismos, y puedan distinguir claramente entre la definición de un problema, las estructuras de datos que se emplean en su resolución, y la implementación de las mismas.

Los temas que se abordan en las prácticas de la asignatura tienen como objetivo profundizar en: las jerarquías de herencia y polimorfismo típicas de la programación orientada a objetos, el uso de la recursividad, la selección de algoritmos clásicos y la programación gráfica.

El impacto que va a tener la introducción de Java en esta asignatura podrá medirse cuando finalice este cuatrimestre. Esta es la primera vez que se imparte. Sin embargo, pueden adelantarse algunos datos que

se han observado en los que va de curso. Al usarse para todas las asignaturas de laboratorio el mismo entorno de desarrollo Java, de dominio público y multiplataforma, el alumno puede instalarse en casa un entorno similar, facilitando en gran medida su trabajo personal.

También se ha observado que la carencia de punteros en el lenguaje mejora el aprendizaje. Con otros lenguajes, el alumno pasaba la mayor parte del tiempo solucionando problemas de gestión de memoria, que producían bloqueos del ordenador, resultados aleatorios, etc. Con Java no existe este problema, y el alumno se centra completamente en la resolución de las prácticas que debe realizar, sin ese tipo de distracciones.

Sin embargo, en el uso de Java en el "Laboratorio de programación" no todo es positivo. Existen también algunos inconvenientes en la introducción de este lenguaje. El principal de ellos es la falta de documentación adecuada. Existen numerosos textos sobre la programación con Java, pero prácticamente todos están orientados a la construcción de Applets, o se dirigen a programadores con experiencia. Incluso el texto recomendado para "fundamentos de programación" sufre de esta limitación.

Otro inconveniente está relacionado con la facilidad de uso de los entornos de programación. No existe un entorno de dominio público que sea lo suficientemente amigable. Por ejemplo, en el laboratorio usamos JDK 1.1.6, que presenta un entorno de depuración inmanejable para el alumno de primer curso. Esto obliga a que la depuración de programas se realice combinando la inspección visual con la utilización de trazas. Otro ejemplo es la ausencia de herramientas de compilación condicional.

Hay que comentar también que la introducción de este lenguaje ha obligado a rehacer parte del soporte de gestión del laboratorio existente. Se han reescrito los programas de evaluación de las prácticas y de detección de copia entre alumnos. Se ha modificado el mecanismo de publicación de soluciones precompiladas, puesto que existen descompiladores (aunque en general los alumnos de primer curso desconocen su existencia).

Por último, mencionar que lo único que no ha variado ha sido el mecanismo de evaluación de los alumnos, aunque puede que sea necesario hacer algún reajuste según la experiencia que obtengamos al final de este curso.

5.3. Laboratorio de software de comunicaciones

Uno de los principales objetivos docentes de los laboratorios de asignaturas de ingeniería telemática como el "Laboratorio de software de comunicaciones" es que el alumno sea capaz de reconocer los problemas de análisis, diseño,

implementación y pruebas de programas de comunicaciones.

Concretamente, que sea capaz de diseñar y llevar a la práctica programas de comunicaciones:

- Ser capaz de diseñar un protocolo adecuado a los requisitos de cada caso.
- Ser capaz de seleccionar el protocolo adecuado de los existentes si se adecua al caso.
- Ser capaz de implementar dicho protocolo utilizando técnicas de programación modernas.
- Conocer y manejar las normas internacionales de interconexión.
- Conocer y manejar las normas de facto disponibles.

Para conseguirlo, y realizar una práctica realista, es necesario que el alumno domine aspectos no directamente relacionados con el problema como son el lenguaje a utilizar, el sistema operativo y la concurrencia.

Durante la última impartición del curso, se han realizado prácticas sobre: implementación de una variante de protocolo similar a TCP sobre UDP, mejora de HTTP 1.0 a HTTP 1.1, implementación de un envío de imágenes sobre RTP y realización de un servidor de Chat sobre RMI.

La experiencia previa en esta asignatura muestra que gran parte de los alumnos tienen lagunas de conocimiento en estos temas, lo que implica una necesidad de nivelación previa al comienzo de las prácticas. Se han llevado a cabo tres tipos de enfoques a este respecto:

- El uso del lenguaje C y UNIX, que lleva a necesitar la mayor parte del tiempo en conseguir que el alumno tuviera un control mínimo del entorno. Por ello las prácticas debían ser sencillas y con un alto nivel de guía, por lo que no se cumplían plenamente los objetivos propuestos.
- Experiencias con el lenguaje C++, que tampoco fueron satisfactorias. El altísimo grado de complejidad del lenguaje llevó a utilizar únicamente un subconjunto muy reducido mediante unas jerarquías de objetos predeterminadas, con lo que la fase de diseño y codificación quedaban seriamente afectadas.
- Las experiencias actuales con Java son muy satisfactorias. El hecho de incorporar comunicaciones y concurrencia en el lenguaje implica minimizar el aprendizaje del lenguaje e interacción con el sistema operativo. Por ello se puede dedicar más tiempo al estudio de los problemas de las comunicaciones y los protocolos, en vez de profundizar en temas accesorios a los objetivos didácticos como ocurría anteriormente.

Referencias

- [1] S. Lorente. "Telecos en el ojo del huracán", Revista BIT, COIT, nº 104, julio 1997.
- [2] BOE. "Real Decreto 1421/1991, de 30 de Agosto, por el que se establece el título universitario de Ingeniero de Telecomunicación y las directrices generales propias de los planes de estudio conducentes a la obtención de aquél". BOE nº 243, 10 de octubre de 1991.
- [3] ACM/IEEE-CS Joint Curriculum Task Force. "Computing Curricula 1991". Report, ACM Press & IEEE Computer Society Press, 1991.
- [4] F. Sáez. "Reflexiones sobre el modelo educativo universitario de informática (hacia una concepción sociotécnica de la tecnología)", Aula Fundesco, Fundesco, febrero 1993.
- [5] J. J. Uhran. "Special Issue on Engineering Education". IEEE Communications Magazine, Vol. 28, nº 12, diciembre 1990.
- [6] D. C. Coll. "Communications Engineering: A New Discipline for the 21st Century", IEEE Transactions on Education, vol. 37, nº 2, mayo 1994.
- [7] G. B. Davis, J. T. Gorgone, J. D. Couger, D. L. Feinstein, H. E. Longenecker. "IS97 Model Curriculum and Guidelines for Undergraduate Degree Programs in Information Systems", ACM, AIS, AITP, 1997.
- [8] F. Culwin. "A Report on the 'Java in the Computing Curriculum Conference' held at South Bank University, London, on 4th February 1997", ACM SIGPLAN Curricular Patterns, 1997. Versión extensa disponible en: <http://www.scism.sbu.ac.uk/jfl/sanjose/sanjose.html>
- [9] G. Ford. "A Progress Report on Undergraduate Software Engineering Education". Informe técnico, Software Engineering Institute, Carnegie Mellon University, CMU/SEI 94-TR-11.
- [10] S. Pawlan, A. Hancock. "Computer Language Usage in CSI: Survey Results". SIGCSE Bulletin, ACM, Vol. 27, nº 3, septiembre 1995.
- [11] Department of Computer Science. "First Course Language for Computer Science Majors". Department of Computer Science, Michigan State University. Versión de abril de 1998. Disponible en: <ftp://ftp.cps.msu.edu:pub/arch/>
- [12] K. Arnold, J. Gosling. "The Java Programming Language", Addison-Wesley, 1996.
- [13] Sun Microsystems. "Java in Education". Accesible en: <http://www.sun.com/products-n-solutions/edu/java/index.html> 1998.
- [14] F. Hosch. "Java (TM) as a First Language: an Evaluation", SIGCSE Bulletin, ACM. Vol. 28, nº 3, septiembre 1996.
- [15] D. Lea. "Some Questions and Answers about Using Java in Computer Science Curricula". Informe técnico de State University of New York at Oswego. Disponible electrónicamente en: <http://gee.cs.oswego.edu/dl/html/javaInCS.html>
- [16] M. J. Pont. "Why Java is Dangerous". IEEE Software, Vol. 15, nº 1, enero 1998.
- [17] Deitel, Deitel. "Cómo programar en Java", Prentice Hall Internacional, 1998.

Acceso a bases de datos de imágenes de radiología mediante ActiveX scripting

J. Álvarez, C. del Barrio, M. de Cabo, E. Fernández, G. García y M. Rodríguez
Departamento de Teoría de la Señal y Comunicaciones e Ingeniería Telemática
Camino del Cementerio, s/n, 47011 Valladolid
Telf: 983-423666, Fax: 983-423667
E-mail: marcab@tel.uva.es; eusfer@tel.uva.es; manrod@tel.uva.es

Abstract:

In this paper we introduce an intranet design based on the World Wide Web (WWW) to integrate radiology images, text and data in hypermedia format. It is based on a three-tier model where the database server is accessed through the Web server not directly from the requesting browser. We used ActiveX scripts to process and establish connections with the database server. ActiveX scripts allow for dynamic HTML creation needed to customise, create, and deliver user-defined Web pages. The database design was based on which kind of data had to be stored and how this data were related. We have made tests to the systems and we have proven their functionality and performing.

1. Introducción

Este proyecto surgió de un estudio detallado de las necesidades de interconexión de diferentes sistemas médicos para conseguir realizar aplicaciones médicas a través de ordenador de forma que se agilizará el servicio y se mejorará el rendimiento. Este estudio se llevó a cabo en el Hospital de Medina del Campo (HMC), Valladolid [1]. Profesionales médicos y técnicos colaboraron para establecer los requisitos necesarios para que los sistemas a implantar cubrieran todas las necesidades y carencias encontradas.

Los requisitos más complejos aparecieron en el área de radiología. Este área es la que más flujo de información genera, y la que más variedad de tipos de información soporta, imágenes fijas, texto, datos demográficos de los pacientes, etc. Por ello fue esta sección la que elegimos para empezar nuestro trabajo. Este documento se centra en la realización de una aplicación pensada para mejorar el trabajo en esta área, aun cuando el esquema utilizado sea fácilmente extensible a otros servicios hospitalarios.

Los médicos responsables de este servicio estaban interesados en un sistema de almacenamiento que les permitiera un acceso rápido y simple a la información que se iba generando. Además se quería dar acceso a esta información a otras secciones del hospital e, incluso, a otros centros sanitarios que pudieran necesitarla.

Para ello se pensó en crear una intranet basada en el WWW con conexión a una base de datos multimedia para ofrecer acceso a la información a aquellos usuarios autorizados para ello.

El hecho de utilizar una interfaz web permite el acceso al sistema desde cualquier plataforma o entorno que disponga de un navegador, evitamos así imponer restricciones al equipo que accede al sistema; servirá cualquiera de los sistemas existentes (Windows, UNIX, Linux, etc.)

Aunque aquí nos encontramos con el problema de que el lenguaje entendido por estos navegadores, HTML, es estático, no puede ser programado. No podemos utilizar variables, bucles, condiciones, etc. Para resolver este inconveniente la industria de software ha tomado dos direcciones.

En la primera se trata de extender la capacidad del propio lenguaje HTML añadiendo facilidades para permitir la programación. Esto nos lleva a extensiones del lenguaje como DHTML, o a lenguajes diferentes como los applets Java o los scripts ActiveX que se interpretan en el cliente. En este caso el programa se transmite al cliente junto con la página web y es ejecutado localmente. El cliente ha de ser capaz de entender este lenguaje con lo que perdemos la libertad de elegir el tipo de navegador ya que no todos entienden el mismo tipo de extensiones de HTML.

La segunda solución evita perder la libertad de elección del navegador. Se basa en la utilización de HTML estándar 100%. El programa que queremos ejecutar con cada página corre en el servidor. Este trabaja con los datos y luego prepara el resultado para su presentación usando HTML. Esta fue la opción elegida.

Dentro de la opción de ejecutar código en el servidor aparecen los scripts ActiveX interpretados en el servidor y las ASPs [2] (Active Server Pages). Podemos considerar las ASPs como una evolución de los CGI [3] (Common Gateway Interface), otra forma de programación en el servidor. Las ASPs contienen ActiveX scripts que son interpretados en el servidor de modo que al cliente web sólo se le envía código HTML estándar.

Este tipo de páginas activas forman parte de la funcionalidad incluida en el servidor web Microsoft IIS (Internet Information Server). Con el uso de esta opción somos dependientes de un tipo concreto de servidor, pero somos capaces de enviar contenidos a cualquier tipo de navegador.

La ejecución del código en el servidor tiene varias ventajas:

- El código está almacenado de una forma segura porque el código nunca es visto por la máquina cliente, ésta sólo ve los resultados generados.
- El servidor puede gestionar los recursos más fácil y eficientemente, por ejemplo, las conexiones a la base de datos.
- Es posible tener el código a ejecutar (servidor web) y los datos a manipular (base de datos) en la misma máquina, de forma que el tráfico en la red se ve reducido.
- Se puede generar código complejo sin el problema de que no sea interpretable por el cliente.

Las ASPs muestran algunas ventajas respecto a CGI debido al uso de ADO (Active Database Object). ADO se comunica con la base de datos una vez recibe una petición, traduciendo la información a un formato entendido por la base de datos. El servidor de base de datos procesa la petición y devuelve los resultados, que serán de nuevo traducidos por ADO a un formato que pueda entender nuestro programa ASP.

Una vez realizada la intranet en el HMC, los centros de asistencia primaria dependientes de este hospital pueden aprovechar las ventajas de tener accesible la información de la sección de radiología desde su ubicación habitual, sin tener que solicitar de antemano la información y sin tener que desplazarse hasta el hospital para conseguirla. Superamos de este modo las distancias que separan los centros asistenciales rurales del centro hospitalario. Aunque por el momento la conexión hospital-centros asociados es actualmente a baja velocidad, ya que el único medio disponible para la conexión es el uso de POTS. En el momento en que en estas zonas esté disponible un acceso más rápido (por ejemplo RDSI) este inconveniente será superado.

2. Arquitectura y componentes del sistema

Una vez se estableció la necesidad de una base de datos accesible mediante Web, se pensó en un modelo de tres capas: el navegador, el servidor web y la base de datos, según se representa en la Fig. 1.

El sistema está basado en los componentes hardware y software siguientes:

- Ordenador cliente con conexión a Internet y un navegador del tipo *Netscape 4.01* o *Microsoft Internet Explorer 4.0*. El navegador debe estar configurado para aceptar e implementar *cookies* (para poder gestionar sesiones de usuarios).
- Ordenador que funciona como servidor Web en el que se encuentra instalado el IIS (*Internet Information Server*) 3.0 (o una versión posterior 4.0).

- Las páginas ASP deben encontrarse instaladas dentro del servidor Web para el caso del IIS 3.0 (o en un ordenador al que tenga acceso el servidor mediante red local de windows) en un directorio virtual, junto con permisos de ejecución.
- ODBC (*Open DataBase Connectivity*), como API entre el servidor Web y el servidor de base de datos. Esto permite conectarse con cualquier base de datos, mediante el *driver* ODBC apropiado.
- SQL Server 6.5 o posterior de Microsoft como servidor de base de datos.

El usuario entra en el sistema e inicia el navegador web, éste establece una conexión con el servidor web y le pide los datos buscados. El servidor web gestiona la conexión con la base de datos y la selección, inserción y borrado de los mismos

2.1 Servidor Web.

Es el encargado de conectar con la base de datos. El acceso desde servidor Web al servidor de la base de datos se ha realizado en los últimos años, mediante *scripts* ejecutables (CGI, VBScript, etc.). De esta manera, el servidor Web se transforma en una pasarela que reproduce información almacenada en el servidor de base de datos.

ASP es una tecnología implementada en IIS que permite realizar una programación en Internet utilizando como interfaz del usuario documentos HTML y como lenguaje de programación cualquiera que esté soportado por la versión ASP (de momento *JScript* y *VBScript*), de modo que los documentos de la *Web* incluyen tanto HTML como sentencias de programa. Al recibir una petición sobre dicho documento el servidor lo interpreta y genera un HTML estándar que, finalmente es transmitido al cliente.

Un archivo ASP puede contener tanto el *script* de la parte servidor, como el código de la parte cliente, si se encuentra añadido al código HTML estándar. Cuando un cliente solicita un fichero ASP, el servidor carga el fichero en memoria y pasa el código de servidor al motor de *scripts* correspondiente. Cualquier otro elemento en el archivo (incluyendo cualquier código cliente) finaliza en la respuesta HTTP que se devuelve al cliente.

Una de las grandes ventajas de ASP es que permite utilizar objetos *ActiveX* que se ejecutan en el servidor, de modo que soporta acciones como acceso a bases de datos mediante el modelo ADO (*Active Data Object*). ADO proporciona acceso fácil a cualquier fuente de datos compatible con OLE/DB u ODBC, por ejemplo Microsoft Access, Microsoft SQL Server, así como también Oracle, Informix, y Sybase. El inconveniente es que para poder utilizar objetos *ActiveX* no basta con conocer HTML, sino

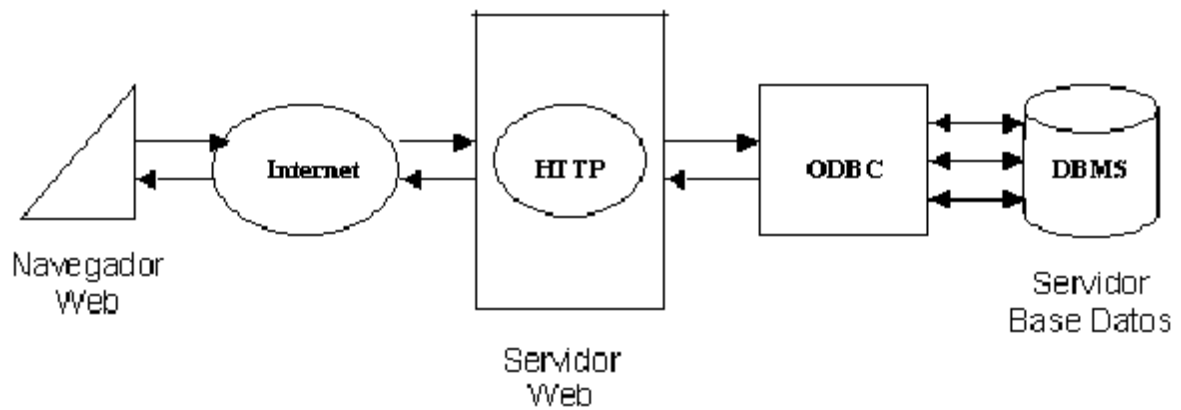


Figura 1: Acceso a la información de la base de datos

que hay que aprender a programar en dicho modelo de objetos.

Al admitir la creación de objetos *Active X* en el servidor, la conexión a cualquier base de datos se realiza de modo sencillo utilizando componentes ADO a través de ODBC. Al realizarse el acceso a la base de datos desde el servidor, debe existir un DNS de ODBC en la máquina que apunte a la base de datos deseada.

Los desarrollos sobre ASP permiten un control total sobre el diseño del documento final. La única desventaja puede ser el gasto de recursos que se produce en el servidor de Internet, pues éste debe interpretar las líneas del fichero y almacenar en memoria todas las variables creadas. Aunque este problema es común a este tipo de herramientas y se puede solucionar con un estudio previo de la utilización del servidor

2.2 Servidor de base de datos

Una aplicación escrita para una BD específica era difícil en el pasado de traducir o migrar a un servidor de BD distinto, porque cada fabricante de BD tenía una API o protocolo particular en la que estaban escritas las aplicaciones. ODBC ofrece una respuesta a este problema mediante la definición de una API en el lado del cliente que es independiente de la BD que haya en el otro extremo. De esta forma, y por medio del controlador ODBC, se puede acceder a la BD de forma transparente.

ODBC reduce en gran medida el esfuerzo necesario en caso de que se tenga que traducir una aplicación de BD a otra distinta, porque aproximadamente el 80% del código es independiente de la BD. Para poder utilizar una base de datos es necesario registrarla como fuente de datos ODBC.

El sistema gestor de base de datos (RDBMS) utilizado en este caso ha sido Microsoft SQL Server.

El diseño de la base de datos es parte importante en la realización de los ASPs encargados de acceder a la misma. Los ASPs han de conocer la estructura de

tablas contenidas para realizar las consultas pertinentes.

Para el diseño del esquema relacional de la base de datos nos basamos en un modelo entidad-relación (E-R) (ver Fig. 2.) desarrollado para el servicio de radiología. Este modelo se estableció a través de contactos con los responsables del servicio.

En el diagrama E-R hemos representado al *Paciente* como una persona que solicita una *Consulta* que pasa un *Médico*. Cada consulta genera un *Estudio* compuesto de varias *Imágenes* y como mucho un *Informe*. Hemos considerado que el informe puede realizarse un tiempo después de generadas las imágenes, por ello dentro de un estudio puede existir o no informe, pero cuando éste exista será único. Si fuera necesario agregar o modificar información el informe tiene *Correcciones*. Las correcciones aparecen porque el informe una vez emitido no puede modificarse. Si fuera necesario variar algo se creará una corrección, dispondremos así de un historial de las modificaciones realizadas al informe desde la primera versión hasta el diagnóstico definitivo. El informe tendrá tantas correcciones como sea necesario.

Por otra parte un *Paciente* tiene una o varias *Historias Clínicas*. La historia clínica de un paciente agrupa, a modo de carpeta, todos los estudios realizados a un paciente. En la actualidad el historial es físicamente una carpeta que guarda toda la información generada en relación con el paciente.

La historia clínica es la forma de diferenciar una serie de datos de un paciente de otra serie de datos del mismo o de otro paciente. Todo paciente que pasa por el HMC tiene asignadas una o varias historias clínicas. Un paciente que pertenece al HMC cuando llega al mismo se le abre un historial con un número único (historia definitiva), pero si un paciente que no es de la zona llega al mismo por algún motivo, como por ejemplo una urgencia, se le asignará un historial provisional. Si este paciente

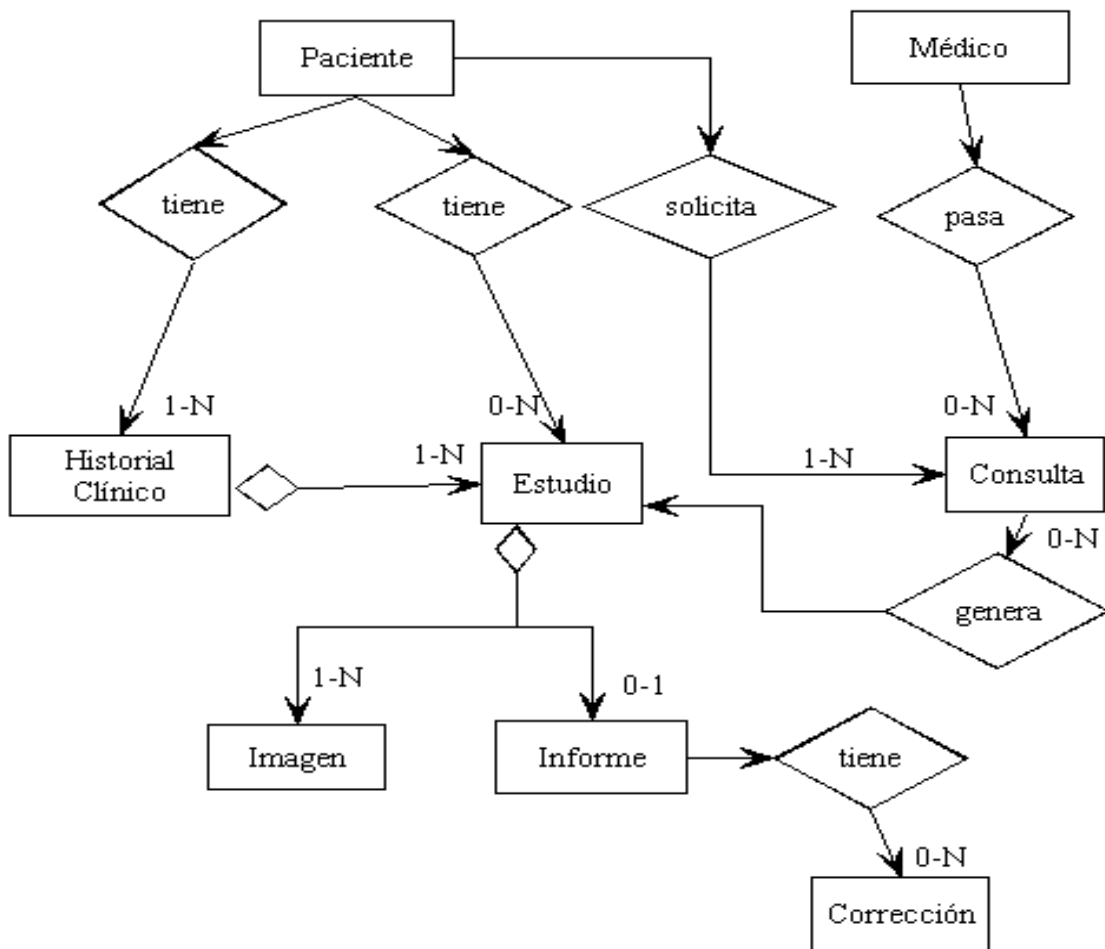


Fig 2: Diagrama Entidad-Relación.

vuelve al HMC se le volverá a asignar otro historial provisional, y tendremos que para un mismo paciente existen dos números de historia clínica, donde cada uno de ellos llevará asociadas las pruebas realizadas en cada una de las veces que el paciente fue atendido en el hospital.

Partiendo del modelo Entidad-Relación es posible aplicar una serie de reglas prácticas relativamente sencillas para obtener los esquemas de las tablas de una base de datos relacional. Aplicando estas reglas reducimos la mayoría de nuestras tablas a 3FN [4].

Una vez se tiene un estudio lógico del sistema y se ha llegado a un grado suficiente de normalización en las tablas (3FN, pe), hay que pensar si no habrá que realizar una desnormalización controlada para obtener un rendimiento mayor (diseño físico). No todas nuestras tablas son 3FN porque en el diseño de toda base de datos no sólo hay que pensar en el diseño lógico de la misma. Si bien el proceso de normalización es indispensable no es el único punto que hay que tener en cuenta. El diseño físico es igualmente importante y a menudo sus requisitos no son alcanzables al aplicar el proceso de normalización. Hay ciertos casos particulares en que un nivel de normalización mayor produce un menor rendimiento porque aumenta el número de consultas

en situaciones de búsqueda de datos combinados de varias tablas.

El rendimiento, la facilidad en las consultas y la integridad de los datos son puntos negros en ciertos diseños, situación que hay que controlar en todo momento.

Las tablas normalizadas suelen ser pequeñas tablas donde los datos son muy específicos de la entidad a la que representan. Esto genera un gran número de tablas y de claves externas para representar las relaciones entre los distintos datos. Si en nuestro sistema la información que más a menudo se solicita proviene de varias tablas, tendremos que realizar un gran número de operaciones join para obtener lo que buscamos. En un caso como éste es conveniente no normalizar tanto las tablas y permitir cierta redundancia en los datos para mejorar la velocidad de acceso a los mismos. Si bien esto conlleva el trabajo de tener que crear triggers [5] que controlen la consistencia de estos datos redundantes en cualquier modificación de los mismos.

La organización física de los datos entre diferentes dispositivos también influye en el tipo de rendimiento que obtendremos. Por ejemplo, la partición horizontal o vertical de las tablas es un

método de reducir tiempo de proceso en las búsquedas. Aunque como se está comentando toda mejora en una de las características de la base de datos: Rendimiento, Integridad de los datos, Espacio utilizado por los mismos, etc. va a suponer un empeoramiento en otra. Hay que considerar cada caso por separado para llegar al mejor diseño, tanto físico como lógico de una base de datos en particular.

3. Resultados

En este momento hemos probado el sistema en un entorno simulado. No ha sido probado aún en el hospital en condiciones de trabajo normal, sólo en nuestro laboratorio.

El prototipo ha demostrado que el sistema funciona. Hemos sido capaces de consultar a la base de datos usando las ASPs, y mucho más importante, hemos probado a almacenar y recuperar imágenes generadas por los estudios de los pacientes. El diseño de la base de datos es efectivo, permite que las consultas se ejecuten con rapidez, y los triggers diseñados para comprobar los datos antes de ser almacenados hacen su trabajo correctamente.

Nuestro sistema también permite el tratamiento de las imágenes de radiología antes de almacenarlas. Integra todas las tareas que los radiólogos realizan en de forma habitual, obtención de las imágenes, manipulación de las mismas para obtener la mejor visión del problema, almacenamiento y recuperación, además de la creación y consulta de informes y correcciones.

Hemos desarrollado un sistema que permite al radiólogo hacer todo su trabajo desde un ordenador, todo integrado con una interfaz gráfica para hacer la aplicación más atractiva y fácil de usar.

Gracias a las diferentes elecciones que hicimos durante la realización del sistema hemos conseguido ofrecer acceso a la información de la sección de radiología a cualquier usuario autenticado con conexión a Internet, independientemente del sistema operativo que use y del lugar en el que se encuentre.

Hemos desarrollado un servidor web capaz de entender y ejecutar código de una forma eficaz y segura. Hemos comprobado que las ASPs tienen ciertas características que hacen que esta solución sea muy eficaz

- La ejecución es más rápida que con CGIs
- La mayoría de las características de las aplicaciones web creadas con ASPs están basadas en componentes reutilizables en otras aplicaciones.
- ASP ofrece un modo simple de acceder a variables del servidor como el tipo de servidor,

modelo y versión del navegador, página desde la que se referencia, etc.

- Por medio de ADO se ofrece un mecanismo simple de acceso a cualquier fuente de datos compatible ODBC.

Gracias a esta última característica hemos mantenido independencia del tipo de gestor de base de datos que ha de utilizarse con nuestro código de acceso a base de datos, cualquier base de datos con conectividad ODBC (Informix, Oracle, Sybase, etc.) será válida para almacenar datos y acceder a ellos a través de la conexión establecida por nuestras páginas ASP.

4. Conclusiones

Los continuos avances en el campo de las aplicaciones de Internet han hecho posible que hoy en día podamos pensar en desarrollar sistemas complejos basándonos en el uso de Internet.

Con el uso de estos avances podemos crear diferentes aplicaciones que muestran al usuario el mismo tipo de interfaz, el navegador. Esto facilitará al usuario el uso de cualquier aplicación nueva. Con la facilidad del acceso a bases de datos se abre la posibilidad de desarrollar almacenes de información accesibles desde cualquier lugar a los usuarios autorizados. Se flexibiliza el acceso a la información requerida. Con el uso de tecnologías como ASP se incrementa la funcionalidad de los servidores, permitiéndoles llevar a cabo tareas que antes no podían ser programadas en estos servidores. Vemos un campo amplio de posibilidades para este tipo de aplicaciones.

Aún cuando la tecnología nos ofrezca todo lo necesario para desarrollar la mejor de las aplicaciones de telemedicina una cuestión que es básica para que la aplicación sea aceptada es la cooperación en el desarrollo de la misma de los usuarios finales. Son ellos los que tienen que ir orientando al ingeniero sobre la forma de trabajar que tienen, que es lo que más valorarían en un sistema y que es menos necesario. Sin esta colaboración podemos tener un buen producto que no sea aceptado porque no se adapta a la forma de trabajo a la que están habituados los que deberían ser sus usuarios. Un trabajo cooperativo entre desarrolladores y usuarios es un factor clave para el éxito del sistema. Por este motivo nuestro trabajo se basó en las especificaciones y comentarios del grupo de médicos que iban a probar el resultado. Hemos intentado imitar en lo posible su forma actual de trabajo para que la aplicación no produzca un cambio en su forma de hacer las cosas.

Agradecimientos

Los autores quieren agradecer al HMC su colaboración, y es especial a J.A. Arranz, J.L. de Francisco y J.A. Guerro.

Este trabajo está siendo financiado por el proyecto TEL97-0750 de la Comisión Interministerial de Ciencia y Tecnología.

Referencias.

[1] I. Soto et al, "*Analysis and Design of Telecommunication Networks for Applications in Telemedicine. Proceedings of the 1st World Congress of the Internet in Medicine*", Brighton, UK, November, 1996.

[2] Johnson, S., "*Using Active Server Pages*", Que Corporation (ed), 1997.

[3] Darnell R., "*HTML*", Sans.net Publishing, 1997.

[4] J.A. Cerrada, M.Collado, "*Técnicas Generales de Diseño Software*" en "*Introducción a la Ingeniería del Software*", Universidad Nacional de Educación a Distancia (ed), España, 1995. ISBN: 84-362-3161-9

[5] D. Solomon, R. Rankins, et al., "*Microsoft SQL Server 6.5*", pp:236-264, Sams Publishing, 1996. ISBN: 0-672-30956-4

Sistema de videoconferencia aplicado en telepatología

N. Pérez, M. de Cabo, Y. A. Dimitriadis y E. Fernández
Dpto. de Teoría de la Señal y Comunicaciones e Ingeniería Telemática
Camino del Cementerio, s/n, 47011 Valladolid
Telf: 983-423666, Fax: 983-423667
E-mail: marcab@gbien.rest.tel.uva.es; yannis@tel.uva.es; eusfer@tel.uva.es

Abstract

An application based on the Netmeeting SDK is described here to offer CSCW services for telediagnosis in pathology. A user interface was designed to fulfil the requirements of the end users, due to the fact that the interface from Netmeeting did not satisfy them. The application used objects directly supplied from the Netmeeting SDK as well as classes defined by the developer. Some videoconference equipments in the market and transmission systems are reviewed, verifying that Netmeeting is the most suitable for our needs. The application was proved in several scenarios in order to compare delays caused by the transmission systems. Finally, pathologists from the HMC (Medina del Campo Hospital) accepted the functionality of the interface. Nevertheless an appropriate election of transmission and computer resources will allow an improvement of quality displaying pathologist images.

1. Introducción

En la emergente Sociedad de Información es cada vez más importante el desarrollo de servicios y aplicaciones telemáticas. Estos servicios pretenden acercar la telemática a la vida cotidiana de los ciudadanos, tal y como se destaca en repetidas ocasiones por las autoridades de la Unión Europea en el diseño de los Programas Marco de I+D [1].

La telemedicina incide directamente en el sistema sanitario y pretende acercar la sanidad al paciente, haciendo que el vivir en lugares remotos o en ambientes rurales no sea un impedimento para tener una sanidad rápida y de alta calidad [2]. El trabajo en este campo data de los años setenta, pero sólo últimamente se ha podido llegar a soluciones más completas debido a los avances telemáticos. Sin embargo, su propósito de modificar múltiples aspectos, tanto humanos (pacientes, médicos) como organizativos (estructura del sistema sanitario), conlleva muchos problemas [3]. Por ello, consideramos que los esfuerzos en servicios de telemedicina deben estar guiados por la adaptación a condiciones reales, así como por su integración en un sistema global.

La colaboración entre los médicos especialistas o de atención primaria, situados en distintos lugares dentro del mismo centro o zona sanitaria, constituye un elemento básico de las aplicaciones de telemedicina. Su necesidad se hace patente con la tendencia actual en España de descentralización de los servicios sanitarios y el funcionamiento de los CAP (Centros de Atención Primaria) y CS (Centros de Salud), así como la existencia de hospitales comarcales, provinciales o autonómicos.

La videoconferencia, en sus múltiples variantes, es un medio de comunicación mediante el soporte de

las redes de ordenadores y permite la creación de un espacio virtual entre los participantes en ella. Este

tipo de aplicaciones se enmarca dentro del campo de CSCW (*Computer Supported Collaborative Work* o Trabajo Colaborativo Asistido por Ordenador), que a partir del año 1984 intenta integrar diversas disciplinas tales como Redes de Ordenadores, Ingeniería de Software, Psicología de Dinámica de Grupos, etc. [4]. Su carácter multidisciplinar y su impacto a las organizaciones o las relaciones sociales le dan cierto dinamismo pero al mismo tiempo introducen muchos obstáculos para su éxito final [5]. Hay que destacar que debido a la complejidad de las aplicaciones CSCW o *groupware*, se han propuesto varios *toolkits*, APIs (*Application Programming Interface*) o entornos de desarrollo software que facilitan la creación de aplicaciones.

En este artículo presentamos una aplicación de trabajo colaborativo basado en videoconferencia para ofertar servicios de telepatología. Esta aplicación está integrada dentro de un sistema global en el contexto de un proyecto I+D de la CICYT. La necesidad de una implantación real y la exigencia de aprobación de este sistema por parte de los usuarios finales (médicos, administradores y pacientes) ha guiado su desarrollo. La aplicación se centra en la colaboración entre el médico general de un CS o CAP con el médico especialista de anatomía patológica del HMC (Hospital comarcal de Medina de Campo, Valladolid), con el fin de compartir datos, imágenes de tejidos o simplemente opiniones. El objetivo último es comprobar la utilidad y los condicionantes de un sistema CSCW en un entorno real.

Primeramente, se hace una breve introducción a la telemedicina, explicando sus principales aportaciones y problemas. Al mismo tiempo, se introduce a los conceptos básicos y principales entornos de desarrollo de aplicaciones de CSCW.

En la siguiente sección se presenta el problema concreto situándole dentro del marco general, junto con el análisis de requisitos. A continuación, se expone la problemática de selección de equipos de

videoconferencia y del medio de transmisión, justificando la topología final del sistema. En las próximas secciones se analiza el interfaz desarrollado mediante el Netmeeting SDK (*Software Development Kit*) y se analizan las pruebas de campo realizadas, finalizando con las conclusiones y las futuras líneas de investigación.

2. Videoconferencia en telemedicina

En esta sección vamos a introducir conceptos básicos y taxonomías que nos permitirán entender mejor los problemas y perspectivas de la videoconferencia o más ampliamente del CSCW en telemedicina.

La telemedicina se puede entender como la provisión del servicio médico-sanitario mediante una combinación de tecnologías de telecomunicaciones y multimedia con los adecuados conocimientos médicos [2]. Desde el punto de vista tecnológico, mediante aplicaciones telemáticas se pueden conseguir varios objetivos tales como [3, 6]:

- Facilitar la administración y la gestión de los hospitales.
- Proporcionar información a los propios pacientes sobre enfermedades, síntomas etc.
- Proporcionar una educación médica para los profesionales en el ambiente rural.
- Facilitar bases de datos de fácil acceso pero de forma confidencial, para la compartición de la información.
- Facilitar la comunicación entre médicos para compartir experiencias y estudios. Especialmente permitir videoconferencias entre especialistas para que ellos puedan compartir historiales e imágenes de diagnóstico.
- Poder dar servicios de emergencia en el ámbito rural, mediante la evaluación y diagnóstico por parte de especialistas. Igualmente, permitir el telecontrol de lentes o microscopios.

Sin embargo, su éxito va encontrando múltiples dificultades referidas a la infraestructura (especialmente de banda ancha), estandarización, la seguridad y privacidad de los datos de pacientes, así como la dificultad de adaptación de las actitudes de médicos y pacientes.

La videoconferencia se ha ido utilizando en múltiples aplicaciones de telemedicina, entre ellas para facilitar la gestión sanitaria, para la formación médica continuada y finalmente para la teleasistencia, en sus variantes de teleconsulta y telediagnóstico. El telediagnóstico es especialmente interesante, ya que nuestra aplicación se refiere a telepatología, es decir el análisis de los tejidos de pacientes a distancia [7].

Desde el punto de vista tecnológico, la videoconferencia se puede considerar como una conferencia en tiempo real, que sigue un cierto

protocolo de sincronización y de participación de los usuarios [8]. Aunque la principal fuente de información es el video, podemos considerar también la necesidad de incluir datos (conversación escrita o compartición de datos) y por supuesto el audio. Todo eso se rige principalmente por los estándares T.120 y H.320 de conferencias multipunto multimedia.

De la exposición anterior, se desprende que la videoconferencia es un medio que debe estar integrado con otras técnicas de trabajo colaborativo con el objetivo de crear un espacio común entre sus participantes. En la siguiente sección trataremos de situar la aplicación a desarrollar dentro del mundo de aplicaciones colaborativas, según los criterios más ampliamente reconocidos. De forma adicional vamos a discutir la elección del *toolkit* más apropiado para el desarrollo del sistema [9].

3. Definición y análisis de requisitos del sistema

El sistema planteado se podría desglosar en las siguientes actividades:

- Consultas interactivas entre el HMC y los CAPs o CSs dependientes del Hospital, para discutir sobre opiniones diagnósticas y terapéuticas, para los pacientes remitidos desde los centros periféricos.
- Diagnósticos conjuntos con otros hospitales, donde pueden estar situados otros especialistas de mayor experiencia.
- Biopsias necesarias durante una operación, conectando de esta forma el quirófano con anatomía patológica.

Las variantes anteriores se citan en orden decreciente de importancia y cubren conexiones inter o intra-hospitalarios dentro de la misma zona sanitaria o no.

El sistema se enmarca claramente dentro de los servicios de teleasistencia, y más concretamente de teleconsulta y telediagnóstico. De esta forma pacientes, médicos generales y especialistas se acercan, evitando desplazamientos y aumentando la calidad de servicios prestados.

Después de múltiples consultas con los facultativos y gerentes del HMC se llegaron a definir las siguientes funciones referidas al patólogo, que se deben ofrecer por nuestro sistema:

- Ver y hablar con otros médicos y/o pacientes. Así se puede autenticar a los participantes y realizar funciones típicas de consultas cara-a-cara.
- Compartir con los otros médicos una imagen digitalizada y previamente grabada en su PC, típicamente referida a tejidos.

- Compartir expedientes de los enfermos o videos pregrabados de la salida de microscopio.
- Enviar cualquier fichero a otros médicos para su análisis posterior o para minimizar retardos importantes en una compartición en tiempo real.
- Mandar mensajes de correo electrónico, en caso de ausencia del otro médico, o para transmitir dudas u otras peticiones que se vayan a discutir en las sesiones síncronas.
- Trabajar sobre imágenes en movimiento extraídas del microscopio con el fin de concretar cuál es la parte más interesante de la muestra.
- Mantener una conversación escrita para que ella quede registrada, así como reducir el ancho de banda requerido.

Dentro de la clasificación de los sistemas CSCW [4] podemos definir nuestro caso, como un sistema de conferencia por ordenador (síncrono), que incluye elementos de mensajería (asíncrono) entre usuarios localizados en lugares remotos entre sí.

El sistema deberá implantarse en un entorno concreto que plantea unos requisitos particulares. Por un lado, el carácter dominante del entorno Windows condiciona la elección de plataforma de *software*. Por otro lado, las características tecnológicas de la zona sanitaria concreta (ambiente rural) nos llevan a tomar decisiones particulares sobre la topología, elementos y medio de transmisión de la *red* (ver sección 4).

Además de los requisitos generales de teleasistencia, aquí deberíamos destacar un requisito esencial de *usuario* es la calidad con la que se capturan, transmiten y visualizan las imágenes. Dado que el diagnóstico depende de estas imágenes, su calidad debe ser óptima, cercana a la obtenida directamente del microscopio o por lo menos con una resolución de 768x576. La gestión de *comunicaciones* (eventos entrantes y salientes de la comunicación síncrona) se suministra por la plataforma de desarrollo de software que describiremos más adelante. Los periféricos relativos al *interface* incluyen teclado, ratón, cámara, micrófono, así como un sistema de digitalización de imágenes microscópicas. La funcionalidad a nivel de *presentación* requiere agilidad en la colaboración, los médicos deben poder señalar la zona de interés en la imagen, observar imágenes o secuencias de imágenes elegidas, o hasta captar y mostrar nuevas imágenes elegidas. El sistema también debe ofrecer una comunicación fácil con los historiales [16] y permitir actividades privadas de cada participante. Como último requisito pero quizás el más importante dejamos el referido al coste económico, que debe ser reducido y siempre dentro de las posibilidades reales tanto del proyecto I+D como del sistema sanitario en general.

4. Análisis de las tecnologías disponibles

4.1. Equipos y medios de transmisión

Para la elección de los equipos y medios de transmisión, que se van a instalar en los distintos centros de salud, se han tenido en cuenta tanto sus condiciones tecnológicas como su localización.

Debido a la localización de los CAPs con los que se va a trabajar, la RDSI no llega a todos esos centros por lo costoso que supone su implantación. Entonces, esta es una restricción muy a tener en cuenta en la elección del sistema de videoconferencia (equipos de sala, equipos compactos, equipos para PC y videoteléfonos son diferentes tipos de segmentos). No se ha optado por ninguno de esos sistemas de videoconferencia, debido a que la gran mayoría ellos están basados en la utilización de la RDSI.

En cambio, se pensó en la utilización de herramientas para trabajo colaborativo, que se comentan a continuación, dado que pueden estar soportadas por diferentes medios de transmisión (p.e. RTC, RDSI, Ethernet). Se ha pensado en el empleo de RTC (aunque esta red está pensada para la transmisión de voz e información analógica) como medio de transmisión para el acceso del médico del CAP al hospital, ya que permite aprovechar la red de comunicaciones más extendida y de menor coste.

Existen otros medios de transmisión, aunque no se estudió su utilización, debido a que las infraestructuras no llegan a los sitios involucrados en este sistema, así como el aumento significativo de coste (IBERPAC, líneas dedicadas y *Frame Relay*).

4.2 Plataforma *software* para CSCW

Uno de los aspectos más importantes para la realización del sistema de trabajo colaborativo es la elección de la plataforma de software. Inicialmente se descartó la opción del desarrollo mediante solamente un lenguaje de programación por la necesidad de tener rápidamente prototipos para su experimentación. Se optó por el uso de uno de los *toolkits* para trabajo colaborativo síncrono, que típicamente gestionan el establecimiento y finalización de una conferencia, mantienen la sincronización y la consistencia en la visualización del espacio común, gestionan los usuarios válidos, etc. Entre otros *toolkits* se consideraron los siguientes:

- Groupkit [10], desarrollado en C++ y Tcl/Tk por la Universidad de Toronto y válido para casi todas las plataformas, es uno de los primeros y más utilizados en entornos universitarios.
- JSDT (*Java Shared Data Toolkit*) [11], producto de Sun que ofrece todas las ventajas bien conocidas de sistemas basados en Java.
- Microsoft Netmeeting SDK (*Software Development Kit*) [12], que permite integrar

aplicaciones en Visual C++ y proporciona controles ActiveX en relación con páginas Web.

La elección de la plataforma de desarrollo se basó en nuestra experiencia anterior en el desarrollo de sistemas CSCW y los requisitos particulares de esta aplicación [13]. Aunque en otros sistemas optamos por el uso de JSDT debido a su independencia de sistemas operativos, hardware y protocolos subyacentes [14] o por el Groupkit por su flexibilidad y carácter abierto [15], aquí decidimos adoptar al Netmeeting SDK. Esa decisión se guió por los requisitos de plataforma de software (Microsoft Windows), la familiarización de los médicos con este entorno y su evidente estandarización en el mercado. Sin embargo, hay que destacar que este entorno es todavía bastante inestable, poco abierto y por supuesto propietario, pero permite la integración y personalización que ya comentamos en los requisitos de usuario. Podemos destacar que Netmeeting se puede caracterizar como una arquitectura distribuida, con interface WYSIWIS (*What You See Is What I See*), sin compartición a nivel de vista u objeto, y finalmente la colaboración es transparente, es decir que las aplicaciones monousuario existentes simplemente se extienden y se comparten.

4.3 Topología del sistema

Debido a los problemas, tanto de infraestructura, como de precio y calidad ofrecida, se decidió utilizar la red local como método de acceso a anatomía patológica, mientras que el acceso del médico del CAP (Centro de Atención Primaria) es mediante RTC.

En la Fig. 1 se muestra el sistema de acceso implantado. Desde un CAP se accede al servidor RAS mediante línea telefónica. El CAP hace una llamada al servidor RAS, que dispone de una serie de direcciones de red, que adjudica al equipo remoto, al tiempo que le da de alta en el dominio al que pertenece.

El servidor RAS también podrá discriminar las entradas. Entonces, realiza un proceso de verificación de la llamada entrante, para comprobar si esa llamada tiene acceso a la red de área local del HMC. Una vez aceptada la llamada, se tendrá una conexión punto a punto desde el CAP al RAS por RTC. De esta manera, la información que envía el CAP va por dos medios diferentes; por un lado, desde el CAP hasta el RAS por RTC, y por otro lado, desde el RAS hasta Patología por la RAL.

Los datos transmitidos por RTC son encapsulados según el protocolo PPP, a partir de los datos encapsulados en paquetes UDP por el equipo de videoconferencia. Los paquetes son luego desencapsulados al llegar al RAS, transmitiéndose por la RAL en forma de paquetes UDP. Desde Patología al CAP se realizaría el proceso de forma

similar, pero recorriendo el sistema en sentido contrario.

La topología de la RAL del HMC será en estrella. Todos los puntos estarán conectados gracias a un *hub* Fast Ethernet. A este *hub* estarán conectados todos los servidores y ordenadores personales de los departamentos del hospital, entre los que se encuentra Patología. Los CAP no tendrán conexión directa a estos ordenadores que comparten esta red, sino que primero tendrán que pasar por el servidor RAS.

5. Diseño de la aplicación

El propósito de esta aplicación fue desarrollar una interfaz de usuario específica para el patólogo, en la que se recogieran de forma más intuitiva todas sus necesidades; una de ellas puede ser lanzar las aplicaciones de visualización de imágenes, a partir de la aplicación de videoconferencia. Este requisito no se cumple con la interfaz propia de NetMeeting, ya que sería necesario minimizar previamente NetMeeting, antes de lanzar la aplicación de forma normal.

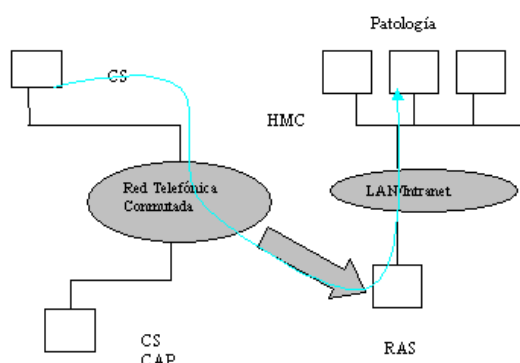


Figura 1: Topología de la red

5.1. Funcionalidades del interfaz

La funcionalidad del programa creado se resume en los siguientes puntos:

- Realización de una llamada. Permite realizar una llamada a varios usuarios o unirse a una conferencia ya establecida. Esto permitirá "charlar" con varios médicos al mismo tiempo sobre un tema de interés común.
- Recepción de una llamada. Siempre que tengamos el programa lanzado podremos recibir cualquier llamada entrante recibida. De esta forma si alguien quiere comunicarse con el patólogo éste estará fácilmente localizable.
- Conversación. Permite el envío y la recepción de mensajes entre los participantes en la conferencia. Estos mensajes pueden ser almacenados, permitiendo posteriormente la reproducción de la conversación. Esta opción es útil para la educación ya que permite a los alumnos ver los comentarios intercambiados por dos especialistas hasta llegar a la emisión de un diagnóstico.

- Transferencia de archivos. Permite el envío de archivos directamente desde la aplicación. De esta forma los participantes pueden mantener localmente copias del material que se está comentando en la sesión de videoconferencia
- Envío de correo. Esta opción ejecuta automáticamente el cliente de correo electrónico instalado.
- Rechazo de un usuario. El iniciador de la conferencia puede admitir o rechazar a los usuarios que quieran unirse a la misma. Se permite así un control sobre quién puede acceder a la información que se está intercambiando en la conferencia.
- Conmutación de la comunicación entre conferenciantes. La aplicación sólo permite recibir sonido y vídeo de un interlocutor, aun cuando en la conferencia existan varios. Esta opción permite pasar de un usuario a otro para establecer la comunicación. Digamos que se pasa el turno de exposición de un conferenciante a otro.
- Captura de una imagen. Esta opción permite capturar una imagen desde el microscopio electrónico al mismo tiempo que se está realizando la videoconferencia. Para ello es necesario tener dos tarjetas de vídeo, una usada para la videoconferencia y otra para la captura de las imágenes microscópicas. En nuestro caso, las tarjetas son C-210 y AV-MASTER respectivamente [13].
- Compartición de aplicaciones. Es muy común la necesidad de ver un programa que el interlocutor está manejando. Desde la interfaz de usuario creada se pueden compartir directamente tres tipos de aplicaciones muy comunes en el entorno de uso del mismo:

1. Microsoft Photo Editor. Esta aplicación es usada para que los médicos compartan una imagen capturada que quieren comentar. Se permite manipular la imagen (cambiar el tamaño, cortarla, etc.) La persona que comparte la aplicación ha de tenerla instalada en su ordenador personal, pero la que disfruta de la compartición no, toda la información que necesita para visualizar la imagen le es transmitida por el canal de videoconferencia.
2. Reproductor multimedia. Su objetivo es la visualización de una secuencia de vídeo pregrabado. Se admiten formato AVI y MOV. Para este último es necesario instalar el controlador QuickTime.
3. Navegador. El navegador es necesario para acceder a la base de datos del hospital y examinar el historial clínico del paciente que se está tratando. Se podrán obtener otras imágenes relacionadas con este paciente u otro tipo de datos que sean necesarios en un momento dado[16].

Estas aplicaciones son las que se comparten directamente desde la interfaz, pero cualquier otra aplicación podrá ser compartida de forma menos intuitiva, de forma similar a cómo se hace con la interfaz propia de NetMeeting.

5.2. Realización del interfaz

Para la realización de nuestra interfaz personal recurrimos al SDK (Software Development Kit) ofrecido por NetMeeting y a la programación mediante Visual C++ [17].

Para ello se ha utilizado la biblioteca de clases COM (Component Object Model). COM es un protocolo que conecta dos módulos software y luego desaparece de escena. Una vez establecida la comunicación los dos módulos podrán comunicarse mediante un mecanismo denominado interfaz. Las interfaces no requieren de puntos de entrada enlazados dinámica o estáticamente, sólo necesitan unas funciones COM de propósito general que inician el proceso de comunicación.

Comentaremos alguno de estos objetos COM que se han utilizado [12]:

Conference Manager: Encargado de todo el sistema de conferencia.

Local System : Ofrece toda la información sobre el usuario local.

Call: Monitoriza y controla las llamadas entrantes o salientes.

Conference: Controla una instancia de una conferencia. Dentro de una aplicación, y con un solo manager puede haber más de un objeto conference.

Member: Representa a un participante de la conferencia.

Channel: Canal para un determinado tipo de datos. Se utiliza tanto para vídeo, audio o datos, como para compartir aplicaciones y hacer transferencia de ficheros. Para cada tipo de datos tenemos un canal específico (*Audio Channel*, *Video Channel*, *File Transfer Channel*, *Application Sharing Channel*, *Data Channel*).

Dentro de una conferencia existen objetos que son imprescindibles para la misma, como son *Conference Manager* y *Conference*. Otros sólo son necesarios en ciertas aplicaciones, pero no en todas. Todos estos objetos ofrecen una interfaz para acceder a sus métodos y poder así crear y gestionar nuestra aplicación. Nos permiten realizar un control de la misma.

Existen otros objetos de la aplicación NetMeeting, a parte de los definidos por COM, que son definidos por el programador y que también se han incluido en nuestro programa. En la Fig. 2 se presenta un diagrama de los mismos, junto con sus relaciones. Se describen brevemente a continuación cada una de las clases de la Fig. 2.

CMgrNotify, *CCallotify*, *ConfNotify*, *CvideoNotify*, son clases que se utilizan básicamente para informar al propio y al usuario de lo que está ocurriendo en el sistema. Heredan las notificaciones de los objetos COM anteriormente expuestos. Son los receptores de notificaciones de eventos que suceden durante la conferencia. También heredan los métodos de las clases *RefCount* y *Cnotify*. De esta última heredan

los métodos Connect y Disconnect. Cnotify es quien gestiona en realidad la conexión hacia una determinada interfaz, que puede ser la barra de estado, la de herramienta, una caja de diálogo, etc. De RefCount utilizan los métodos AddRef y Release, para incrementar o destruir una referencia. El uso de estas referencias es debido a que las interfaces COM no tienen destructores virtuales. Cada clase COM tiene un miembro de datos que controla cuántos usuarios tiene un objeto. Cada vez que se crea una nueva instancia del interfaz se llama a AddRef y cuando se finaliza el uso se llama a Release. Cuando el dato toma el valor 0 el objeto se autodestruye.

TagUser. Especifica las características de un miembro de la conferencia, nombre, dirección, etc.

PREF. Define las preferencias del usuario.

TagCol. Define la anchura de todas las columnas que aparecen en la aplicación, así como de su título.

BSTRING y *BTSTR*. Se utilizan porque sus métodos proporcionan una traducción de los tipos de datos.

CMD y *MSD*. Se utilizan para enviar comandos o mensajes a los handlers adecuados.

COBNODE y *COBLIST*. Sirven para la gestión de listas. Coblist tiene métodos que permiten saber quién es la cabeza o el final de la lista, y métodos para añadir, borrar, comparar u obtener la posición de un elemento. Cobnode es un nodo de dicha lista y tiene un elemento que identifica su posición y otro para saber cuál es el siguiente elemento de la lista.

Así la realización de nuestro proyecto se hizo con dos tipos de objetos. El primero proporcionado directamente por el SDK de NetMeeting, accediendo a sus interfaces, y el segundo tipo se consiguió creando nuevas clases que heredan de las clases SDK.

6. Pruebas de campo realizadas

Se ha optado en este caso realizar las pruebas de campo primeramente en RAL (Red de Área Local), que presenta una velocidad de transmisión mayor que la presentada por otros medios de transmisión, como pueden ser modems y RTC. Esto permite comprobar cómo el empleo de modems degrada la calidad del servicio. Estas dos pruebas se realizaron

antes de la prueba en el HMC, debido a que esta última supone una complejidad mayor.

Se han realizado también medidas de *throughput* para la transmisión de vídeo, sonido, e imágenes fijas, empleando los distintos medios de transmisión. Se ha confirmado cómo este valor es superior para las pruebas en RAL.

6.1. Pruebas en LAN

Respecto a la calidad de la señal de vídeo en ningún momento es extraordinaria, sin embargo creemos que sí sirve para nuestro propósito la identificación de la otra persona.

La señal de sonido llega al otro extremo con un pequeño retardo, aunque con buena calidad. En algunas pruebas se producían cortes de señal, posiblemente provocados en los momentos de gran utilización de la red.

6.2. Pruebas con módem

En este último caso el medio va a ser algo diferente al utilizado en el hospital; en vez de acceder a RTC, como aparece en la Fig. 1, se utiliza la red Ibercom en modo local sin acceso a RTC. La velocidad de los modems en ambos extremos es a 33600 bps.

En cuanto a la calidad, no se pudo apreciar prácticamente diferencia, respecto a la que ofrece la LAN; tanto el vídeo, como el sonido y la compartición de aplicaciones ofrecen una calidad semejante.

6.3. Pruebas en el HMC

La aplicación se probó en el HMC con usuarios reales, en un entorno más semejante al verdadero. Para ello, se montó una subred con su propio servidor de dominio, servidor DNS y servidor de acceso telefónico a redes.

Un ordenador implementaba todo lo necesario para el trabajo del patólogo, mientras que el ordenador del otro extremo representaba la funcionalidad para el médico del CAP, y accedía esta vez por línea telefónica. Este último ordenador también estaba dentro del hospital, no en el CAP que sería su

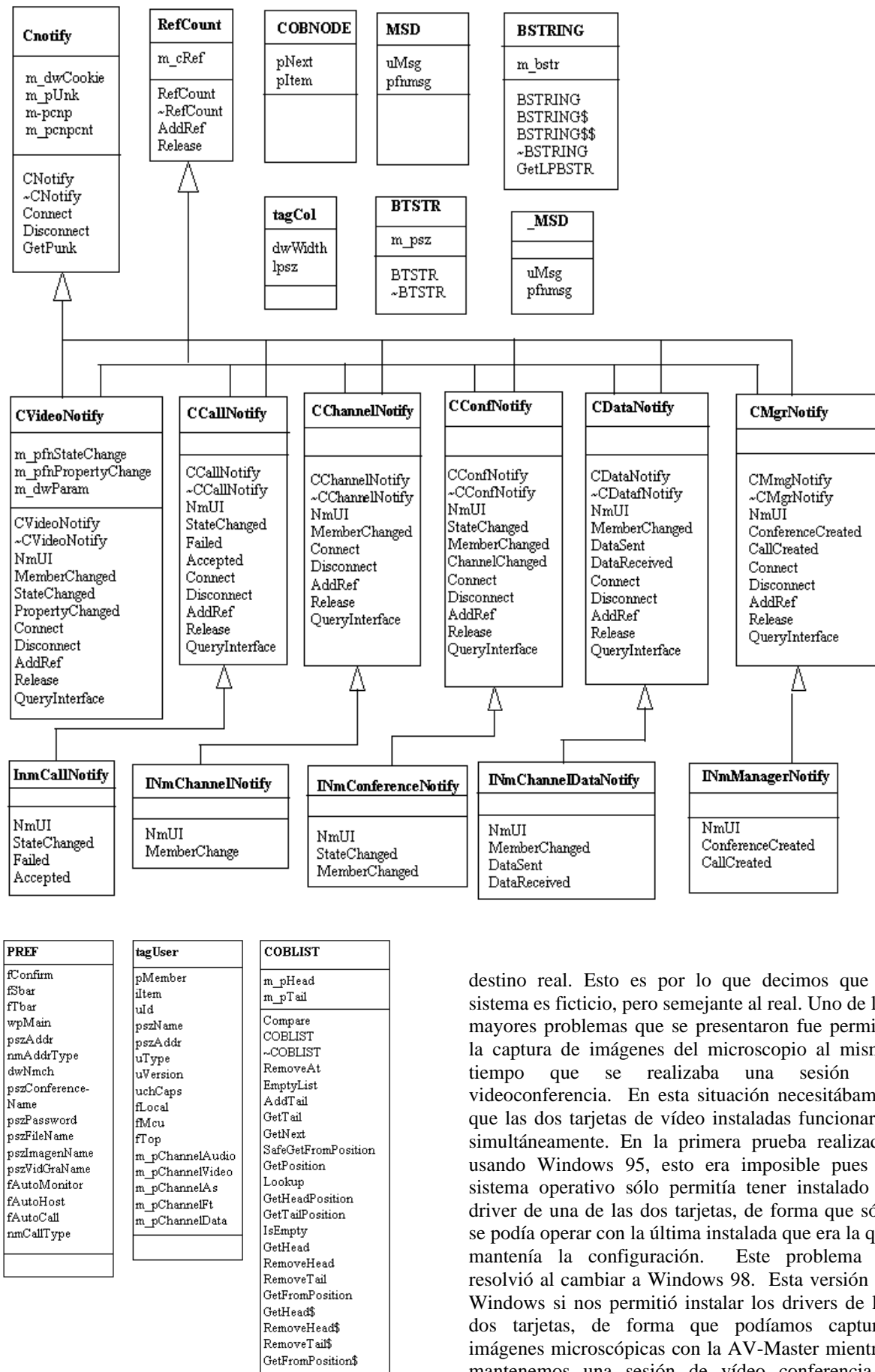


Figura 2: Objetos diseñados por el programador

destino real. Esto es por lo que decimos que el sistema es ficticio, pero semejante al real. Uno de los mayores problemas que se presentaron fue permitir la captura de imágenes del microscopio al mismo tiempo que se realizaba una sesión de videoconferencia. En esta situación necesitábamos que las dos tarjetas de vídeo instaladas funcionaran simultáneamente. En la primera prueba realizada, usando Windows 95, esto era imposible pues el sistema operativo sólo permitía tener instalado el driver de una de las dos tarjetas, de forma que sólo se podía operar con la última instalada que era la que mantenía la configuración. Este problema se resolvió al cambiar a Windows 98. Esta versión de Windows si nos permitió instalar los drivers de las dos tarjetas, de forma que podíamos capturar imágenes microscópicas con la AV-Master mientras mantenemos una sesión de vídeo conferencia a través de la C-210.

Se comprobó que la calidad de vídeo apenas tenía diferencias, respecto a las pruebas anteriores, aunque la calidad de sonido era algo peor, ya que la señal se entrecortaba con mayor frecuencia.

La compartición de imágenes fue mucho más lenta, ya que aparece la imagen en el otro extremo con mayor retardo. La calidad de la imagen no fue la óptima para el usuario final, ya que los recursos informáticos tampoco fueron los adecuados.

7. Conclusiones y líneas futuras

Como conclusión se puede resaltar las mejoras que todavía quedan por realizar. Estas mejoras se pueden llevar a efecto, optando por cualquiera de las siguientes soluciones:

1. Mejora de la calidad de los recursos informáticos, en cuanto a configuración, capacidad de procesamiento y tarjetas de interfaz.
2. Utilización un medio de transmisión mejor adaptado a la videoconferencia. Esto depende fundamentalmente de que el nuevo medio de transmisión pueda llegar a todos los puntos en los que haya que realizar pruebas de campo.
3. Creación de una herramienta de videoconferencia desde el principio. Esto permitirá evitar los inconvenientes del empleo de Netmeeting. Entre otros resaltar la aparición de *bugs* propios de ese entorno.

Esta última línea supone no sólo crear un entorno colaborativo, sino también el tratamiento de señales de sonido y vídeo. El tratamiento de estos datos es diferente, ya que la aparición de retardos o pérdidas las harían ininteligibles.

Agradecimientos

Los autores quieren agradecer al HMC su colaboración, y es especial a J.A. Arranz, J.L. de Francisco y C. Perna. Este trabajo está siendo financiado por el proyecto TEL97-0750 de la Comisión Interministerial de Ciencia y Tecnología.

Referencias

- [1] Information Society Technologies, 1999, Workprogramme, Draft 2.1, Dec. 22, 1998. <http://www.cordis.lu/ist>
- [2] J.E. Cabral, J.H. Kim. "Multimedia Systems for Telemedicine and their Communications Requirements". IEEE Communications Magazine, Vol. 34, No.1, Julio 1996.
- [3] J.L. Ritter, A.D. Little. "Telecare as Vehicle on the Information Superhighway". Proceedings of the 1995 annual HIMSS Conference, San Antonio, Texas, 12-16 de Febrero 1995, ISBN: 1048-8987.
- [4] C.A. Ellis, S.I. Gibbs, G.L. Rein. "Groupware: some issues and experiences", Communications of the ACM, vol. 34 no 1 pp 38-58, Enero 1991.
- [5] J. Grudin. "Groupware and Cooperative Work: Problems and Prospects" en "The Art of Human

Computer Interface Design", Addison-Wesley, 1990.

- [6] P. Palomo. "Análisis y Diseño de Redes de Telecomunicaciones Aplicadas a Diversos Servicios Hospitalarios", Proyecto Fín de Carrera, ETSIT, Universidad de Valladolid, Marzo 1997.
- [7] S. Akselen et al. "Telemedicine and ISDN", IEEE Communications Magazine, vol, 31 no 1, pp 46-51 Enero 1993.
- [8] D. Coleman. "Groupware: Collaborative Strategies for Corporate LANs and Intranets", Prentice Hall, 1997. ISBN: 0-13-727728-8.
- [9] W. Reinhard, J. Schweitzer, G. Völsen. "CSCW Tools: Concepts and Architectures", IEEE Computer, vol. 27, no. 5, pp. 28-36, Mayo 1994.
- [10] Roseman, S. Greenberg. "Groupkit: A Groupware Toolkit for Building Real-Time Conferencing Applications". Proceedings of the ACM Conference on Computer Supported Cooperative Work. CSCW92, Toronto, Canadá Noviembre 1992.
- [11] Sun Microsystems, "Java Shared Data Toolkit User Guide", disponible en <http://java.sun.com/products/java-media/jsdt>.
- [12] Software Development Kit de NetMeeting. <http://www.Microsoft.com/netMeeting/sdk>
- [13] C. Osuna et al. "ATOIDI: Aplicación para el Aprendizaje y Desarrollo de Habilidades Sociales en un Entorno Distribuido". Actas del Simposio Español de Informática Distribuida. SEID99. Santiago de Compostela. 24-26 de Febrero 1999. ISBN: 84-8408-060-9.
- [14] J.I. Asensio et al. "Hierarchy and Competition in CSCW Applications: Model and Case Study". Journal of Decision Systems (en fase de impresión).
- [15] Noelia Pérez, "Análisis, diseño e implantación de un sistema de videoconferencia para aplicaciones en telemedicina". Proyecto Fin de Carrera, ETSIT, Universidad de Valladolid. Febrero, 1999.
- [16] C. Del Barrio, M. De Cabo, E. Fernández y G. García. "Accesing Radiology Images Database by Active X Scripting", Proceedings of the 3th Congress of the Internet in Medicine, London, UK, Noviembre 1998.
- [17] David. J. Kruglinski, "Programación avanzada con Visual C++", McGraw Hill (Ed.), 1995.

Generación de programas gráficos en X Window. Herramienta *xgui*

Carlos M. Ramírez-Casañas, José R. Ansó Magaña
Departamento de Ingeniería Telemática
Universidad de Las Palmas de Gran Canaria
Campus de Tafira s/n, Edificio de Electrónica y Telecomunicación,
35017 (Las Palmas de Gran Canaria)
Telf: 928-451224, Fax: 928-451243
E-mail: cramirez@cic.teleco.ulpgc.es

Abstract

In this article we present a software tool for programming Graphic User Interfaces on Unix environment, that we have called "xgui". This tool allows the programmer to make simple graphic applications quickly and, at the same time, allowing the user to gain new experience on Motif library functions. From the teacher's point of view, xgui is designed for beginning to apply their knowledge to graphic interface design. This software provides a set of options that any programmer can use with any graphic application. Furthermore, a professional appearance can be reached with a minimum cost. The tool's purpose is to solve problems for students who have to develop engineering projects without having much time available. In this article, we give a step by step guide on how to use this tool. At the end of this paper, the design and the modules of xgui are illustrated.

1. Introducción

En este artículo se presenta una herramienta denominada *xgui*. Ésta permite a los programadores en el entorno del sistema operativo UNIX, trabajar con el paquete de funciones *Motif*^[2][3] para el desarrollo de interfaces gráficas. Todo ello sin necesidad de poseer conocimientos previos de la programación con *Motif* [1].

Desde el punto de vista docente, para el desarrollo de prácticas académicas, trabajos de cursos o Proyectos Fin de Carrera, en los que se requieran interfaces con un aspecto amigable y profesional, los diseñadores de las aplicaciones tenían que emplear un tiempo considerable en el aprendizaje de la programación gráfica. Con esta herramienta se intenta minimizar este tiempo y permitir a los programadores centrarse en los objetivos propios de cada aplicación y no en su aspecto.

La herramienta *xgui* proporciona el código que permite crear los componentes gráficos (botones, etiquetas, formularios, etc.) que se utilizan con más frecuencia en la elaboración de una interfaz gráfica de usuario (GUI, *Graphic User Interface*)[4]. De esta forma, se facilita un generador de código que proporciona líneas para su inserción en el programa fuente con el fin de obtener una aplicación gráfica con aspecto profesional.

Con respecto a otras herramientas[6][7][8] que se encuentran en el mercado, podemos decir que *xgui* no pretende ser una herramienta que sustituya a las existentes en cuanto a la generación de GUIs. Incluso sirve de iniciación en la práctica de la programación gráfica. Tal es el caso de alumnos de

asignaturas propias de Telemática, a los que no se les oculta el código y pueden saber qué hace el programa en cada momento.

Podemos decir que *xgui* se concibió como una herramienta didáctica que, a medida que se utilizara, permitiese la asimilación de los conocimientos y la adquisición de la práctica necesaria para que, en último extremo, posibilitara la independencia de ésta. Con esto se intenta minimizar la inversión temporal que inicialmente hay que realizar en el estudio de funciones específicas para la realización de una aplicación gráfica sencilla.

En los próximos apartados se presenta con mayor detalle la herramienta *xgui*, su descripción y su utilización. A continuación se expone el diseño de la estructura de la aplicación. Por último se enumeran una serie de conclusiones.

2. La herramienta *xgui*

La herramienta *xgui* se basa en generar el código fuente necesario para la construcción de la interfaz gráfica de usuario, siguiendo los requerimientos del programador.

Primero debemos indicar que existen muchas clases de *widgets*^[5] en el juego de funciones *Motif*. Esta aplicación soporta hasta veinte de estos *widgets*, que han sido considerados como los más habituales en cualquier tipo de aplicación gráfica.

La aplicación proporciona una interfaz en la que el usuario tiene a la vista la mayor parte de las opciones y mecanismos de funcionamiento de la misma. Por otro lado, *xgui* posee un funcionamiento

¹ Biblioteca de funciones en C que permiten utilizar los elementos gráficos para el desarrollo de aplicaciones en el entorno del sistema operativo UNIX.

² Objetos que controlan la interacción entre el programa de una aplicación y una ventana en el entorno X Window.

bastante mecánico, puesto que los pasos a seguir para la creación de una GUI son cíclicos (creación o apertura de una existente; creación, eliminación o modificación de *widjets* ya construidos; guardar la interfaz creada).

2.1. Descripción

En cuanto al aspecto de la interfaz principal de *xgui*, podemos decir que sigue un modelo o patrón estándar de aplicaciones gráficas. Esto implica dividir la ventana principal en las siguientes partes o componentes, tal como se puede observar en la figura 1:

- Barra de menú. Contiene los distintos menús desplegables con todas las opciones de la aplicación (bajo la barra de título).
- Barra de herramientas. Opciones más utilizadas de los menús *Fichero* y *Edición* (bajo la barra de menús).

- Nombre de la interfaz gráfica actual. Muestra el nombre de la GUI actual (situada a la derecha de la barra de herramientas).
- Ruta. Indica al usuario la ruta en la que se localiza el archivo que contiene la GUI actual (parte inferior de la aplicación).
- Barra de componentes, gestores y menús. Dividida en tres partes y contiene todos los *widjets* implementados por la aplicación (parte izquierda de la ventana).
- Ventana de trabajo. Ocupa el resto de la ventana principal.

Los menús *Fichero*, *Edición* y *Ayuda* contienen las opciones habituales (*abrir*, *guardar*, *guardar como*, *salir*, *cortar*, *copiar*, *pegar*, *borrar*, *buscar*, etc.) de la mayoría de las aplicaciones gráficas.

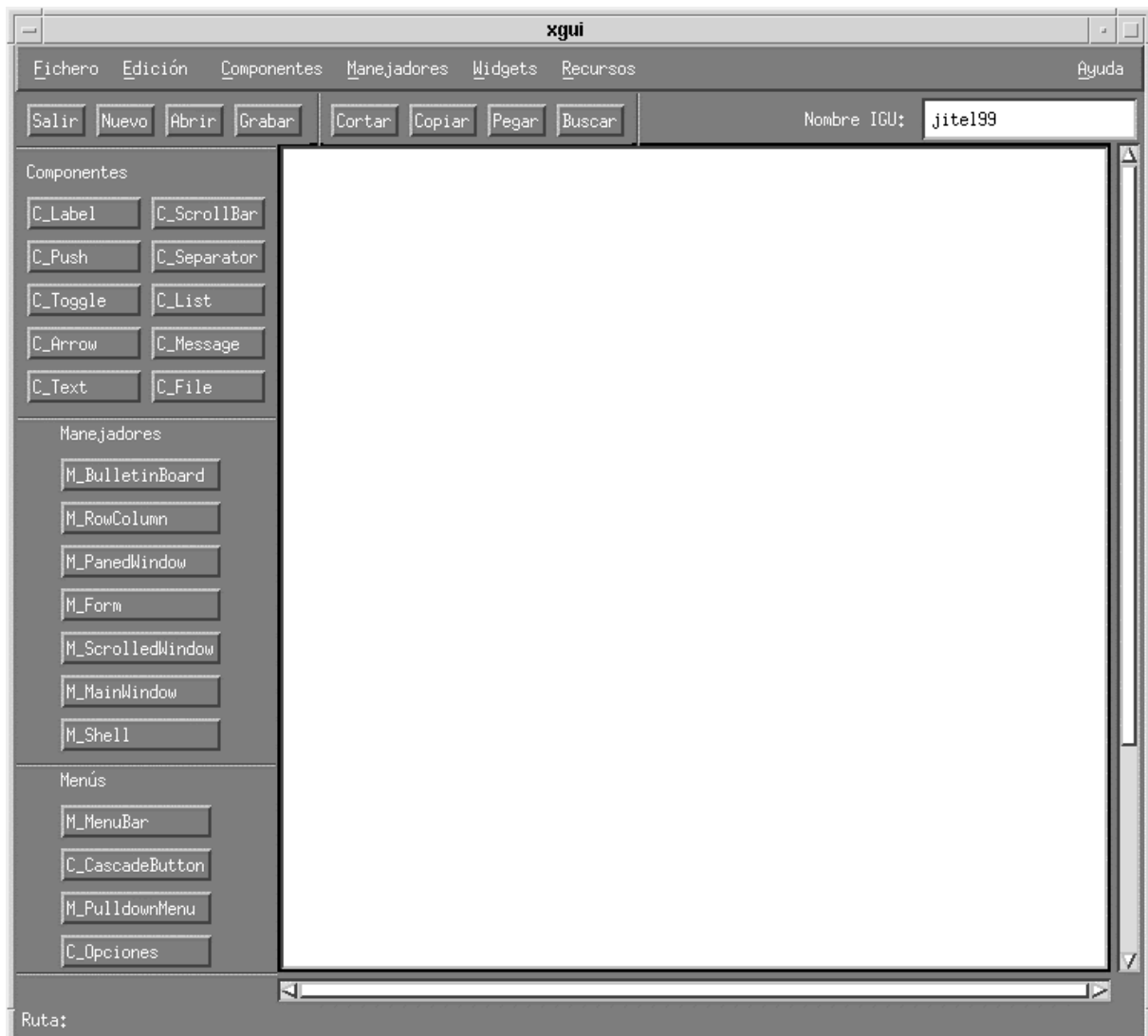


Fig. 1: Interfaz principal

El siguiente menú es el denominado *Componentes*. Éste contiene todos los *widgets* que constituyen los elementos básicos de la aplicación que se está diseñando (*etiquetas, botones, cajas de texto, etc.*).

A continuación nos encontramos con el menú *Manejadores* donde se encuentran los *widgets* que gestionan a los componentes del menú anterior (*BulletinBoard, RowColumn, Form, etc.*).

El menú *Widgets* contiene las opciones de gestión de los *widgets* ya construidos. Permite la visión o modificación de un *widget* ya creado, su eliminación, o mostrar el código generado tras su creación.

Por último, podemos destacar que el contenido de las opciones del menú *Recursos* trata sobre *Motif* en su totalidad, de forma que al pulsar sobre cualquiera de las opciones aparece el cuadro de diálogo correspondiente a la ayuda seleccionada.

2.2. Utilización

Para comprender los conceptos que permiten manejar la herramienta *xgui* no es necesario poseer grandes conocimientos del lenguaje C³. Sin embargo, a la hora de crear una aplicación gráfica más profesional, es necesario que el programador domine los conocimientos que le permitan dotar de funcionalidad a la interfaz que está diseñando.

Como se ha mencionado anteriormente, *xgui* genera código y éste debe ser volcado a dos archivos distintos. El primero de ellos constituye un archivo C estándar y contiene el código fuente en C generado por la creación del *widget* y por la gestión de los eventos del mismo. El segundo de ellos es un archivo de texto cuyo contenido representa el código de los valores de los recursos del *widget* creado (propiedades como la altura, el ancho, posición, etc.).

Los motivos por los que se emplearon dos archivos para la creación de interfaces gráficas se fundamenta en los siguientes argumentos:

- Incrementar la flexibilidad de un programa, permitiendo al usuario personalizar el aspecto de la interfaz de la aplicación con la que está trabajando.
- Desde el punto de vista del programador, los ficheros de recursos facilitan la tarea de la programación. Esto se debe a que se decide qué *widgets* utilizar, se escribe el programa que los crea y entonces se pueden ajustar las especificaciones en el fichero de recursos hasta alcanzar el aspecto deseado. Todo esto sin necesidad de modificar el código del programa



Fig. 2: Interfaz del ejemplo

y, por tanto, sin necesidad de volver a compilarlo.

- Simplificar el archivo que contiene el código fuente, evitando de esta forma sobrecargarlo con especificaciones de valores de recursos.

Para que los valores de los recursos almacenados en el archivo de recursos surtan efecto, se debe establecer la variable de entorno: *XAPPLRESDIR*. En ella se debe indicar el nombre del directorio donde se va a insertar el fichero de recursos o bien, por omisión, asignar el directorio actual de trabajo (en el sistema de archivos UNIX se representa por un punto, por ejemplo: *XAPPLRESDIR=.*).

2.3. Un ejemplo práctico

Para obtener una aproximación más clara de lo que hace *xgui*, presentamos un ejemplo en el que se trata de obtener una representación de la información referente al contenido y autores del presente artículo, tal como se muestra en la figura 2.

Este ejemplo muestra dos *widgets* del tipo botón y un *widget* del tipo etiqueta, todos ellos sobre un *widget* del tipo *bulletin board* que gestiona los recursos de los primeros.

Con la pulsación sobre el botón *Artículo* aparecerá en la zona destinada a la etiqueta una breve descripción del artículo, tal como se muestra en la figura 3.

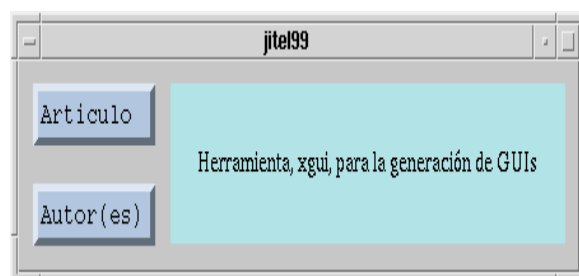


Fig. 3: Acción botón *Artículo*

³ Dado que *Motif* es una biblioteca de funciones escritas en C.

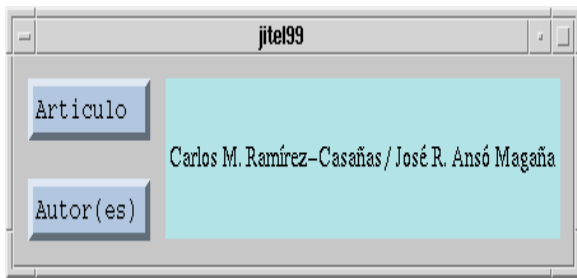


Fig. 4: Acción botón *Autor(es)*

En la figura 4 podemos observar el efecto que se obtiene al pulsar sobre el botón *Autor(es)*.

Los pasos necesarios para realizar esta interfaz son los que se describen a continuación. La primera etapa en toda creación de cualquier GUI, utilizando *xgui*, es la de acceder a la opción *Nuevo* del menú *Archivo*. Aparecerá el diálogo que solicita el nombre del *widget* que hará de interfaz entre la aplicación y el servidor gráfico. En nuestro ejemplo, tal como se puede observar en la figura 5, el nombre introducido es *jitel99*.

Tras pulsar sobre el botón *Aceptar* aparecerá en la ventana de trabajo el código de la función *main()*. Este código habrá que seleccionarlo, evitando los comentarios si así se desea, para copiarlo al editor que se esté utilizando.

Activando la función *Copiar* pasamos una copia del código seleccionado al portapapeles. Más tarde sólo se ha de activar la función *Pegar* en el editor que se utilice. Esto permite insertar en el archivo fuente C, denominado en este caso *jitel99.c*, el código fuente de la función *main()*.

A partir de este punto podemos comenzar la construcción de los *widgets* que constituyen nuestra interfaz gráfica. Cuando se crea un *widget*, se hace como hijo de otro *widget* (su padre). Las relaciones paterno-filiales producidas por este proceso forman una estructura en árbol que se conoce como el árbol de instancias.

En *xgui* todos los diálogos de construcción de *widgets* contienen una zona dedicada a este concepto. Además cada diálogo permite a los usuarios establecer propiedades inherentes a la clase del *widget* que deseamos diseñar (recursos de apariencia, posición, tamaño, etc.).

El primero *widget* a construir es de la clase *bulletin board*. En la figura 6 observamos el diálogo de construcción de este *widget*, una vez hemos seleccionado el botón *M_BulletinBoard* de la interfaz principal.

Tras introducir las relaciones con el *widget* padre (*jitel99*) y al pulsar sobre el botón *Aceptar* la ventana de trabajo mostrará el código de creación de este *widget*. El código fuente se vuelca al archivo

jitel99.c y las especificaciones de recursos al archivo *Jitel99*.

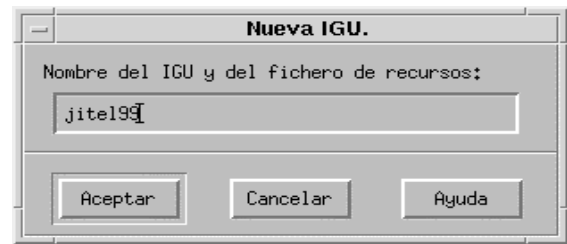


Fig. 5: Diálogo de creación de nueva GUI

El siguiente *widget* a crear es el botón *Artículo* cuyo código se inserta a continuación del *bulletin board*.



Fig. 6: Diálogo del *BulletinBoard*

El diálogo de creación se muestra en la figura 7.

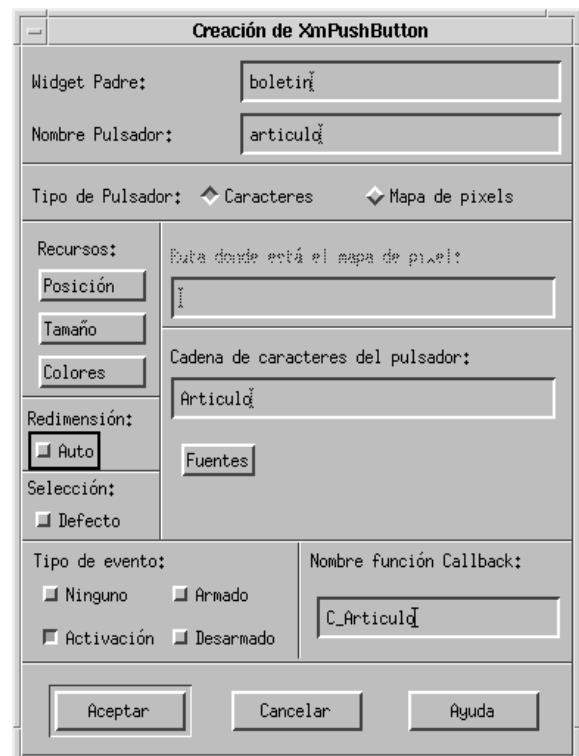


Fig. 7: Diálogo de creación de botón

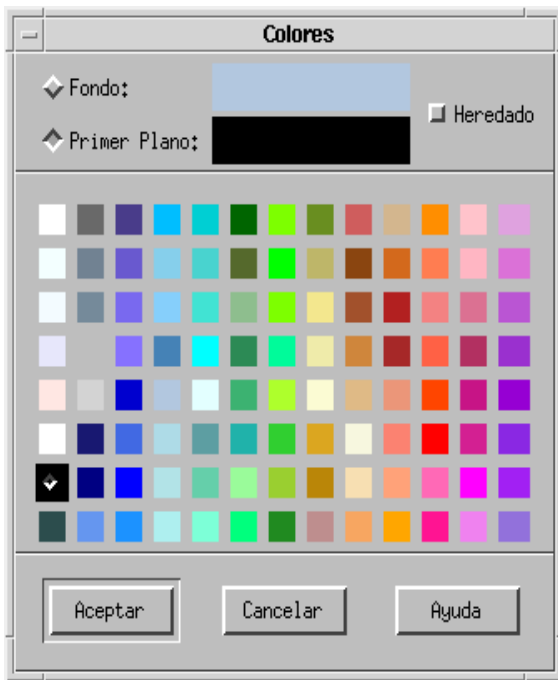


Fig. 8: Diálogo de recurso *Colores*

Un aspecto importante a mencionar es que para obtener una respuesta al pulsar sobre el botón, es necesario añadir el código correspondiente a la función *Callback*⁴ que controla el evento de activación del botón. En el ejemplo la denominamos *C_Articulo*.

Los valores de los recursos se mantienen inalterables, salvo el de los colores y las fuentes de caracteres cuyas especificaciones se pueden observar en las figuras 8 y 9 respectivamente.

El otro botón que aparece en nuestro ejemplo, *Autor(es)*, se construye de forma idéntica al que se acaba de mostrar, siguiendo los pasos que se han citado.

El último *widget* que nos queda por crear es la etiqueta. Ésta nos permite mostrar las cadenas de caracteres que deben aparecer al pulsar sobre cada uno de los botones. Su construcción es análoga al de los botones y su cuadro de diálogo se muestra en la figura 10.



Fig. 9: Diálogo de recurso *Fuentes*



Fig. 10: Diálogo de construcción de *Etiqueta*

En el caso en el que una vez compilado y ejecutado el programa, el resultado obtenido no se ajuste a los objetivos inicialmente planteados en cuanto a su aspecto, podemos acceder al diálogo de modificación de los *widgets* ya creados. Para ello utilizamos la opción *Ver/Modificar* que se encuentra en el menú *Widgets* de la interfaz principal, con lo que activaremos el diálogo de la figura 11.



Fig. 11: Diálogo *Ver/Modificar Widgets*

⁴ Función que contiene el código del evento (*click*, *doble click*, etc.) que se desea controlar.

Cada uno de los *widgets* viene representado por una línea en el cuadro de diálogo. Estas líneas están compuestas por el nombre del *widget* que se desea mostrar y el nombre del *widget* padre separados por el carácter punto (por ejemplo, *etiqueta.boletin*).

Por último, si el usuario necesita una ayuda interactiva para consultar cualquier función o aspecto de la programación con *Motif*, éste puede recurrir a las opciones que se encuentran en el menú *Recursos*. De esta manera se accede al diálogo que se muestra en la figura 12.

3. Estructura de la herramienta, *xgui*

La aplicación se estructuró en siete módulos principales: *Inicialización*, *Construcción de widgets*, *Almacenamiento en memoria*, *Gestión de widgets*, *Edición*, *Recursos* y *ayuda*, y un séptimo módulo compuesto por funciones comunes a varios archivos (manejo de cadena de caracteres, vectores de controles, etc.), como se puede observar en la figura 13.

3.1. Módulos de la aplicación

El primero de los módulos se encarga de la creación de la interfaz principal de la herramienta *xgui* y de colocar el programa bajo el control del gestor de eventos. Este módulo, a su vez, consta de dos archivos. El primero de ellos es el responsable de la conexión con el servidor gráfico y de la construcción de las tablas de recursos internos desde los archivos de recursos correspondientes. El segundo tiene por misión la creación y activación de todas las opciones de los desplegables de la barra de menús, así como las opciones de las barras de botones.

El módulo de construcción de *widgets* está formado por todos los ficheros de diálogos de construcción de *widgets* (*label*, *row_column*, *bulletin_board*, *arrow_button*, etc.). Cada uno se encarga de la creación y gestión de los diálogos de construcción de los *widgets* implementados en esta aplicación. Además existe una parte que controla las especificaciones de recursos comunes a la mayoría de los anteriores (posición, tamaño, colores, fuentes y márgenes).

El módulo de almacenamiento de memoria utiliza estructuras de datos dinámicas en las que se almacena la información correspondiente a cada uno de los *widgets* que conforman la aplicación (nombre del *widget*, nombre del *widget* padre, vector de recursos, etc.).

El módulo de gestión de *widgets* contiene dos tipos de tareas bien diferenciadas. Por un lado la gestión de creación, grabación o recuperación de las GUIs con todos sus *widgets* asociados. Por otra parte la manipulación de cada *widget* por separado (modificación, eliminación o visualización del código generado por cada *widget*).

Las opciones de cualquier editor (copiar, cortar, pegar, borrar, buscar) se encuentran en el módulo de edición.

En el módulo de funciones comunes reside el conjunto de operaciones que se invocan desde los módulos anteriores para facilitar las tareas más comunes a todos ellos (control de botones y gestión de cadena de caracteres; control de eventos comunes a más de un *widget*; creación y gestión de mensajes).

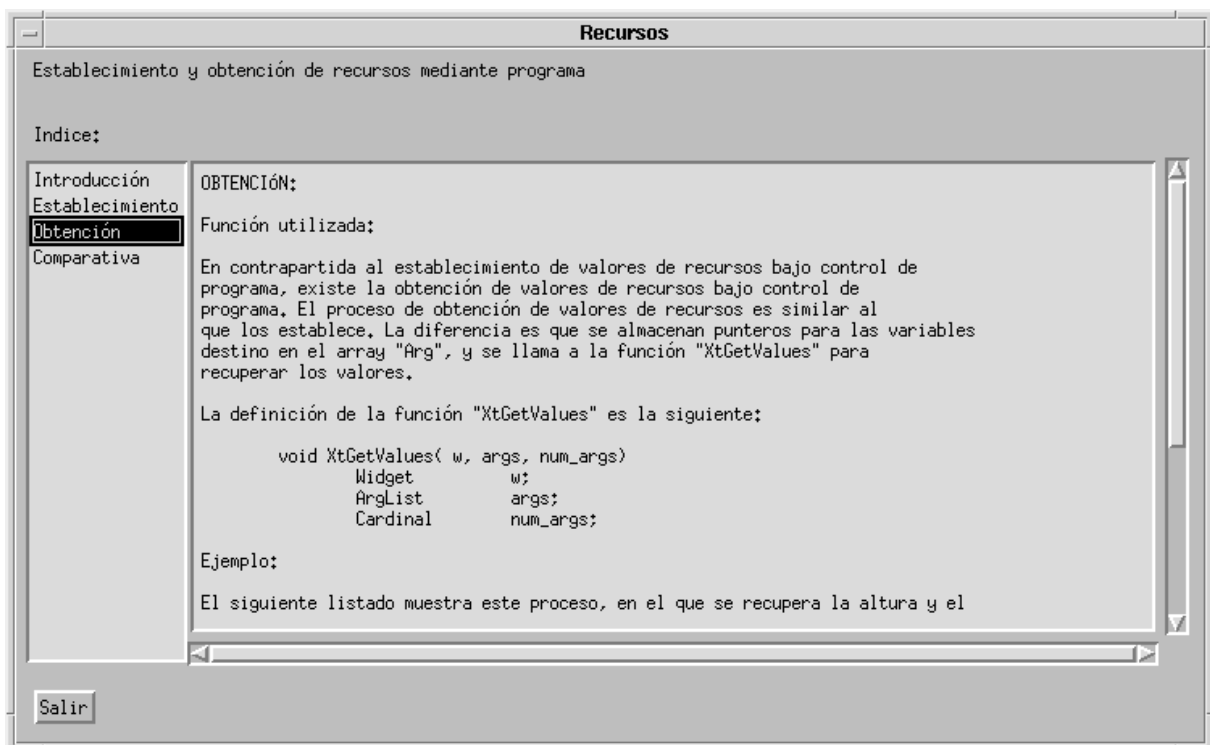


Fig. 12: Diálogo de ayuda *Motif*

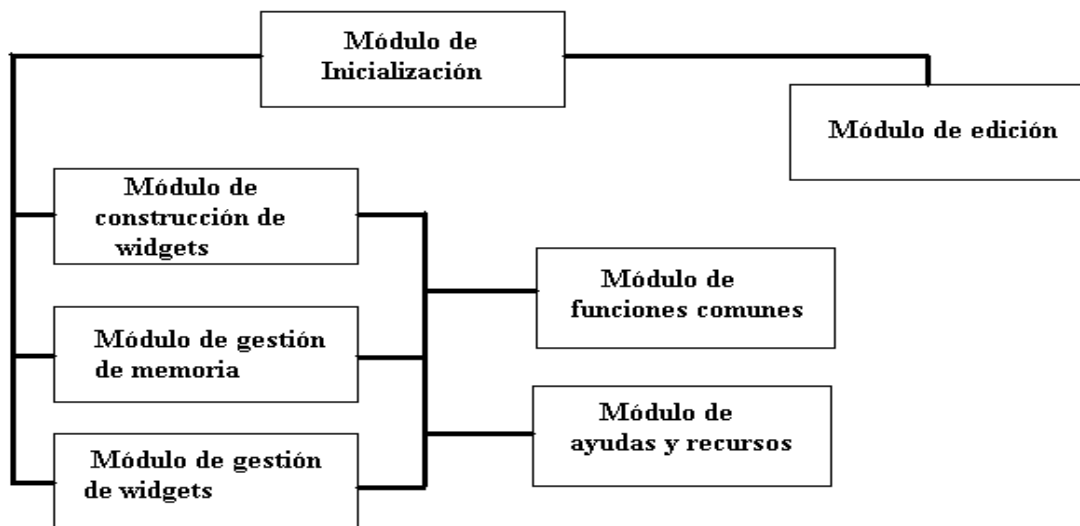


Fig. 13: Estructura de *xgui*

Por último, el módulo de recursos y ayudas crea y gestiona los diálogos de ayuda que posee la aplicación tanto de *Motif* como la guía de usuario de *xgui*.

4. Conclusiones

En este artículo se ha presentado una herramienta amigable y fácilmente incorporable en entornos académicos que nos permite utilizar los elementos gráficos sin conocimientos previos de la programación de interfaces gráficas de usuarios.

Esta aplicación intenta servir como una herramienta de software didáctico que introduce a sus usuarios en la programación con *Motif* a medida que la utilizan.

Por otro lado, se ha realizado una aplicación de uso intuitivo que posibilita al usuario adquirir conocimientos y práctica sobre la programación gráfica. Todo esto de forma gradual y sin necesidad de saber *a priori* qué significa cada una de las líneas insertadas en los archivos destino (código fuente en C y archivo de recursos).

Queremos destacar que la aplicación cuenta con un conjunto de ejemplos básicos de interfaces gráficas de usuario. Estos pueden ser tomados como punto de partida para posibilitar la realización de tareas muy sencillas o bien como modelos a imitar.

La creación de esta herramienta ha supuesto que los alumnos de Ingeniería de Telecomunicación, en particular y, en general, cualquier alumno de carreras técnicas, pudiera acceder al código que genera y gestiona los GUIs de forma sencilla.

Un aspecto relevante de esta herramienta, es que permite a sus usuarios utilizar el entorno de edición que consideren adecuado (*emacs*, *vi*, *textedit* u otros). Todo ello sin necesidad de tener que someterse al nuevo editor que proporciona la aplicación, con los beneficios en términos de

productividad que esto supone. De esta forma el programador puede realizar su trabajo de manera más cómoda sin ningún paso intermedio para empezar a utilizarla.

Por último permite que con el tiempo y tras la asimilación de ciertos conceptos que se van empleando a lo largo de su utilización, se pueda trabajar de forma independiente de la herramienta para crear sus propias aplicaciones gráficas.

Referencias

- [1] K.D.Gregory. "Programming with Motif". Springer-Verlag, 1992.
- [2] D.Heller. "Motif Programming Manual (Volume VI. The Definitive Guide to the X Window System)". O'Reilly & Associates Inc., 1992.
- [3] D.Flanagan. "Motif Tools (Volume VI C. The Definitive Guide to the X Window System)". O'Reilly & Associates Inc., 1994.
- [4] A.Nye, T.O'Reilly. "X ToolKit Intrinsic Programming Manual (Volume IV. The Definitive Guide to the X Window System)". O'Reilly & Associates Inc., 1990.
- [5] D.Flanagan. "X ToolKit Intrinsic Reference Manual (Volume V. The Definitive Guide to the X Window System)". O'Reilly & Associates Inc., 1990.
- [6] User Interface Development Environment. Amulet. www.cs.cmu.edu/afs/cs/project/amulet. Carnegie Mellon University.
- [7] The Tool Command Language and the Tool Kit. www.tcltk.com. University of California Berkeley.
- [8] Toolkit QT. www.troll.no/products/qt.html. Troll Tech.

Interfaz de Usuario basada en Mundos VRML para un Museo Virtual Tridimensional en Internet

J. F. Díez Higuera & F.J. Díaz Pernas & R. Mompo Gómez
E.T.S. Ingenieros de Telecomunicación. Universidad de Valladolid
Campus Miguel Delibes - Camino del Cementerio, s/n 47011-Valladolid
e-mail: josdie@tel.uva.es

Abstract

The use of Internet for Art spreading has been up to now limited to some photographs, but now, thanks to VRML (Virtual Reality Modeling Language), any museum can be recreated and anybody can visit it with virtual reality thousands of miles away. Last years have brought several attempts to develop Virtual Museums in Internet, most of them actually being no more than a compilation of texts about their collections plus bidimensional (2D) pictures. This work describes a design which, using the most advanced technologies now in use in the Web, implements a Virtual Museum for its access through Internet so it can not only provide the information so far available in the Web, but also allows a three-dimensional (3D) walk through the whole Museum, plus a presentation of single objects in 3D. The Art and the technique work together in a particular case, the Museum of Valladolid, giving rise to a simply spectacular result.

1. Introducción

En los últimos años, han alcanzado gran popularidad los clientes de propósito general del tipo World Wide Web para la consulta estática de texto e imágenes, es decir, los programas de navegación Web, debido en gran medida a la gran expansión que ha experimentado la red Internet. Mediante un sistema de hipertexto, las personas que disponen de una conexión a Internet y de un programa de navegación (como Netscape o Internet Explorer) pueden seleccionar y ver en su pantalla información procedente de todo el mundo.

De esta manera cada vez hay una mayor cantidad de personas que acceden a Internet y demandan todo tipo de información, de forma que tanto empresas como particulares están encontrando en Internet un lugar idóneo para difundir toda clase de información, desde información puramente comercial hasta cualquier tipo de hobbies o curiosidades con la ventaja de la ausencia de fronteras, lo cual es ideal para el negocio de las pequeñas y medianas empresas.

No es de extrañar, por tanto, que cualquier institución vea ahora la posibilidad de darse a conocer en todo el mundo. De ahí surge la idea de diseñar una plataforma de servicios telemáticos interactivos que permitan la visita a un Museo a través de Internet, de forma que no sea necesario viajar para poder contemplar las maravillas del mismo. Este es el objetivo del trabajo presentado en este artículo.

Y puesto que vamos hacia una sociedad de la imagen, la mejor manera de mantener el patrimonio vivo es adecuar los museos a los nuevos tiempos, de forma que se facilite a la gente la posibilidad no sólo de visitar y conocer el museo desde casa sino que, además, pueda realizar una visita por el edificio así como por sus salas. Esto implicaría la incorporación

de la realidad virtual [5] como instrumento para realizar esta tarea, aunque actualmente no se contemple esta posibilidad en los museos que se pueden visitar a través de Internet [1][4]. Con ello se pretende emular acciones y experiencias con que interactuamos en el mundo real aunque, obviamente, con ciertas limitaciones. Se consigue así que el usuario pueda andar, examinar e interactuar con el entorno a diferencia de los medios tradicionales como la televisión que presenta unos gráficos excelentes pero carece de interactividad. Aunque esta idea no es nueva, sí es actualmente cuando está logrando una mayor expansión debido, sobre todo, a la creación de un lenguaje común para la descripción de escenas en 3D e hiperenlaces WWW (una analogía del HTML para realidad virtual): VRML, *Virtual Reality Modeling Language* [6].

Esta es quizá, la novedad más interesante para los internautas ávidos por descubrir nuevos sitios interesantes en Internet y para la gente que la utiliza como pasatiempo, pero VRML permite, además, ver las piezas del museo en tres dimensiones, es decir, nos permite ver una pieza tridimensional desde cualquier ángulo y perspectiva, girarla en cualquier sentido... en resumen, manipularla, algo totalmente prohibido en un museo real. Una posibilidad realmente interesante para la gente que entiende de arte, tanto expertos como aficionados puesto que permite visualizar cualquier detalle a partir una replica virtual exacta de la pieza seleccionada.

La idea general para el diseño de dicha plataforma, sería tener un servidor Web mantenido por el museo al que los usuarios podrían acceder y consultar cualquiera de las piezas o, simplemente darse una vuelta por el museo[2]. Para llevar a cabo este cometido, el sistema global consta de dos grandes bloques: el de interfaz con el usuario y la aplicación cliente-servidor que se encarga del acceso al servidor que va

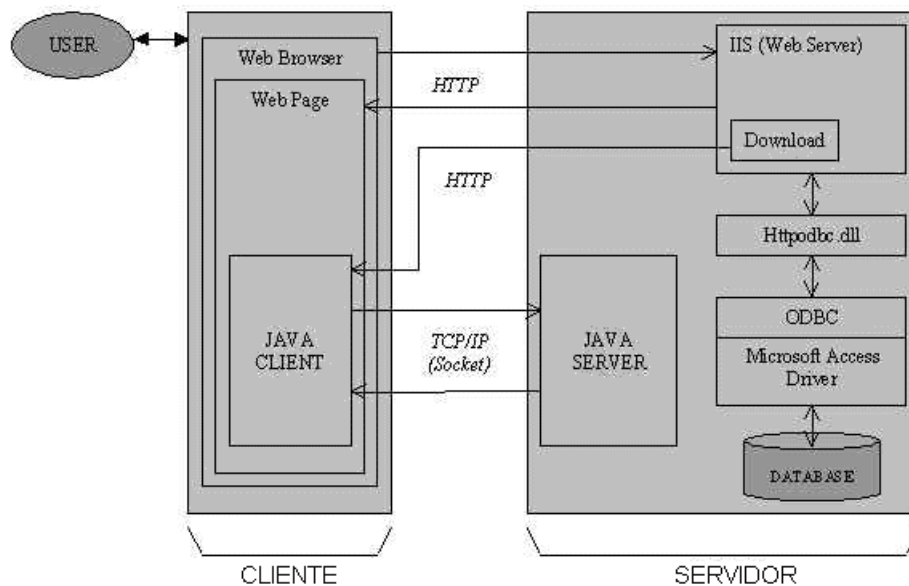


Figura 1. Arquitectura general del sistema.

a contener la información que vaya solicitando el usuario. El primero de los bloques está por encima del otro e interactúa con él.

El presente artículo se encargará de la descripción del primero de los bloques que es el que presenta la información al usuario. Como particularización del sistema propuesto, se presentará el Museo de Valladolid, ubicado en el Palacio de Fabio Nelli. Podrá hacerse una visita virtual sin más requerimientos que tener instalado en el navegador uno de los muchos programas que permiten la visión de mundos VRML. Describiremos los diferentes mundos que hemos creado y cómo el usuario puede interactuar con ellos de forma que pueda pasar de un mundo a otro y, además, pueda obtener todo tipo de información correspondiente a cada pieza. A continuación, después de una breve descripción de cada uno de los mundos se presentará una serie de secuencias procedentes de una simulación que nos permitirá ver alguna de las posibilidades que ofrecen estos mundos virtuales.

2. Diseño y estructura del Museo Virtual

El módulo de presentación-interacción con el usuario esta basado en un sistema de páginas Web donde en una parte de la pantalla se representa el mundo virtual correspondiente mientras que en el resto figura el nombre del contenido de la pantalla más las opciones que se dejan para que el usuario seleccione la que más le interesa. Vamos a tener un sistema básico por el que podremos movernos por el edificio y las salas y, ya en estas, se podrá seleccionar un cuadro o una pieza 3D para obtener información acerca de los mismos. La estructura general de este módulo es la siguiente aparece en la fig. 1.

Como puede observarse en la fig. 2 el recorrido del museo empieza en la fachada del mismo que posteriormente describiremos. A partir de aquí el usuario

puede desplazarse a su gusto y ver cualquier cosa que desee. Para acceder al patio interior no hay más que picar en la parte inferior de la puerta. Esto le llevará a otra página distinta. Aunque esto podría hacerse en el mismo archivo, lo mejor es separar ambas partes debido a varios factores. Primero, se reduce la extensión del mundo con lo cual se agiliza el sistema, evitando una espera demasiado larga. Segundo, le damos la opción al usuario de entrar o no, es decir, puede visualizar la fachada y decidir que aquello no le interesa con lo cual evitamos traer información que no se va a utilizar. El único inconveniente es que restamos continuidad a las acciones del usuario, es decir, debe esperar a que el fichero que contiene el patio interior llegue con lo cual se corre el riesgo de pérdida de interés por parte del mismo. Sin embargo, es preferible esta espera una vez que el usuario se ha interesado por el museo a una espera inicial demasiado larga.

Una vez que estamos dentro del patio, el usuario puede pasear por él, subir unas escaleras para acceder al segundo piso o incluso pulsar en la puerta de entrada para volver otra vez a la fachada. También tiene la opción de entrar en una sala que contiene algunas de las obras más representativas del museo. Si decide pasar a la sala entraremos en el siguiente bloque. En la sala tenemos ya las piezas de museo. Se puede recorrer la sala con total libertad y mirar las piezas desde casi cualquier perspectiva que deseemos. Si al usuario le interesa una pieza puede seleccionarla mediante el ratón y aparecerá una página Web distinta en caso de que se trate de un cua-

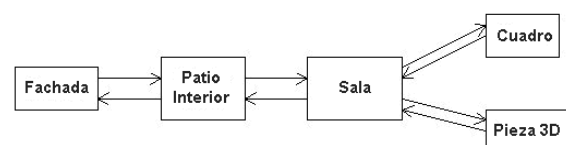


Figura 2. Estructura general de la visita al Museo.



Figura 3. Página Web inicial

dro o de una pieza 3D con una serie de información que el usuario puede utilizar.

En este momento entra en acción la aplicación Java cliente-servidor de la que hablábamos en un principio y que se encargará de proporcionar la información requerida por el usuario. La información de texto solicitada aparecerá en una página Web mientras que si se solicita una vista 3D de una pieza (obviamente no se puede solicitar una vista 3D de un cuadro) ésta aparecerá también en tres dimensiones en formato VRML, lo cual va a permitir que el usuario juegue con ella y pueda ver hasta los más mínimos detalles dependiendo de la resolución que tenga la pieza. También puede cambiar su aspecto, es decir, visualizar la misma pieza pero construida sobre otro material.

2.1. Fachada

Es el primero de los mundos virtuales que aparecen al visitar el Museo de Valladolid. Está compuesto de dos ficheros: uno que contiene la fachada propiamente dicha, más otro que contiene la plaza y los edificios anexos al Palacio. La razón de la existencia de dos ficheros en vez de uno es la comodidad en el diseño, puesto que consume bastante memoria en el equipo donde construimos estos mundos. Sin embargo, esto es totalmente transparente al usuario puesto que aunque deben llegarle dos ficheros, éstos constituyen un mismo mundo virtual.

En la parte donde se despliega el mundo virtual, el usuario puede moverse por él y ver el emplazamiento

del Palacio. En la fig. 3 aparece la imagen inicial mientras que la fig. 4 muestra una de las posibles vistas que pueden conseguirse.

Por otra parte, la fachada se compone de varios agrupamientos de objetos que pueden realizarse con el nodo *Group*, (los nodos describen los objetos y sus propiedades) mientras que al final, cada elemento acaba siendo un nodo *Shape*. Hay que hacer notar en el código el uso de la semántica *DEF-USE*. Esto es importante a la hora de hacer objetos que se repiten en la escena como pueden ser los balcones, los barrotes de los mismos, las columnas, etc. Mediante el uso de estas directivas VRML permite la creación de un objeto y su repetición en distintos lugares de la escena cuantas veces se quiera.



Figura 4. Vista panorámica del Palacio de Fabio Nelli.



Figura 5. Vista inicial del patio interior.

Para pasar al siguiente mundo virtual, utilizamos un nodo *Anchor* para representar la parte inferior izquierda de la puerta de forma que cuando el usuario pulse con el botón izquierdo del ratón sobre ella aparecerá en el siguiente mundo virtual. Es fácil averiguar que la puerta es sensible puesto que cuando se pasa el ratón sobre ella, la forma de éste cambia. De cualquier forma, está prevista una utilidad de ayuda en la primera página del museo para los visitantes no iniciados.

2.2. Patio

Este módulo se carga cuando el usuario pulsa en la puerta de la fachada del anterior módulo. Se compone de un fichero básico más otros tres ficheros que contienen la balaustrada, los medallones y las columnas. La razón de dicha separación es la misma que la argumentada en el caso de la fachada. La composición del conjunto representa el patio interior del Palacio de Fabio Nelli. Finalmente, justo al otro extremo del patio se encuentra la puerta que da paso a las dos habitaciones virtuales del museo. También está la posibilidad de que el usuario pulse la puerta de entrada y vuelva otra vez a la fachada del museo.

En la fig. 5 aparece la perspectiva inicial mientras que en las figs. 6 y 7 se muestran diferentes perspectivas del mismo.



Figura 6. Vista general del patio interior.



Figura 7. Vista general desde arriba (esquina izquierda).

2.3. Sala

Accedemos a ella a través del patio. No es una sala real contenida en el Museo sino una sala virtual que contiene las obras más representativas del mismo. Esto es así debido al carácter experimental del proyecto que pretende basarse en un museo real para no crear luego falsas expectativas. Sin embargo, se ha intentado ser lo más fiel posible de forma que el aspecto que ofrece es el mismo que ofrecen la mayoría de las salas del Museo. Contiene principalmente obras pictóricas, aunque puede apreciarse también un jarrón tridimensional y un sarcófago. La novedad consiste en que el usuario puede pasearse por la sala e ir de obra en obra pudiendo seleccionar en cualquier instante la que más le interese para obtener información sobre ella. En las figs. 8 y 9 se pueden ver distintas perspectivas de la sala.

La interacción se realiza de la misma manera en la que pasábamos de un mundo a otro, es decir, mediante un nodo *Anchor* que debe pasar como parámetro el identificador correspondiente de la pieza en la base de datos y el nombre de la sala para que se obtenga la página Web correspondiente donde el usuario solicitará la información que quiera de la pieza. En este momento es cuando la aplicación cliente-servidor se encarga de acceder a la base de datos y de construir la correspondiente página con dicha información. Finalmente, desde la sala puede



Figura 8. Vista inicial de la sala virtual

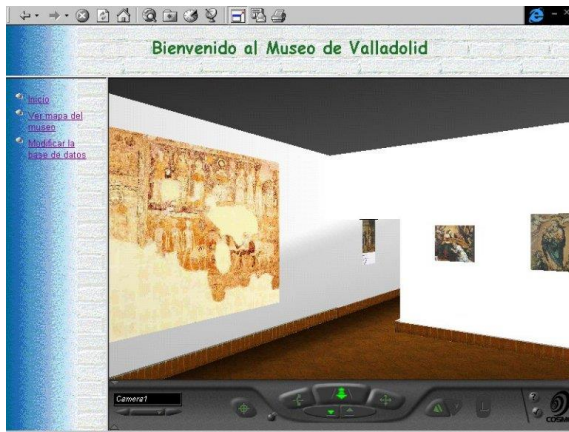


Figura 9. Vista de la sala virtual desde el lado opuesto.

volverse al patio a través de la puerta sensible que aparece en la sala y que es igual que la que aparecía en el patio para acceder a la sala.

2.4. Representación de una pieza tridimensional

Como ya se ha mencionado, una de las cualidades de este sistema es que nos permite la visualización de una pieza en 3D en formato VRML lo cual implica que el usuario puede manipularla a su antojo (ver fig. 10). Operaciones que se pueden realizar son: alejamiento, acercamiento a una determinada zona, rotación en cualquier dirección... Además, al usuario se le da opción de pedir más resolución por si quisiera ver algún detalle que no estuviese demasiado claro y fuese de suma importancia para él. También puede visualizar cualquier pieza construida sobre

otros materiales, para lo cual debe elegir la opción correspondiente.

Las piezas en 3D que se representan son una réplica exacta de las que aparecen en el museo (En caso contrario no tendría demasiado sentido puesto que estamos hablando de piezas de museo). Para ello, se utiliza el sistema de captación descrito en [3] y posteriormente se construye el fichero VRML.

Conclusiones

En este artículo se ha presentado una técnica innovadora que permite acercar la cultura a todo el mundo. Como parte de la interfaz con el usuario y mediante la utilización del Virtual Reality Modeling Language se han construido una serie de mundos virtuales que presentan el entorno del museo, el patio interior del edificio y una sala representativa del mismo. Dentro de esos mundos virtuales el visitante puede moverse por ellos e interactuar con el entorno dándole una nueva dimensión a la visita. El usuario puede elegir a dónde ir y qué ver, casi de forma tan real como si estuviera en el propio museo y, lo más importante, puede seleccionar cualquier pieza 3D para su visión detallada, la cual podrá manipular a su antojo. Esto supone una gran innovación sobre los museos que actualmente se pueden encontrar en Internet, que no ofrecen este tipo de visita.

Se ha demostrado la viabilidad del proyecto, así como su correcto funcionamiento en la red para el caso particular del Museo de Valladolid. Una de las claves de VRML que hace esto posible es la extensión



Figura 10. Ejemplo de una pieza tridimensional.

del código de los mundos virtuales. Todos los mundos virtuales presentados anteriormente tienen una longitud de código inferior a 107 Kb lo cual asegura que su utilización a través de Internet será viable. Hay que comentar que la complejidad de los mundos virtuales puede variar dependiendo de lo que se quiera mostrar, por tanto, siempre va a haber un compromiso entre velocidad y complejidad puesto que a mayor complejidad de los mundos virtuales mayor longitud de código tendrá y, por tanto, su transmisión a través de Internet será más lenta. Así, la única restricción a este sistema es que el usuario necesita tener un VRML browser para poder visualizar los mundos virtuales, lo cual no es tanto problema puesto que en Internet es posible encontrar varios de libre distribución.

Finalmente, comentar que con este artículo se abre otra posibilidad al uso de VRML en Internet y se pone de manifiesto el por qué del éxito que esta tecnología está teniendo en los últimos tiempos. Aunque todavía queda mucho por andar, está claro que se camina hacia una nueva dirección en cuanto a las posibilidades que ofrece Internet, pasando de un mundo bidimensional a un universo tridimensional.

Agradecimientos

Los autores quieren agradecer el apoyo proporcionado por CEDETEL (Centro para el Desarrollo de las Telecomunicaciones en Castilla y León), así como las aportaciones a este trabajo realizadas por los

miembros del Grupo de Telemática Industrial de la Universidad de Valladolid. La colaboración de D^a Eloisa Wattenberg y D. Javier Sarabia ha contribuido notablemente a la implantación real del sistema propuesto.

Referencias

- [1] Cheng, J. & S. Malaika. *Web Gateway Tools. Connecting IBM and Lotus Applications to the Web*. Wiley Computer Publishing, 1997.
- [2] Díez Higuera, J.F. & F.J. Díaz Pernas. *MINERVA: Multimedia on the IntERnet for Virtual Arts*. In Proceedings of the International Conference on Multimedia Computing and Systems, pp. 825-829, Florence (Italy), June 1999.
- [3] Gómez García-Bermejo, J. *Obtención de las Informaciones Tridimensional y Colorimétrica. Caracterización Óptica de Superficies*. Universidad de Valladolid (España), Dpto. de Ingeniería de Sistemas y Automática, ETSIT, Tesis Doctoral, Mayo 1995.
- [4] Intervista Software, Inc. "*Compelling uses of 3D on the Web*", 1998 <http://www.intervista.com/vrml/whitepapers/3duses.html>
- [5] Jerry Isdale. "*What is Virtual Reality*", 1993. <http://www.cms.dmu.ac.uk/~cph/VR/whatisvr.html>
- [6] San Diego Supercomputer Center Inc., "*The VRML Repository*", 1998. <http://www.sdsc.edu/vrml/>

Capítulo 5A: Análisis de Tráfico

Algoritmo Distribuido de Reconfiguración de Redes Virtuales ATM basadas en VPs.
M. Nicolau, J. Abella, G. Corral, F. Garcia (U. Ramón Llul)

Arquitectura de Encaminamiento para Servicios Garantizados.
M.C. Oria, J.A. Lozano, A. Ariza y F. Sandoval (U. Málaga)

Análisis comparativo de modelos dinámicos de control de admisión en redes ATM.
M. Fernandez Veiga, C. Lopez Garcia (U. Vigo)

Algoritmo Distribuido de Reconfiguración de Redes Virtuales ATM basadas en VPs

Miquel Nicolau, Jaume Abella, Guiomar Corral, Ferran Garcia
Àrea de Telemàtica d'Enginyeria i Arquitectura La Salle, Universitat Ramon Llull
Paseo Bonanova 8, 08022 Barcelona Telf: 93-2902400, Fax: 93-2902416
E-mail: {miqueln,jaumea,guiomar,fgarcia}@salleURL.edu

Abstract

In this paper we propose a distributed algorithm for the design of virtual networks based on a layout of virtual paths (VP's), on a given ATM network. We study the different proposals found on the literature and propose a possible solution based on economic models. Then we describe the algorithm and the simulation test-bed implemented. Finally, results, conclusions and future work are given.

1. Introducción

El diseño y configuración de redes de comunicación es un tema que ha tomado recientemente una importante relevancia básicamente por dos factores: el crecimiento espectacular de la complejidad de las redes de comunicación y la aparición de las redes ATM.

De hecho, el diseño y configuración de redes de comunicación no es un tema nuevo, pues existe un gran número de trabajos publicados sobre este tema. Sin embargo, las redes ATM presentan ciertas características claramente diferenciadas respecto a las redes más convencionales utilizadas hasta hace poco. Principalmente deben soportar múltiples tipos de servicios, garantizando la calidad de servicio (*Quality of Service*, QoS) contratada y, por ello, son redes mucho menos predecibles, ya que no se dispone de datos estadísticos fiables sobre las características de los diferentes tráficos ni de los patrones de petición de conexión. Estas propiedades hacen que los problemas de bloqueo y, por lo tanto, la asignación de ancho de banda (*Bandwidth*, BW) a los caminos virtuales (*Virtual Paths*, VPs), sean un elemento muy importante para el uso eficiente de las nuevas infraestructuras.

El objetivo de este proyecto es analizar, diseñar y realizar un sistema que permita una reconfiguración automática de las redes virtuales definidas en una red ATM, basándose en un algoritmo distribuido de asignación y mapeo del ancho de banda de los VPs necesarios, completamente independiente de una unidad gestora central. Dicha reconfiguración deberá realizarse para cada franja horaria que defina el administrador, cuya duración podrá variar desde unas pocas horas hasta varios días.

Para que esta solución sea aplicable en un entorno distribuido sobre los propios nodos ATM, es necesario que el diseño sea escalable, que el tiempo de cálculo no sea muy elevado, que el espacio de memoria sea limitado y que la sobrecarga debido a la propia señalización del algoritmo no implique una

reducción significativa del rendimiento de la red. Además, el sistema debe ser robusto y debe tolerar fallos en los equipos de comunicación.

Para ello, se ha dividido el trabajo en cinco secciones organizadas de la siguiente forma. En la segunda sección se introducen las distintas alternativas evaluadas. La tercera sección presenta el algoritmo implementado, basado en modelos económicos. Las secciones cuatro y cinco presentan el entorno de simulación implementado y los resultados obtenidos. Finalmente, se exponen las conclusiones y las líneas de futuro.

2. Estudio de alternativas

2.1 Precedentes

El diseño de redes de VPs, en su forma más simple, no es más que un problema de distribución de una serie de servicios (el ancho de banda de cada VP) sobre un conjunto de recursos (los nodos y los enlaces de la red), donde los principales puntos que deben resolverse son los siguientes:

- Definir la topología lógica de la red de VPs sobre la red física, definiendo los nodos extremo de cada VP.
- Dimensionar el ancho de banda óptimo de todos estos VPs, con la finalidad de minimizar una determinada función de coste.
- Planificar la reserva de los recursos redundantes necesarios para soportar los posibles fallos de nodos o enlaces que se produzcan en la red.

Actualmente, aunque el conjunto de trabajos relacionados es realmente extenso, son pocas las soluciones que aborden el problema en su totalidad, debido a la complejidad que esto representa.

La mayoría de los trabajos sólo aportan una **solución parcial** al problema partiendo, por ejemplo, de una topología de VPs determinada y proponiendo un algoritmo para el dimensionamiento del ancho de banda de cada VP [1], [2], [3], [4], [5], [6], [7], [8].

La razón fundamental es que la solución de la primera fase se convierte en un problema NP-Completo, tal y como se demuestra en [9]. Esto descarta encontrar un algoritmo global que encuentre, en cada caso, la mejor solución en un tiempo de cálculo razonable.

Las distintas alternativas que proponen una **solución integral**, tratan tanto el diseño de la red virtual como el dimensionamiento del ancho de banda de los VPs, [9], [10], [11], y también analizan la supervivencia de la red a posibles fallos, planteando modelos de planificación de los VPs redundantes que deben existir para soportarlos, [12] y [13].

Atendiendo a su implementación, todos estos algoritmos pueden clasificarse en algoritmos **síncronos** o **asíncronos** y en algoritmos **centralizados** o **distribuidos**, tal y como expone [14].

Los algoritmos síncronos actúan sobre la configuración de la red de VPs llamada a llamada, readaptando el ancho de banda de cada VP a las nuevas características del tráfico ofrecido, pero partiendo de una configuración inicial de VPs ya dada. Éste es el esquema presentado en [15], [16] y [17]. En cambio, los asíncronos mantienen una determinada configuración de la red de VPs durante un cierto periodo de tiempo T. Una vez finalizado este periodo se recalcula la nueva configuración, utilizando para esto algún método de predicción del tráfico que deberá soportar la red durante el nuevo periodo. Ejemplos de algoritmos asíncronos son los que se describen en [5], [8] [14], y [18].

Por otro lado, atendiendo al lugar dónde se ejecuta el algoritmo, podemos clasificarlos en centralizados y distribuidos. Los primeros se ejecutan en un punto concreto de la red, normalmente el mismo gestor, el cual necesita recibir y enviar información a todos los nodos de la red. Distintos algoritmos centralizados aparecen en [9], [14], [15]. En cambio, los distribuidos se ejecutan en cada nodo de la red, los cuales se pasan información entre ellos; ver [1], [4] y [15].

Todas las alternativas analizadas tienen un punto en común: se basan en la minimización de una cierta función de coste con la finalidad de encontrar la configuración óptima de la red. Ahora bien, la disparidad de las propuestas radica en el método aplicado para encontrar una solución que minimice dicha función. Los principales métodos de minimización están basados en teoría de control clásico [7], relajaciones lagrangianas [10], [11], [13], algoritmos genéticos [10], [19], métodos convolucionales [3], teoría de juegos [1], modelos económicos [1], [20], [21], [22] y métodos heurísticos [6], [7], [9], [13], [15].

La mayoría de estos métodos requieren una implementación centralizada del algoritmo, y son

pocas las alternativas que plantean una solución distribuida, como son la teoría de juegos y los modelos económicos. Esta última ha sido la alternativa escogida para la implementación del algoritmo, por su propia naturaleza distribuida.

2.2 Modelos económicos

Los Modelos Económicos introducen los conceptos de “dinero” y “precios” (*pricing*) para conseguir la simplificación y la descentralización de los algoritmos de distribución de recursos, convirtiendo los posibles escenarios a plantear en simples modelos de mercado. Para ello, unifican la naturaleza heterogénea de los distintos recursos del sistema en una única medida que es el precio, e introducen los conceptos de clientes, que comprarán recursos en función de su riqueza y de los precios establecidos, y de proveedores, los cuales controlarán los precios hasta conseguir el equilibrio entre la oferta y la demanda [22].

Existen otros modelos económicos que no se basan en el *pricing*, como los modelos basados en el intercambio. En estos modelos se introduce el concepto de agente, el cual dispone inicialmente de un conjunto de recursos del sistema. A lo largo del proceso, dichos agentes se intercambian recursos mientras este intercambio les produce algún beneficio.

En cualquier caso, estamos frente a un esquema completamente descentralizado en el que no hay un elemento central que controle los recursos, sino que existen un conjunto de entidades distribuidas que interactúan entre ellas para conseguir el mismo objetivo: maximizar el beneficio propio y, con ello, el beneficio de toda la red. Utilizando únicamente una unidad simple como es el precio del BW de los enlaces de la red, y tal y como se describe en el siguiente apartado, ha sido posible implementar dicho algoritmo distribuido para reconfiguración de redes basadas en VPs.

3. Descripción del algoritmo.

La solución que finalmente se ha desarrollado, basada en modelos económicos, utiliza el concepto de precio para regular la compartición de recursos entre los distintos usuarios de la red. Debido a esta implementación distribuida, el algoritmo se convierte en un método de búsqueda local, no disponiendo de información de toda la red, pero suficiente para encontrar una solución en un tiempo razonable.

El algoritmo parte de las necesidades de comunicación de los usuarios, expresadas mediante una topología lógica –origen, destino, ancho de banda y QoS de cada VP- y la topología física de la red –nodos y enlaces que la forman-. El objetivo del mismo es obtener el conjunto de rutas que cubran las necesidades de QoS de todos los caminos virtuales.

En este sistema de mercado el recurso que se desea “vender” es el ancho de banda de los enlaces físicos. Para ello se crean en cada nodo de la red dos entidades: consumidor y suministrador. El suministrador es el propietario de todos los enlaces físicos que empiezan en el nodo y se encarga de regular el precio de dichos enlaces. El consumidor se hace cargo de establecer todos los caminos virtuales requeridos por los usuarios que acceden a la red a través de dicho nodo en función de los precios existentes en la red, y de la topología física que le proporciona el PNNI (*Private Network-Network Interface*) [23].

En cada iteración, los agentes consumidores escogen, de entre el conjunto de rutas que garantizan la QoS requerida, la rutas más económicas para cada VP, e informan de sus intenciones a los agentes suministradores implicados. Una vez realizadas todas las peticiones, los agentes suministradores vuelven a evaluar el precio de los enlaces, a partir de una función de coste predefinida, anunciándolos posteriormente a todos los consumidores de la red.

De esta forma, los principales elementos a diseñar son los agentes **consumidores**, los agentes **suministradores** y las **funciones de pricing**.

3.1 Consumidor

El consumidor parte de la topología lógica definida por el administrador de la red y de la topología física.

La topología lógica es una lista de caminos virtuales que debe establecer entre el nodo al cual pertenece y un nodo destino. En esta lista, se encuentra asociado a cada camino virtual el ancho de banda necesario y sus características de QoS, como son el *Cell Transfer Delay* máximo (maxCTD) y el *Cell Delay Variation* máximo (maxCDV). Dado que la solución presentada es un algoritmo distribuido, el nodo no tiene información sobre todos los caminos virtuales de la red, sino sólo de los que le afectan directamente. De esta forma, la información que contendrá la topología lógica de cada consumidor será distinta en cada nodo y mucho menor de la que tendríamos en una implementación centralizada.

La topología física de la red contiene información sobre todos los nodos de la red y los enlaces físicos. Esta información se encuentra en cada nodo de la red, proporcionada por el propio PNNI [23], por lo que no es necesario incrementar el espacio de memoria requerido y se elimina la fase previa de descubrimiento de dicha topología.

Para satisfacer la QoS de cada uno de los VP's, el consumidor calcula inicialmente las k mejores rutas entre él y el nodo destino que cumplan con las restricciones de QoS. En cada iteración el consumidor escoge, para cada camino virtual, la ruta más económica y hace llegar una petición de ancho

de banda a cada uno de los nodos que forman parte de la ruta escogida, actuando de forma similar para las rutas de los VPs redundantes.

Cuando los consumidores reciben los nuevos precios, evalúan el coste de cada VP y seleccionan aquellos que superan un determinado precio máximo. Dichos VPs deberán modificar su ruta en función de una cierta probabilidad *Prob*_{cambio}, calculando nuevamente la ruta más económica. Si esta nueva ruta tiene un coste inferior a la anterior se procede finalmente al cambio de ruta, avisando a los nodos que formaban parte de la ruta anterior para que liberen el ancho de banda reservado.

3.2 Suministrador

La función del suministrador consiste en recoger todas las peticiones de ancho de banda que le corresponden y actualizar los precios de la red. Una vez se han recibido todas las peticiones de ancho de banda, el suministrador contabiliza, para cada enlace físico, la cantidad de ancho de banda reservada en el enlace. A partir de este valor calcula los nuevos precios y, finalmente, los propaga a todos los nodos de la red.

3.3 Funciones de pricing

El comportamiento de las dos entidades anteriores y, por lo tanto, el comportamiento del algoritmo en sí dependerá, principalmente, de las funciones de *pricing* utilizadas. Dichas funciones son las que utilizan el consumidor y el suministrador para decidir su comportamiento. El primero utiliza una **función de beneficio** para determinar si el VP correspondiente modifica o no su ruta. El segundo utiliza una **función de precio** para establecer el coste del BW de los enlaces en función de las reservas realizadas por los distintos consumidores.

En un modelo económico, la función de beneficio del consumidor debe provocar una disminución del beneficio de éste a medida que consume más recursos (figura 1(a)), o ser simplemente una función escalón o conjunto de escalones que permitan decidir la cantidad de recurso a reservar en función del precio de éstos (figura 1(b,c)) [24].

En nuestro caso, la función utilizada es la función escalón, ya que el comportamiento del consumidor es simplemente un “comportamiento binario”, ya que sólo decide si, en función del precio, el VP debe cambiar o no de ruta. No cabe la posibilidad de aumentar o disminuir el BW de los VPs, ya que el BW requerido es conocido e invariable, pues se trata de una implementación asíncrona.

En el caso del suministrador, éste debe disuadir a los consumidores a reservar más ancho de banda del disponible en cada enlace físico.

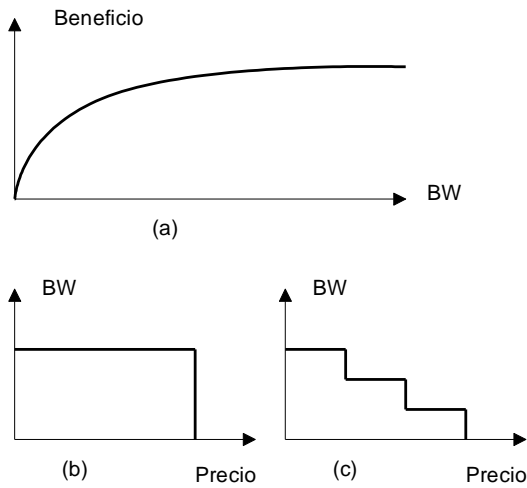


Figura 1: Funciones de beneficio del consumidor

De esta forma, el precio del ancho de banda debe tender a infinito a medida que nos acercamos a la capacidad máxima del enlace (figura 2(a)), pudiendo sustituir este valor infinito por un valor máximo supuestamente inaceptable por los consumidores (figura 2 (b,c)).

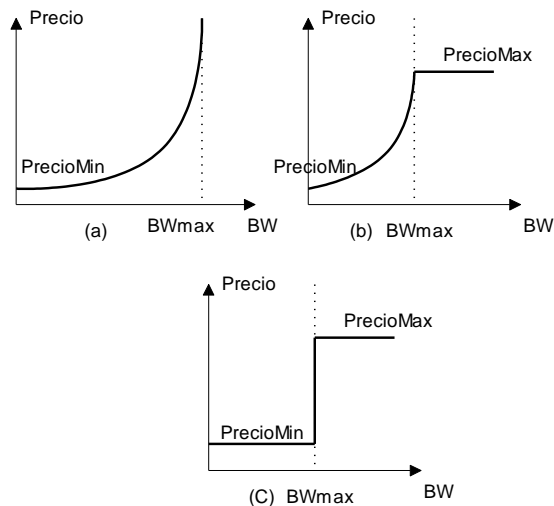


Figura 2: Funciones de precio del suministrador

Dichas posibilidades han sido analizadas sobre una plataforma de simulación, exponiendo posteriormente los resultados obtenidos.

4. Entorno de simulación

En esta sección se describe el entorno de simulación utilizado y las pruebas realizadas para comprobar el correcto funcionamiento del algoritmo.

Todas estas pruebas han sido realizadas utilizando el entorno de simulación OPNET [25], el cual permite el modelado, simulación y análisis de redes de comunicaciones, protocolos y sistemas distribuidos.

El entorno de simulación está formado por una red de 6 conmutadores, parcialmente conectados entre sí con enlaces bidireccionales de 150Mbps, con un determinado CTD (*Cell Transfer Delay*) y CDV (*Cell Delay Variation*), con la topología física detallada en la figura 3. Sobre esta red se han definido en distintas simulaciones volúmenes de 100 hasta 500 VPs a crear entre los distintos pares de nodos origen-destino.

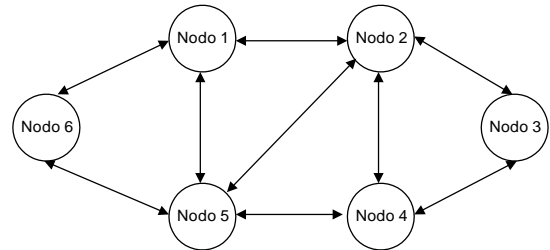


Figura 3: Topología física de la red de pruebas

La plataforma utilizada ha sido una estación WinNT Workstation sobre un Pentium-II con 64Mbytes de memoria RAM.

Para estudiar el comportamiento de la red se han extraído numerosas estadísticas de las distintas simulaciones. Los principales datos que se han extraído han sido la evolución del precio de cada enlace, el porcentaje de utilización de cada enlace y el precio medio de toda la red.

El precio de cada uno de los enlaces sirve para detectar qué enlaces están más saturados a lo largo de la ejecución del algoritmo. El porcentaje de utilización mostrará la eficacia del algoritmo para mapear correctamente el conjunto de VPs en situaciones próximas a la congestión. Finalmente, el precio medio del BW de todos los enlaces de la red nos proporcionará una medida indicadora de la evolución global de la negociación, así como de la convergencia del algoritmo hacia una solución final.

5. Resultados obtenidos

Como ya se ha comentado el algoritmo ha sido sometido a todo un banco de pruebas bajo distintas situaciones de densidad de VPs y de saturación de los enlaces de la red, cuyos resultados se exponen a continuación.

5.1 Convergencia y eficiencia del algoritmo

La herramienta más simple que dispone cada nodo de la red para evaluar la convergencia del algoritmo, y así decidir cuando se ha llegado a una solución global de toda la red, es el precio medio del BW de todos los enlaces de la red. Utilizando esta medida se observa el hecho de que, al tratarse de un algoritmo de búsqueda local, no se dirige a la solución de una forma directa, sino que se aproxima y se aleja sucesivamente (figura 4) guiado por una

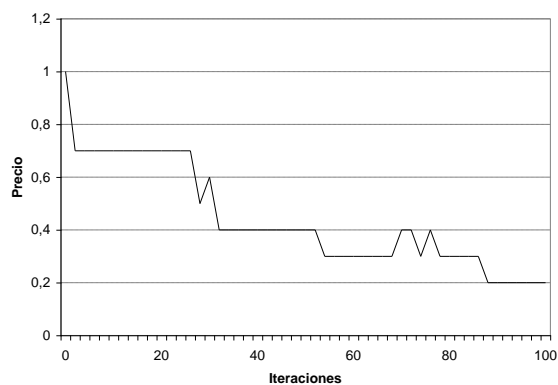


Figura 4: Precio medio del BW de los enlaces de la red

heurística, siendo en este caso una heurística económica.

Examinando esta gráfica se observa que existen periodos en los que el precio medio de la red no se modifica, lo que podría parecer que durante estas iteraciones el sistema no progresa hacia la solución final. Esto sería un grave inconveniente, ya que significaría que el algoritmo está desaprovechando tiempo de cálculo y de comunicaciones. Para comprobar que esto no es cierto es necesario conocer la evolución del porcentaje de utilización de todos los enlaces de la red.

Esta evolución se observa en la figura 5, la cual refleja también el comportamiento global de los consumidores respecto a las reservas de BW en los distintos enlaces de la red. Inicialmente todos los consumidores tienden a reservar BW a través de los enlaces centrales de la red, para conseguir el camino más corto entre el origen y destino de cada VP. Al no poder absorber todas las peticiones a través de estos enlaces tienden a reservar BW a través de otros enlaces con mayor disponibilidad de BW y, consecuentemente, con un precio menor.

Como puede apreciarse, la reserva de ancho de banda de los distintos enlaces se modifica constantemente, aunque realmente depende de los valores de la $Prob_{cambio}$ asignada a cada VP. Para conseguir una buena relación entre la convergencia y la estabilidad del sistema se ha asignado a esta variable valores inversamente proporcionales al número de VPs que gestiona cada consumidor ($NumVPs$), siendo K una constante a determinar, la cual indicará el número medio de cambios que realizará cada consumidor respecto a los VPs que él controla:

$$Prob_{cambio} = \frac{K}{NumVPs}$$

En función de esta constante hemos podido observar empíricamente tres distintos comportamientos del algoritmo:

- Convergencia hacia una solución en un número de iteraciones más que aceptable (figuras 4 y 5), para valores de K entre 1 y 2.

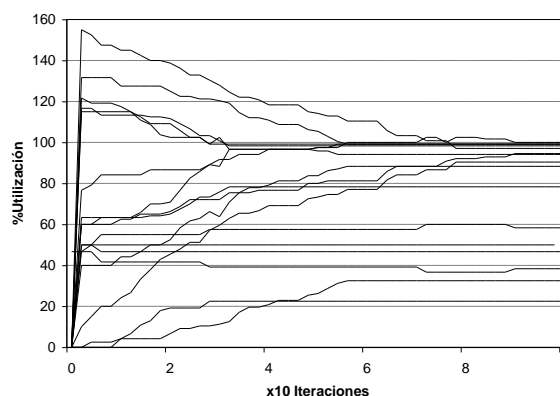


Figura 5: Evolución de la reserva de BW en los enlaces

- Inestabilidad del sistema, debido principalmente a que se producen demasiados cambios en la red en cada iteración (figuras 6 y 7), para valores de K superiores a 2.
- Lentitud de convergencia, debido a que se producen pocos cambios en cada iteración (figuras 8 y 9), para valores de K inferiores a 1.

También es importante conocer como se comporta el algoritmo en el caso de que las demandas de los usuarios estén muy próximas a la capacidad real de la red. Para ello se ha disminuido la capacidad de algunos enlaces de la red, para provocar que la mayoría de enlaces sufran reservas entre el 90 y el 99% de su capacidad.

Como puede observarse en las figuras 10 y 11, el algoritmo vuelve a dar con la solución aunque, como era de preveer, aumenta el número de iteraciones.

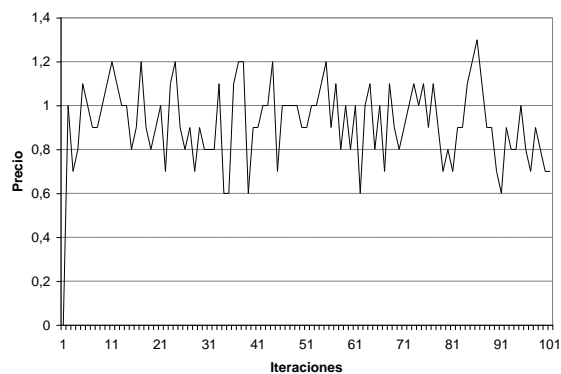


Figura 6: Evolución del precio en un entorno inestable

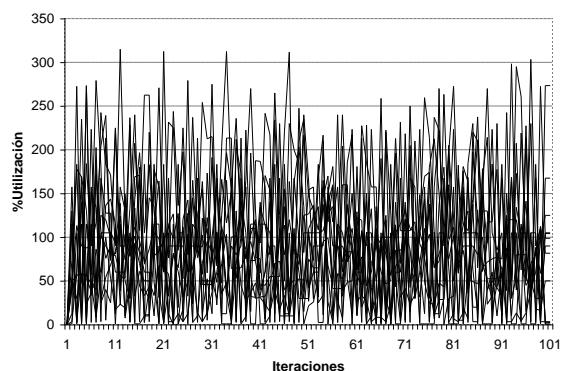


Figura 7: Reserva de BW en un entorno inestable

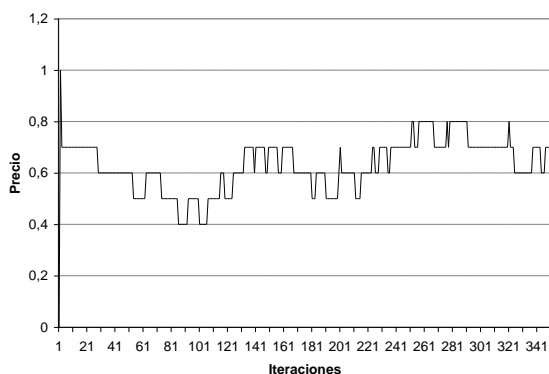


Figura 8: Evolución del precio con lentitud de convergencia

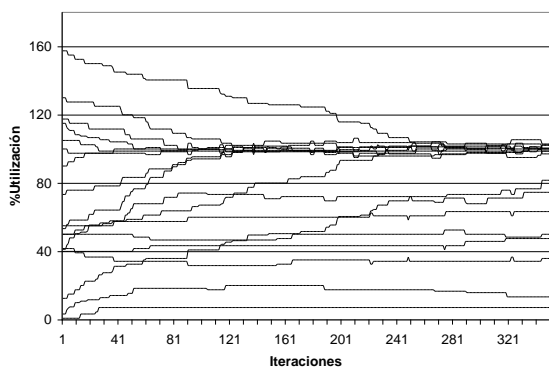


Figura 9: Reserva de BW con lentitud de convergencia

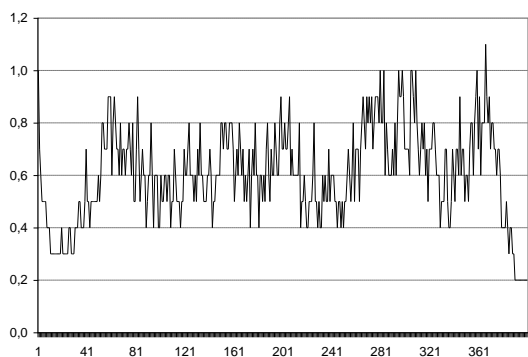


Figura 10: Evolución del precio con saturación en la red

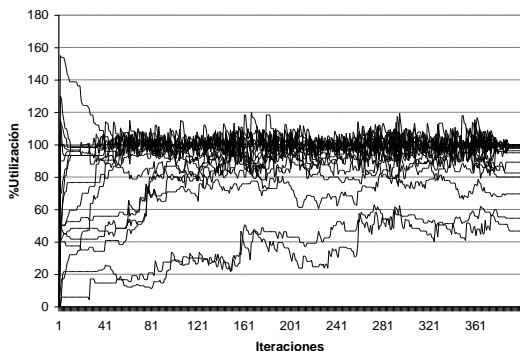


Figura 11: Evolución de la reserva de BW con saturación en la red.

5.2 Coste temporal de la solución

Para tener una idea intuitiva del coste computacional del algoritmo se ha utilizado el término *iteración*. Por iteración se entiende el proceso completo de actualización de precios (cálculo y posterior propagación) y posteriormente de modificación de la ruta de cada VP.

El número de iteraciones necesarias para que el algoritmo llegue a la solución final es variable en cada caso y en cada ejecución. De todas formas, en el caso de que cada consumidor realice entre 1 y 2 cambios de ruta en cada iteración, la convergencia del sistema se produce, en media, en torno a las 100 iteraciones. Este valor es mucho más que aceptable en un entorno distribuido y donde el resultado no deberá obtenerse en tiempo real, ya que los cambios de configuración en la red de VPs se producirán en intervalos de tiempo desde pocas horas a varios días.

6. Conclusiones y líneas futuras

En este artículo se ha presentado un algoritmo distribuido para el diseño de redes de VPs basadas en VPs. Después de un exhaustivo estudio de los trabajos previos existentes se ha optado por una implementación basada en modelos económicos, consiguiendo un algoritmo final con un coste de cálculo reducido, que ocupa un espacio de memoria mínimo y que no sobrecarga la propia red debido a la señalización del mismo. En situaciones de carga normal de la red el algoritmo necesita alrededor de unas 100 iteraciones para encontrar una solución, demostrando su correcto funcionamiento incluso en situaciones de saturación en los enlaces.

Las líneas futuras que deja abiertas este trabajo son muy extensas. En primer lugar es necesario un estudio más detallado del comportamiento del algoritmo en función del parámetro K , así como de otros parámetros que puedan influir en la convergencia del mismo.

Atendiendo a las distintas configuraciones de VPs para las distintas franjas horarias, deberá existir un algoritmo nuevamente distribuido que realice la migración entre topologías de VPs.

Respecto a la supervivencia de los servicios que ofrece la red, es necesario profundizar en el parámetro de “disponibilidad” que requiere cada servicio, estableciendo la relación con el número de VPs redundantes que garantizan la existencia del servicio.

Otro parámetro que no se tiene en cuenta todavía es la *schedulable region* [26], la cual nos determinaría el número de VPs de cada servicio que puede soportar realmente cada nodo de la red. En nuestro estudio hemos tenido en cuenta la capacidad de los enlaces, pero no se ha tenido en cuenta la capacidad

de las colas del nodo a la hora de garantizar la calidad de servicio de todos los VPs que soporta.

Otra restricción que no se tiene en cuenta es el número de VPs que puede soportar una red ATM en cada enlace físico. En cada uno de ellos el formato de la celda ATM en la NNI [27] identifica cada VP con 12 bytes. Esto limita el número de VPs en cada enlace físico a 4096, a los que habrá que eliminar los valores reservados por los estándares. Para evitar que este número sea sobrepasado será necesario combinar tramos comunes de varios VPs en uno de solo, dividiendo los VPs extremo-a-extremo en distintas secciones.

El algoritmo diseñado se basa en una topología PNNI totalmente plana, sin jerarquías de *peer-groups*. Será necesario realizar una implementación recursiva que pueda aplicarse a cualquier esquema jerárquico PNNI.

Finalmente deberán estudiarse las distintas medidas a tomar en el caso de que no se encuentre una solución en un número de iteraciones razonable, implementando, por ejemplo, una eliminación selectiva de servicios en función de las prioridades asignadas.

Referencias

- [1] N.Aneroussis, A.Lazar, "An architecture for controlling Service Demand in ATM Networks based on Pricing Agents". Department of Electrical Engineering and Center for Telecommunications Research. Technical Report.
- [2] R.Fabregat-Gesa, J.Sole-Pareta, J.L.Marzo-Lázaro y J.Domingo-Pascual. "Adaptive Routing Based on Real-Time Calculations using the Convolution Approach." Universitat de Girona. Escola Politècnica Superior. Girona, Catalunya.
- [3] R.Fabregat-Gesa, J.L.Marzo-Lázaro, J.Domingo-Pascual, "Bandwidth Allocation Based on Real Time Calculations Using the Convolution Approach", GLOBECOM'94, pp.788-793.
- [4] A.Lazar, A.Orda, D. Pendarakis, "Virtual Path Allocation in Multiuser Networks: A Noncooperative Approach", Proceeding of the 1995 IEEE INFOCOM, Boston MA, 1995.
- [5] D.Mitra, J.Morrison, "ATM Network Design and Optimization: A Multirate Loss Network Framework", IEEE Transactions on Networking, Vol.4, N°4, Aug.1996
- [6] A.Orda, G.Pacifici, D.Pendarakis, "An Adaptive Virtual Path Allocation Policy for Broadband Networks", Department of Electrical Eng. Technion, Haifa 32000 Israel., IBM Research Division. T.J. Watson Research Center. Yorktown Heights, New York 10598. USA.
- [7] A.Pistillides, J.Lambert, D.Tipper, "Dynamic Bandwidth Allocation in Broadband_ISDN using a Multilevel Optimal Control Approach". IEEE Int.Conf. on Systems Eng., Kobe, Japan, Sept.1992.
- [8] J.Fingerhut, S.Suri, J.Turner. "Designing Minimum Cost Nonblocking Communication Networks", Department of Computer Science. Washington University. St. Louis, USA.
- [9] S.Ahn, R. Tsang, S. Tong and D.Du, "Virtual Path Layout Design on ATM Networks", Proc. INFOCOM '94, March 1994.
- [10] D.Medhi, D.Tipper "Some Approaches to Solving a Multi-Hour Broadband Network Capacity Design Problem with Single-Path Routing". Department of Computer Networking. University of Missouri-Kansas City, Department of Information Science and Telecommunications. University of Pittsburgh.
- [11] D.Medhi, C.Lu. "Dimensioning and Computational Results for Wide-Area Broadband Networks with Two-level Dynamic Routing". IEICE Transactions on Communications.
- [12] K.Murakami, H.Kim., "Virtual Path Routing for Survivable ATM Networks". IEEE/ACM Transactions on Networking. Vol.4, n°.1, Feb. 1996.
- [13] C. Wu, S.Lee "A Generic Optimization Model for Survivable VP Planning in ATM Networks". Department of Electrical Engineering, National Chung Cheng University, Chiayi, Taiwan.
- [14] N.Aneroussis, A. Lazar, "Virtual Path Control for ATM Networks with Call level Quality of Service Guarantees", Proceedings of the 1996 INFOCOM, San Francisco, CA, March 1996.
- [15] O.Gerstel, A.Segall, "Dynamic maintenance of the Virtual Path Layout". LPCR Technical Report 9415, Technion, January 1995.
- [16] K.Sato, S.Ohta, I.Tokizawa, "Broadband ATM Networks Architecture based on Virtual paths", IEEE Transactions on Communications, vol.38, pp-1212-1222, Aug.1990.
- [17] S. Ohta, K. Sato "Dynamic Bandwidth Control of the Virtual Path in an Asynchronous Transfer Mode Network", IEEE Transactions on Communications, Vol.40, n°7, Jul.1992.
- [18] J.Fingerhut, R.Jackson, S.Suri, J.Turner. "Design of Nonblocking ATM Networks", Department of Computer Science. Washington University. St. Louis, WUCS-9603,1/96. USA.
- [19] K.Tang, K.Ko, K.Man, S.Kwong, "Topology Design and Bandwidth Allocation of Embedded ATM Networks Using Genetic Algorithm", IEEE Communications Letters, Vol.2, n°6, pg-171-173, Jun.1998.
- [20] N.Aneroussis, A.Lazar, "A Framework for Pricing Virtual Circuit and Virtual Path Services in ATM Networks". AT&T Research, 600 Mountain Avenue, Murray Hill, NJ 07974-0636.

- [21] J.Murphy, L.Murphy, "*Bandwidth Allocation by Pricing in ATM Networks*". Proc. of IFIP Broadband Communications'94. Paris, France, March 1994.
- [22] D.Ferguson, C.Nikolaou, J.Sairamesh, Y.Yemini, "*Economic Models for Allocating Resources in Computer Systems*". IBM T.J. Watson Research Center, Hawthorne, N.Y 10532.
- [23] The ATM Forum, "*Private Network-Network Interface, Specification Version 1.0*", March 1996.
- [24] L.Murphy, J.Murphy, "*Feedback and Pricing in ATM Networks*". Department of Computer Science and Engineering, Auburn University, AL 36849, USA.
- [25] MIL 3, "OPNET Modeler User Manual", Release 4.0, 1998, <http://www.mil3.com>.
- [26] A.Lazar, A.Orda, D.Pendarakis, "*Virtual Path Allocation in Multiuser Networks: A Noncooperative Approach*", *Proceeding of the 1995 IEEE INFOCOM*, Boston MA, 1995.
- [27] M.de Prycker, "*Asynchronous Transfer Mode, Solutions for Broadband ISDN*", Second Edition, Ellis Horwood, ISBN 0-13-178542-7.

Arquitectura de encaminamiento para servicios garantizados

M.C. Oria, J.A. Lozano, A. Ariza y F. Sandoval
Dpto. de Tecnología Electrónica. E.T.S. Ingenieros de Telecomunicación
Universidad de Málaga, Campus de Teatinos, 29071 Málaga, España
Telf: 95-2132896, Fax: 95-2131447
E-mail: oria@dte.uma.es

Abstract

Real-time multimedia applications support requires that underlying transport networks provide strict Quality-of-Service (QoS) guarantees in terms of bandwidth, delay, jitter and loss probability. These parameters are not granted by the mechanisms implemented in current data networks architectures. In order to support guaranteed services, routing must fulfill multiple restrictions and this problem is NP-complete. However, its complexity can be reduced if the network service disciplines are rate-proportional. This paper proposes a routing architecture that supports guaranteed services and its associated management and control mechanisms. The management architecture is hierarchical and centralised. The network is logically separated into subnetworks. In each subnetwork, a Network Control Center centralises the most complex management and control tasks. It computes the routing tables for all access nodes and maintains the topology and state information. Routing finds feasible paths meeting QoS requirements while making and efficient use of resources and achieving low computational complexity in the nodes.

1. Introducción

En los últimos años está creciendo de forma vertiginosa el número de aplicaciones multimedia. Estas aplicaciones generan flujos continuos de información y exigen que las redes de transmisión les garanticen un conjunto variado de prestaciones muy estrictas, en términos de ancho de banda, retardo y jitter extremo a extremo y tasa de pérdidas. Este conjunto de parámetros define el nivel de calidad que la red debe ofrecer y garantizar a la aplicación, y es lo que se denomina calidad del servicio (QoS: *Quality of Service*).

Las redes de comunicación de datos entre ordenadores más extendidas, basadas en el protocolo IP, funcionan mediante transmisión de datagramas y ofrecen solamente un tipo de servicio de transporte *best-effort*. Los paquetes pertenecientes a una misma sesión son encaminados y compiten por los recursos de transmisión de forma independiente, por lo tanto, experimentan diferentes retardos en las colas de salida y llegan al destino con retardos aleatorios e imprevisibles. Estas redes no pueden ofrecer ninguna garantía de QoS a las aplicaciones y, por lo tanto, no son adecuadas para transportar información multimedia. Las aplicaciones que se construyen sobre ellas son elásticas, el retardo es tolerable y el requisito principal es que la transferencia de información extremo a extremo sea fiable.

Para ofrecer soporte a aplicaciones con requisitos de QoS es necesario, en primer lugar, la definición de un nuevo modelo de servicios que permita a las aplicaciones indicar a la red el nivel de calidad exigido para una conexión y, en segundo lugar, el desarrollo de una arquitectura de red orientada a conexión. Esta arquitectura debe ser un marco para especificar e implementar los mecanismos de gestión y control de recursos necesarios que den

soporte al modelo de servicios [3]. Esta arquitectura debe integrar cuatro mecanismos fundamentales: encaminamiento, control de admisión, reserva de recursos y disciplina de planificación del servicio (*packet scheduling disciplines*) [9][11][14].

El proceso de establecimiento de una conexión en una red orientada a conexión consta de dos fases: la fase de selección de la ruta y la fase de asignación de recursos. En la primera fase, el algoritmo de encaminamiento busca la ruta óptima entre el origen y el destino. En la segunda fase, se realiza el control de admisión (CAC: *Connection Admission Control*), que comprueba si existen suficientes recursos disponibles para cursar la conexión sobre la ruta seleccionada, teniendo en cuenta las características del tráfico y los requisitos de QoS de la conexión. Si la conexión es admitida, se reservan los recursos necesarios en cada nodo de la ruta de forma que se cumpla la QoS extremo a extremo. Durante la transmisión, las disciplinas de planificación del servicio se encargan de asegurar las garantías de QoS a nivel de paquetes individuales en cada puerto de salida.

El mecanismo de encaminamiento debe seleccionar la ruta óptima para cada conexión teniendo en cuenta dos objetivos fundamentales: cumplir los requisitos de QoS de cada conexión y realizar un uso eficiente de los recursos de la red, que se traduce en maximizar el tráfico cursado (*throughput*) o minimizar la tasa de bloqueo de conexiones.

Los algoritmos de encaminamiento que tienen en cuenta múltiples restricciones o parámetros de QoS, como ancho de banda, retardo y probabilidad de pérdidas, tienen un coste computacional demasiado alto para ser usados en la práctica. Está demostrado que la búsqueda de un camino que minimice simultáneamente dos o más métricas aditivas (retardo, jitter, coste o número de saltos) y/o

multiplicativas (pérdidas) es un problema NP-completo [15]. Sin embargo, este resultado ha sido obtenido sin tener en cuenta la relación entre el encaminamiento y los mecanismos de gestión de colas. Ciertas disciplinas de planificación del servicio en las colas reducen la complejidad del algoritmo de encaminamiento, ya que introducen dependencias entre los parámetros de QoS [9][11].

En este artículo se estudia una arquitectura de encaminamiento para redes de servicios integrados que soporten conexiones garantizadas. En el apartado 2 se propone la arquitectura de encaminamiento, el esquema de gestión centralizado y los protocolos de intercambio de información. El apartado 3 introduce las disciplinas de servicio. En el apartado 4 se explica el mecanismo de encaminamiento orientado a QoS y la relación que guarda con las disciplinas de servicio. El modelo de simulación y los resultados se describen en los apartados 5 y 6. Por último, se presentan las conclusiones.

2. Arquitectura de encaminamiento

La arquitectura de encaminamiento especifica todos los aspectos relacionados con la búsqueda de rutas para establecer conexiones y la gestión asociada [8]: estructura lógica de la red (plana o jerárquica), tipo de protocolo de encaminamiento (basado en estado de los enlaces o vector de distancia), algoritmo de encaminamiento (en el origen o *hop by hop*) y los protocolos de intercambio de información si se trata de un mecanismo dinámico.

Un mecanismo de encaminamiento que soporte QoS debe seleccionar la ruta óptima de forma que se cumplan los requisitos de cada conexión a la vez que se realice un uso eficiente de los recursos. Cuando se trata de una red multiservicio, habría que añadir un tercer objetivo: que el mecanismo de encaminamiento no suponga una carga computacional muy alta en los nodos, lo que supondría un retardo en el establecimiento de la conexión que puede llegar a ser considerable. Muy pocos trabajos tienen en cuenta la complejidad de cálculo del encaminamiento [16], a pesar de su importancia en redes de servicios integrados. A continuación se van a estudiar los aspectos de la arquitectura de encaminamiento que influyen en el soporte de la QoS.

Por una parte, los mecanismos de encaminamiento estáticos no son adecuados para soportar QoS, ya que, al no tener en cuenta el estado de la red, no utilizan de forma eficiente los recursos. Los algoritmos de encaminamiento dinámicos tienen en cuenta el estado de la red y, por lo tanto, seleccionan el camino óptimo de forma que se haga un uso eficiente de los recursos y evitando, a la vez, la congestión. Los distintos algoritmos de encaminamiento dinámicos intentan maximizar el tráfico cursado empleando diversos criterios de selección del camino óptimo [1][5][6][7]. La elección de un criterio u otro supone dar mayor

relevancia a uno de los dos objetivos fundamentales de eficiencia: economizar el consumo de recursos y tratar de distribuir uniformemente la carga en la red. Como contrapartida, el mantenimiento y actualización de la información de estado plantea algunos problemas.

Por otra parte, si se utiliza un algoritmo de encaminamiento en el que cada nodo selecciona el siguiente enlace en el camino al destino (encaminamiento *hop by hop*), el nodo origen no dispone de la ruta completa, por tanto, no hay forma de conocer a priori si la conexión va a ser rechazada por algún nodo intermedio o si se va a poder ofrecer la QoS solicitada. Esto supondría muchos intentos de conexión rechazados en nodos intermedios y una baja eficiencia en la elección de los caminos. Este es el mecanismo de encaminamiento utilizado en redes de datos distribuidas, como Internet. Por el contrario, un algoritmo de encaminamiento en el que el nodo origen seleccione la ruta completa al destino (encaminamiento en el origen) presenta mayores prestaciones a la hora de ofrecer garantías de QoS extremo a extremo. La ruta puede ser calculada bajo demanda por el nodo origen o estar almacenada en una tabla de rutas.

En resumen, para garantizar la QoS la solución más eficiente es un algoritmo de encaminamiento en el origen que tenga en cuenta los requisitos de QoS y el estado de los recursos. Un algoritmo con estas características presenta dos problemas fundamentales:

1.- Los algoritmos de cálculo de caminos que tienen en cuenta múltiples restricciones suelen ser muy complejos y suponen una carga de procesamiento elevada en los nodos y, por tanto, un mayor retardo en el establecimiento de las conexiones.

2.- Cada nodo debe mantener una base de datos con el estado de toda la red que le permita conocer en cada momento las prestaciones de QoS que ofrece cada enlace y cada nodo. Esto supone que cada nodo debe enviar paquetes de información al resto de los nodos de la red. La actualización de los datos se puede realizar de forma periódica o cada vez que se produzca un cambio en la ocupación de algún recurso, y lo más habitual es utilizar un mecanismo de inundación (*flooding*), que puede generar una carga elevada de tráfico en la red. Por otra parte, las tablas que almacenan el estado de cada nodo pueden llegar a tener un tamaño considerable. La escalabilidad de esta arquitectura queda limitada por ambos aspectos.

Como solución a los problemas anteriores, este artículo propone una arquitectura de encaminamiento jerárquica¹ y centralizada para soportar conexiones con garantías de QoS. La red se

¹ El *ATM Forum* también adopta una arquitectura de red jerárquica [2] para poder aplicar el encaminamiento a redes de gran tamaño.

divide en dominios y en cada uno de ellos hay un Centro de Control de Red (CCR) o nodo "gestor" que realiza las funciones de gestión y de control de red más complejas. Entre otras funciones, calcula las rutas óptimas para todos los nodos del dominio. De esta forma, los nodos quedan liberados de funciones como el encaminamiento. El CCR se comunica con cada uno de los nodos del dominio y con el resto de CCR de su mismo nivel mediante la red de gestión.

El mecanismo de encaminamiento se basa en la utilización de tablas de rutas en cada nodo de acceso. Las tablas son calculadas por el CCR del dominio y contienen el mejor camino a cada destino, junto con cierta información de prestaciones asociada (ancho de banda residual y parte fija del retardo). El nodo origen es el responsable de los mecanismos de control para el establecimiento y liberación de conexiones de forma que se cumplan los requisitos de QoS.

De esta forma, la parte más compleja del mecanismo de encaminamiento se ubica en el CCR, encargado de calcular las tablas de encaminamiento y el mantenimiento de dichas tablas de acuerdo con la información de estado que recibe. Para ello, el CCR debe mantener una base de datos con información de los recursos que forman el dominio. La información que recibe el CCR es de dos tipos:

- Información topológica. En la fase de inicialización de la red, cada nodo intercambia mensajes de reconocimiento con los nodos vecinos para obtener información de conectividad y, una vez que recoge esta información, la envía al CCR. De esta forma el CCR recopila la información topológica que necesita.
- Información de estado. Durante el funcionamiento normal de la red, los nodos envían al CCR paquetes de estado periódicamente o cuando se produce un cambio de estado significativo en la ocupación de alguno de los enlaces de salida, cambio provocado por la reserva o liberación de los recursos asociados a una conexión.

Además de la información de gestión que circula por la red para mantener las tablas de estado del gestor y las tablas de rutas, también se intercambia información de control: señalización de red y señalización entre el usuario y el nodo de acceso.

La Fig.1 muestra donde se localizan las interfaces de gestión y control que se han definido: interfaz U, entre el *host* generador de tráfico y el nodo de acceso a la red, interfaz R, entre dos nodos de la red, e interfaz I, entre cada nodo y el CCR. Sobre estas interfaces se definen cuatro protocolos de comunicación:

- Protocolo de inicialización. En la fase de inicialización de la red se produce un intercambio de información en la interfaz R y la

interfaz U que permite a cada nodo obtener información de conectividad.

- Protocolo de información. Define la secuencia de mensajes intercambiados en la interfaz I. Mediante estos mensajes el CCR recibe información topológica y de estado de los recursos de la red y los nodos reciben las tablas de rutas actualizadas.
- Protocolo de señalización en la interfaz U. Define el formato, secuencia y significado de los mensajes intercambiados entre un *host* generador y el nodo de acceso: mensaje de petición de establecimiento de una conexión a un destino particular², mensaje de liberación, aceptación y rechazo de conexión.
- Protocolo de señalización de red (interfaz R). Define el formato, secuencia y significado de los mensajes intercambiados entre nodos de la red para establecer y liberar conexiones de red. Los mensajes de señalización en la interfaz U se mapean en mensajes del protocolo de nivel de red.

Cuando llega una solicitud de conexión a un nodo de acceso, se inicia el proceso de establecimiento que consta de dos fases: selección del camino y establecimiento del estado de conexión en cada nodo del camino.

En primer lugar, el nodo de acceso selecciona de su tabla de encaminamiento la ruta completa hasta el destino. A continuación, el mecanismo de control de admisión genérico (GCAC: *Generic CAC*) [2] compara la disponibilidad de recursos sobre el camino completo con los que requiere la conexión. Para ello tiene en cuenta, por una parte, las características de prestaciones asociadas al camino, y por otra, la caracterización del tráfico y los requisitos de QoS. La información que tiene almacenada el nodo origen en su tabla y que utiliza el GCAC es aproximada, debido a la latencia en el intercambio de información de estado. El estado de ocupación de algunos enlaces puede haber cambiado desde la última vez que el CCR recibió una notificación. No obstante, se ha comprobado que el mecanismo de GCAC asegura una probabilidad muy elevada de que la ruta seleccionada tenga realmente recursos disponibles en cada nodo del camino.

A continuación, si se acepta la conexión, el nodo origen se encarga de enviar el mensaje de petición adecuado a través de la señalización para establecer la conexión sobre la sucesión de nodos y enlaces que forman el camino. Cuando un nodo del camino recibe el mensaje de petición de conexión, el control de admisión local (CAC) comprueba si hay suficientes recursos disponibles para garantizar los requisitos de QoS. Si la conexión es admitida, se realiza la reserva de recursos y se envía el mensaje de petición al siguiente nodo. El CAC local es

² El mensaje de petición de conexión incluye la especificación de tráfico y los parámetros de QoS: ancho de banda, tamaño de ráfaga, retardo y jitter.

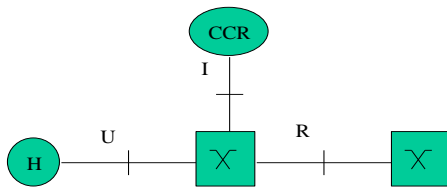


Figura 1. Localización de interfaces

necesario puesto que la información exacta y actualizada de la ocupación de los enlaces de salida sólo la conoce el propio nodo.

Cuando el mensaje de petición alcanza el nodo destino y todos los recursos han sido reservados, la conexión se considera establecida y para indicarlo, se envía un mensaje de aceptación en dirección al nodo origen. Por el contrario, si la conexión no supera el CAC en algún nodo del camino, se envía un mensaje de rechazo en dirección al nodo origen para que los nodos anteriores en el camino liberen los recursos asignados a la conexión. Cuando el mensaje de aceptación o rechazo llega al nodo origen, se envía el mensaje de notificación correspondiente al *host* que generó la conexión.

Para liberar una conexión, el *host* envía al nodo de acceso un mensaje de liberación. El nodo de acceso envía el mensaje correspondiente de señalización de red a través del camino. El mensaje de liberación se propaga de nodo a nodo liberando los recursos reservados.

3. Disciplina de servicio

A la hora de diseñar el algoritmo de encaminamiento es muy importante tener en cuenta la disciplina de servicio, que se encarga de clasificar las celdas de las conexiones en las colas de salida y decidir el orden y tiempo de servicio de cada cola. La disciplina de servicio introduce dependencias entre los parámetros de QoS y, por lo tanto, reduce la complejidad del algoritmo de encaminamiento [9][11].

La RFC2212 del IETF [12] recomienda el uso de una disciplina del tipo *rate-proportional*[17]. Este tipo de disciplinas son aplicaciones prácticas de la disciplina ideal, GPS (*Generalized Processor Sharing*), y las principales son: WFQ (*Weighted Fair Queueing*) [4][10], WF2Q (*Worst-Case Weighted Fair Queueing*), VC (*Virtual Clock*), SCFQ (*Self Clocked Weighted Fair Queueing*) y FBFQ (*Frame based Fair Queueing*) [13]. Todas estas disciplinas aseguran que las conexiones que comparten un enlace de salida reciben la parte proporcional que les corresponde de la capacidad del enlace, suponiendo que cada conexión tenga reservado un espacio de buffer independiente. En consecuencia, los parámetros de QoS de una conexión son función del ancho de banda reservado y las características del tráfico que genera.

Para caracterizar el tráfico suponemos que cada conexión o cada fuente de tráfico está controlada por un *leaky bucket*, cuyos parámetros (σ, b) caracterizan el régimen binario medio y el tamaño máximo de

ráfaga, respectivamente. Suponiendo una disciplina de servicio tipo *rate-proportional*, para un camino p compuesto por n enlaces se pueden calcular el retardo, $D(p, r, b)$, y jitter, $J(p, r, b)$, extremo a extremo, así como el tamaño de buffer necesario en cada nodo del camino para que no se produzcan pérdidas de celdas, $B(p, b, h)$, mediante las fórmulas siguientes:

dónde r es la cantidad de ancho de banda que se reservará al flujo y debe cumplir $r > \sigma$. L es el tamaño máximo de paquete en la red. C_i es la capacidad total del enlace i -ésimo de la red y *propi*, el retardo de propagación.

4. Mecanismo de encaminamiento

El control de encaminamiento tiene un doble objetivo: seleccionar caminos que cumplan las restricciones de QoS de las conexiones, y hacerlo utilizando de la forma más eficiente los recursos de la red. Para que un camino sea válido para cursar cierta conexión garantizada será necesario que el camino P seleccionado cumpla las restricciones de ancho de banda, retardo y jitter que, suponiendo una disciplina tipo *rate proportional*, se traducen en las

$$D(p, r, b) = \frac{b + nL}{r} + \sum_{i=1}^n (L/C_i + \text{propi}) \quad [1]$$

$$J(p, r, b) = \frac{b + nL}{r} \quad [2]$$

$$B(p, b, h) = b + hL \quad [3]$$

condiciones siguientes:

Para realizar un uso eficiente de la red, la selección de caminos debe minimizar la utilización de recursos, a la vez que evitar la aparición de congestión, distribuyendo de forma adecuada la carga de tráfico y evitando recursos muy saturados. Dependiendo del criterio de optimización que usemos para seleccionar los caminos podemos distinguir dos categorías de algoritmos de encaminamiento:

1. Algoritmos que dan prioridad a minimizar la utilización de recursos. Podemos señalar:

- **Camino más corto.** Selecciona el camino con menor número de enlaces.
- **Widest-shortest.** Selecciona el camino con menor número de enlaces entre el origen y el destino, en caso de igualdad elige el que

$$B(P) = \min_{i \in P} (b_i) \leq r \leq \sigma \quad [4]$$

$$D(P) \leq Dreq \quad [5]$$

$$J(P) \leq Jreq \quad [6]$$

presente mayor ancho de banda libre.

2. Algoritmos cuya prioridad es distribuir uniformemente la carga en la red. Podemos señalar:

- **Camino menos cargado:** Selecciona el camino que presenta un mayor ancho de banda libre.

- **Shortest-widest.** Selecciona el camino que presenta mayor ancho de banda libre entre un nodo origen y un nodo destino, en caso de igualdad, elige el más corto.

En este artículo se propone un mecanismo de cálculo de los caminos basado en una modificación del algoritmo *Widest-shortest*. El cálculo de los caminos se realiza en el CCR según el procedimiento siguiente:

- 1.- Sea $R_{O-D}=\{R^i_{O-D}\}$ el conjunto de rutas que presentan el número mínimo de enlaces y número mínimo de enlaces más uno, para cada par de nodos origen-destino. Estos conjuntos se calculan únicamente en la fase de inicialización o cuando se producen cambios en la topología de la red.
- 2.- Para cada R_{O-D} , el CCR selecciona la ruta que presenta mayor ancho de banda residual. Cada vez que el gestor recibe información de estado de algún recurso, calcula de nuevo el ancho de banda residual de las rutas afectadas y si es necesario, envía rutas actualizadas a los nodos afectados.

Se van a comparar las prestaciones del mecanismo de encaminamiento basado en tabla de rutas combinado con tres mecanismos de gestión, que se diferencian básicamente en el momento en que los nodos envían información de estado al CCR:

- **Mecanismo dinámico periódico.** Los nodos envían periódicamente al CCR información de ocupación de sus enlaces. A mayor frecuencia de envío de información de estado, el CCR contará con una información más exacta, pero la carga de tráfico en los enlaces de gestión será mayor.
- **Mecanismo dinámico no periódico.** Los nodos envían información de ocupación cuando se produce un cambio significativo en la ocupación de algún enlace. Para ello, se establecen ciertos umbrales de ocupación y cada vez que la ocupación traspasa un umbral, se envía información de estado al CCR. Si establecemos

más umbrales, el CCR tendrá almacenada en su base de datos una información más exacta, pero la carga de tráfico en la red de gestión será mayor.

- **Mecanismo estático.** Es el caso de un algoritmo de encaminamiento basado en tabla fija de rutas que almacena los caminos más cortos en número de enlaces.

5. Modelo de simulación

Se ha simulado la arquitectura de red y los mecanismos de encaminamiento mediante Opnet, una herramienta de desarrollo y simulación jerárquico orientada a objetos. Se ha realizado una simulación mixta: las conexiones se simulan a nivel de sesión, no generan paquetes, sin embargo, se transmiten mensajes de señalización y de información para simular los protocolos de control y gestión en las tres interfaces (U, R e I).

5.1. Modelo de red

Se ha simulado el modelo de red MCI de Internet (Fig.2). Los enlaces son todos *full-dúplex*. El CCR o nodo gestor se supone conectado directamente con todos los nodos mediante enlaces de gestión. A cada nodo se conecta un *host* generador de tráfico.

Todo el tráfico ofrecido corresponde a conexiones garantizadas. Suponemos que las conexiones llegan de acuerdo con una tasa exponencial y son bidireccionales y simétricas, esto es, demandan el mismo ancho de banda en los dos sentidos. El tiempo de duración de sesiones está distribuido exponencialmente con una media de 20 minutos. La carga de tráfico ofrecida a la red es función del tiempo medio de duración de la conexión, del régimen binario de las conexiones y la tasa de llegadas. La tasa de tráfico es distinta para cada par de nodos origen-destino, por tanto, el tráfico en la red está irregularmente distribuido. En esta situación se muestran más claramente las ventajas de los algoritmos dinámicos sobre los estáticos.

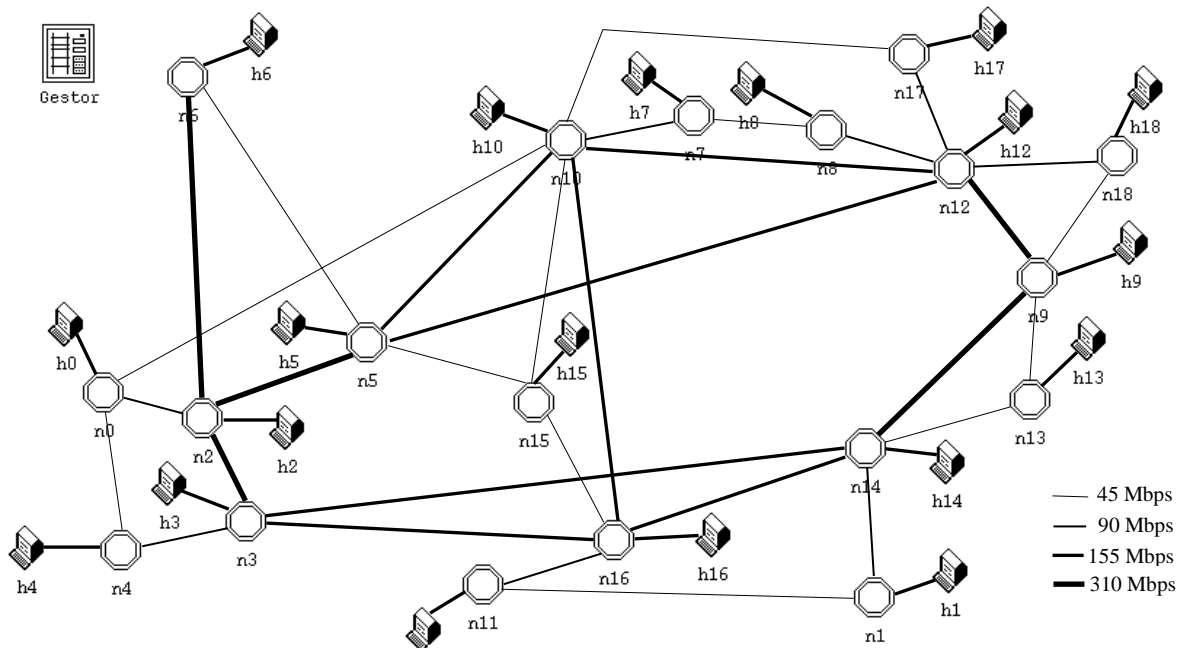


Figura 2. Modelo de red

El tráfico de cada conexión se especifica mediante los parámetros del *leaky bucket* (σ, b) , que se seleccionan como sigue: el régimen binario medio de la conexión, σ , está uniformemente distribuido entre 1 y 5 Mbps. El tamaño máximo de ráfaga, b , está uniformemente distribuido con igual probabilidad en uno de los intervalos: [4KB, 8KB] y [16KB, 20KB]. Estas distribuciones modelan una combinación de video de baja y alta calidad [9].

La *QoS* solicitada por una conexión se expresa en términos temporales: retardo y jitter máximos tolerables. El límite superior del retardo está uniformemente distribuido en el intervalo [80ms, 120ms] o, con igual probabilidad, en el intervalo [200ms, 240ms]. El jitter está uniformemente distribuido entre 20ms y 50ms. Cuanto más estrictos sean los límites de retardo o jitter, mayor será el ancho de banda que se deberá asignar a la conexión para garantizar la *QoS*. No tener en cuenta el requisito de tasa de pérdidas de celdas (CLR: *cell loss rate*) equivale a suponer que los nodos tienen un espacio de *buffer* que podemos considerar prácticamente infinito³. Se consideran cuatro subtipos de conexiones garantizadas, atendiendo al tamaño de ráfaga y al retardo requerido.

5.2. Medidas de prestaciones

La principal medida de eficiencia de un algoritmo de encaminamiento es la tasa de bloqueo de conexiones, es decir, la relación entre número de conexiones rechazadas y número total de conexiones ofrecidas a la red. Sin embargo, en una red en la que se generan conexiones con diferentes requisitos de *QoS*, la tasa de bloqueo global no proporciona suficiente información acerca de la eficiencia del mecanismo de encaminamiento, sería conveniente conocer la tasa de bloqueo específica de cada tipo o subtipo.

Por otra parte, una conexión garantizada puede ser rechazada bien porque el mecanismo de encaminamiento no encuentre un camino con suficientes recursos para cumplir la *QoS* solicitada o bien porque la ocupación de algún enlace del camino haya variado desde la última vez que se notificó al CCR. En el segundo caso, se puede considerar que el nodo de acceso ha seleccionado una ruta incorrecta. Si la relación entre el número de selecciones incorrectas y el número de conexiones rechazadas es muy alta, la información de estado de que disponen el encaminamiento y el GCAC no está suficientemente actualizada.

6. Resultados

Sobre el modelo de red se comparan diferentes mecanismos de encaminamiento basados en tabla de rutas que utilizan la variante del algoritmo *widest-shortest*: actualización periódica de tablas cada 30s o 60s y actualización basada en un mecanismo de umbrales de ocupación. Se compara un mecanismo que utiliza 3 umbrales, 50%, 75% y 90%, con otro que utiliza umbrales con separación exponencial y con histéresis. Si la ocupación crece, los umbrales son: 18%, 42%, 62%, 77,5% y 90%, pero si la ocupación decrece, los umbrales se sitúan en el punto medio de los intervalos anteriores.

En la tabla 1 se puede observar la tasa de bloqueo de conexiones que se produce con cada uno de los mecanismos en función del tráfico medio ofrecido en MBytesps. Para todos los tráficos, los mecanismos dinámicos presentan un funcionamiento muy superior. El mecanismo periódico con $T=30s$ presenta la menor tasa de bloqueo, seguido por el dinámico con 3 umbrales y el periódico con $T=60s$ y, por último, el basado en umbrales exponenciales, aunque las diferencias no son significativas.

³ El límite de CLR depende exclusivamente del espacio de *buffer*.

Tabla 1. Bloqueo de conexiones (%)

Tráfico medio (MBps)	152	228	266	304
Estático	9,1	19,3	24,2	28,5
Dinámico 3 umbrales	1,5	10,4	15,8	21,1
D. umbrales exp.	1,7	11,2	16,7	22,0
D. Periódico (T=60s)	1,5	10,5	15,9	21,3
D. Periódico (T=30s)	1,3	10,0	15,3	20,5

Las tablas 2 a 5 presentan el porcentaje de bloqueo de conexiones por subtipos en función del tráfico medio ofrecido en MBytesps. De nuevo, en todos los casos, las tasas más bajas de bloqueo corresponden a los mecanismos dinámicos. Se puede observar como las conexiones de los subtipos 2 y 3, que presentan un rafagueo mayor, experimentan una mayor tasa de bloqueo que las correspondientes a los subtipos 0 y 1. Sin embargo, las diferencias de bloqueo no son significativas respecto al parámetro retardo.

Tabla 2. Bloqueo conexiones subtipo 0 (%)

(Tamaño max ráfaga [8KB,4KB] y retardo [80ms,120ms])

Tráfico medio (MBps)	152	228	266	304
Estático	8,6	17,9	22,7	27,0
Dinámico 3 umbrales	1,4	9,8	14,7	19,7
D. umbrales exp.	1,6	10,4	15,5	21,0
D. Periódico (T=60)	1,2	9,7	14,8	19,5
D. Periódico (T=30)	1,2	9,4	14,2	18,9

Tabla 3. Bloqueo conexiones subtipo 1 (%)

(Tamaño max ráfaga [8KB,4KB] y retardo [200ms,240ms])

Tráfico medio (MBps)	152	228	266	304
Estático	8,7	18,1	22,9	26,8
Dinámico 3 umbrales	1,3	9,4	14,6	19,7
D. umbrales exp.	1,7	10,0	15,4	20,5
D. Periódico (T=60)	1,4	9,6	14,6	19,5
D. Periódico (T=30)	1,2	8,9	14,1	18,9

Tabla 4. Bloqueo conexiones subtipo 2 (%)

(Tamaño max ráfaga [16KB,20KB] y retardo [80ms,120ms])

Tráfico medio (MBps)	152	228	266	304
Estático	9,5	20,6	25,6	30
Dinámico 3 umbrales	1,5	11,3	16,8	22,5
D. umbrales exp.	1,8	12,0	18,1	23,7
D. Periódico (T=60)	1,6	11,3	17,0	22,9
D. Periódico (T=30)	1,5	10,8	16,5	22,1

Tabla 5. Bloqueo conexiones subtipo 3 (%)

(Tamaño max ráfaga [16KB,20KB] y retardo [200ms,240ms])

Tráfico medio (MBps)	152	228	266	304
Estático	9,8	20,6	25,3	30,2
Dinámico 3 umbrales	1,6	11,1	17,1	22,5
D. umbrales exp.	1,8	12,2	18,0	23,8
D. Periódico (T=60s)	1,7	11,4	17,2	23,0
D. Periódico (T=30s)	1,5	11,1	16,5	22,1

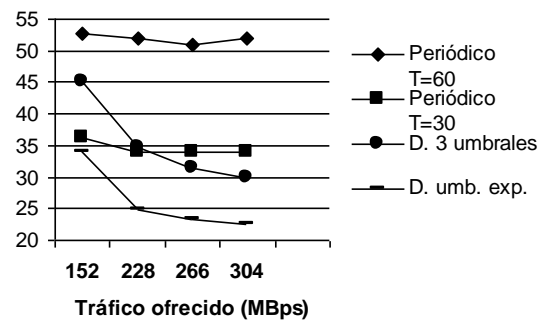
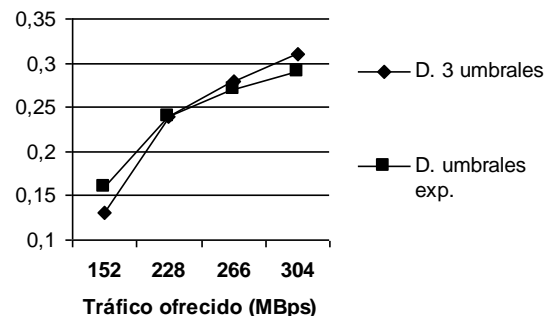
Los resultados de la Fig. 3 corresponden al porcentaje de selecciones incorrectas de rutas respecto al número total de conexiones rechazadas. Se considera que se ha producido una selección incorrecta cuando el encaminamiento encuentra un camino adecuado para la conexión, pero la conexión es rechazada posteriormente por el CAC de algún nodo intermedio de la ruta. Se puede observar como el algoritmo dinámico con actualización cada T=60s presenta un número de selecciones incorrectas por encima del 50%, lo que indica que la información de estado en el CCR y las tablas de encaminamiento no están suficientemente actualizadas. Para los

mecanismos dinámicos el número de selecciones incorrectas decrece a medida que aumenta la carga, en contraste con los periódicos, para los que permanece prácticamente constante, esto se debe a que los umbrales no están uniformemente distribuidos, la distancia es menor a medida que la ocupación crece y, por tanto, para cargas altas, la información de estado será más exacta.

Por último, la Fig. 4 compara el número de mensajes de notificación que recibe el CCR por segundo utilizando los dos mecanismos basados en umbrales. Este número es constante para los mecanismos periódicos: 19 y 9,5 mensajes por segundo para T=60s y T=30s, respectivamente. Se puede observar como los mecanismos basados en umbrales necesitan enviar un número muy inferior de mensajes de notificación que los mecanismos dinámicos periódicos para obtener un rendimiento similar en todos los casos de carga estudiados.

7. Conclusiones

En este artículo se propone una arquitectura de encaminamiento para redes que soporten conexiones con requisitos de QoS, junto con una arquitectura de gestión jerárquica y centralizada. La red se divide en dominios, y en cada uno de ellos un centro de control de red (CCR) realiza las tareas más complejas de gestión y control, entre ellas, el cálculo de las tablas de rutas. De esta forma, el procesamiento que realizan los nodos es menor: encaminan las conexiones de acuerdo con la tabla de rutas y únicamente mantienen información de los recursos locales. El CCR se encarga de actualizar las tablas de rutas teniendo en cuenta la información de

**Figura 3.** Selecciones incorrectas (%)**Figura 4.** Mensajes de notificación cada 100s

estado que le envían los nodos.

Se ha simulado un mecanismo de encaminamiento estático, basado en el camino más corto, y cuatro dinámicos, periódicos, con distintos periodos de actualización, y no periódicos, basados en un mecanismo de umbrales. Los mecanismos dinámicos basados en umbrales presentan la mayor eficiencia, la probabilidad de bloqueo es similar a la de los mecanismos periódicos, pero la carga en la red de gestión es mucho menor.

El modelo de red presentado en este artículo es un punto de partida para estudios futuros:

- Estudio exhaustivo de los mecanismos basados en umbrales, en concreto, comparación de las prestaciones obtenidas, en tasa de bloqueo, y carga que los mensajes de estado producen en la red de gestión.
- Comparación de la eficiencia de red alcanzada con otros algoritmos de encaminamiento.
- Implementación de otros tipos de tráfico e integración en el modelo de red. Estudio de la influencia de la carga de unos tipos de tráfico sobre la probabilidad de bloqueo de los otros.
- Utilización de caminos virtuales para transportar distintos tipos de tráfico y simplificar los mecanismos de control de red. Desarrollo de algoritmos de gestión dinámica de ancho de banda, medida y comparación de las prestaciones obtenidas.

Agradecimientos

Este trabajo ha sido parcialmente financiado por la Comisión Interministerial de Ciencia y Tecnología (CICYT), proyecto n. TIC96-0743.

Referencias

[1] G.R. Ash. "Dynamic Routing in Telecommunications Networks". McGraw Hill Telecommunications, 1998.

[2] ATM Forum Technical Committee. "Private Network Network Interface. Specification Version 1.0 (PNNI 1.0)". URL <http://www.atmforum.com/atmforum/specorder.html>. Marzo 1996. af-pnni-0055.000.

[3] R. Braden, D. Clark, S. Shenker, "Integrated Services in the Internet Architecture: an Overview". Junio 1994. IETF RFC 1633.

[4] S. Keshav, A. Demers, S. Shenker. "Analysis and Simulation of a Fair Queueing Algorithm". ACM SIGCOMM. 19(4): 2-12. 1989.

[5] R.J. Gibbens. "Dynamic Routing in Circuit-Switching Networks: The Dynamic Alternative Routing Strategie". PhD thesis, University of Cambridge. Julio 1988.

[6] S. Gupta, K.W. Ross, M. El Zarki. "Routing in Communications Networks, M. Streenstrup" Cap: "On Routing in ATM Networks", pp. 49-74, Prentice-Hall, 1995.

[7] K.R. Krishnan, "Adaptative State-Dependent Traffic Routing Using On-Line Trunk-Group Measurements". Proc of ITC-13, pp. 407-411. Copenhagen 1991.

[8] W.C. Lee, M.G. Hluchyj, P.A. Humblet. "Routing Subject to Quality of Service Constraints in Integrated Communications Networks". IEEE Network, pp. 46-55. Julio/Agosto 1995.

[9] Q. Ma. "Quality of Service Routing in Integrated Services Networks". PhD thesis, Carnegie Mellon University, Pittsburg. Enero 1998.

[10] A. Parekh, R. G. Gallager, "A Generalised Processor Sharing Approach to Flow Control in Integrated Service Networks - The Multiple Node Case". Proc. IEEE INFOCOM'93, pp. 521-530. San Francisco, USA, April 1993.

[11] C. Pournavalai, G. Chakraborty, N. Shiratori, "A New Distributed QoS Routing Algorithm for supporting Real Time Communications in High Speed Networks". IEICE Transactions on Communications, (10):1493-1501. Octubre 1997.

[12] S. Shenker, C. Partridge, R. Guerin. "Specification of the Guaranteed Quality of Service". Septiembre 1997. IETF RFC 2212.

[13] D. Stiliadis, A. Varma. "Efficient Fair Queueing Algorithms for Packet Switched Networks". IEEE/ACM Transactions on Networking, 6(2):175-185. Abril 1998.

[14] S. Verma, R.K. Pankaj, A. León-García. "Interaction between QoS Routing and Resource Reservation". Proceedings INFOCOM'98. San Francisco. Marzo 1998.

[15] Z. Wang, J. Crowcroft. "Quality of Service Routing for Supporting Multimedia Applications". IEEE Journal on Selected Areas in Communications, 14(7): 1228-1234. Septiembre 1996.

[16] G.G.Xie, D. Hensgen, T.Kidd, J.Yarger. "SAAM: An Integrated Network Architecture for Integrated Services". Sixth International Workshop on QoS, IWQOS'98. Napa, California, USA, pp. 117-126. 18-20 Mayo 1998.

[17] H. Zhang. "Service Disciplines for Guaranteed Performance Service in Packet Switch Networks". Proceedings of the IEEE, 83(10). Octubre 1995.

Análisis comparativo de modelos dinámicos de control de admisión en redes ATM

Manuel Fernández Veiga Cándido López García
Área de Ingeniería Telemática. Universidad de Vigo
Campus universitario s/n, 36200 Vigo (PONTEVEDRA)
Telf:986-812249 Fax: 986-812216
E-mail: {mveiga,candido}@ait.uvigo.es

Abstract

Quality of service guarantees are an important issue in ATM network design. In this paper, we consider call admission control schemes to an ATM link fed by heterogeneous ON-OFF traffic sources, and compare two different simple models of dynamic admission control methods. Whereas the first model has full knowledge of traffic parameters, the second one considers uncertainty in the traffic process. A performance evaluation study of both methods is conducted, focusing on the maximum achievable region and the feasibility of call admission strategies. Besides this, two performance criteria, namely cell loss probability and call blocking probability, are simultaneously considered.

1. Introducción

Un aspecto determinante en la ingeniería de las modernas redes de comunicaciones es el análisis y diseño de métodos simples y efectivos de control de tráfico capaces de proporcionar garantías en los parámetros de calidad de servicio. Es un hecho bien conocido que cuando el tráfico presenta características estadísticas homogéneas (conocidas) y la red debe ofrecer, básicamente, un solo tipo de servicio, la síntesis de estos métodos de control de tráfico es relativamente simple y directa.

Muy al contrario, en una red como ATM (*Asynchronous Transfer Mode*), concebida para la integración de diferentes clases de servicio y el transporte de tráfico con propiedades estadísticas heterogéneas y desconocidas[1], la consecución de una utilización elevada de los elementos de la red es un objetivo contrapuesto a la consecución de garantías en los parámetros de prestaciones. En el contexto de ATM, cuyo modo de servicio es orientado a conexión, no resulta posible lograr un equilibrio entre calidad de servicio y utilización sin decidir acerca de la admisión de nuevas sesiones¹. El problema básico del control de admisión (CA) de tráfico consiste, por tanto, en determinar en tiempo real si, con los recursos de red disponibles, es posible satisfacer los requisitos de calidad de servicio demandados por una nueva conexión sin incumplir los de las conexiones ya en curso.

Tradicionalmente, se ha intentado hallar una respuesta a esta cuestión tratando de establecer una relación directa entre los parámetros estadísticos que describen la dinámica del modelo de tráfico y las variables de prestaciones. Desde el punto de vista del usuario, estas medidas son inmediatamente dos: la probabilidad de pérdida de celdas y la

probabilidad de bloqueo en el establecimiento de una conexión. La literatura técnica al respecto es muy abundante (cf. las monografías [2], [3, y la lista de referencias allí contenidas]). No obstante, este clásico procedimiento paramétrico se enfrenta con varias dificultades:

- a) La complejidad de caracterizar adecuadamente, en términos estocásticos y haciendo uso de pocos parámetros, cada flujo de tráfico;
- b) La dificultad de analizar matemáticamente los nodos de conmutación cuando el proceso de tráfico a su entrada no es simple;
- c) Aun en los casos en que un análisis aproximado sea factible, la acusada sensibilidad de las variables de prestaciones frente a los parámetros del modelo, sensibilidad que se puede atribuir al rigor de los objetivos de calidad pretendidos: probabilidades de pérdida de los datos del orden de 10^{-9} y probabilidades de bloqueo no superiores de 10^{-2} [1];
- d) La pobre caracterización que aportan medidas de calidad de servicio tales como probabilidades de pérdida o bloqueo *promedio*, que como se muestra más adelante conducen a sistemas sobredimensionados.

La síntesis de reglas de admisión de conexiones basadas en la adopción previa de (un conjunto reducido de) modelos de tráfico y el estudio de la capacidad de un multiplexor en tales condiciones ha sido, pues, fuertemente criticada por la falta de adecuación entre los modelos y el tráfico real observado[6].

En parte por estas razones, se viene abriendo paso en los últimos años un enfoque diferente. Consiste en basar las decisiones de admisión de una

¹ Los términos sesión y conexión se usan indistintamente a lo largo de esta comunicación, con el significado genérico de flujos de datos identificados de alguna manera por la red y a los que se asigna una parte de los recursos de ésta.

nueva sesión en unos pocos parámetros *simples* --- por ejemplo, PCR (*Peak Cell Rate*), SCR (*Sustained Cell Rate*)--- declarados por la fuente de tráfico, y en la observación del grado de utilización de los elementos de la red. Tales métodos de admisión basados en medidas e inferencias están siendo activamente investigados [4-7]; entre sus ventajas se encuentran que: a) toleran una caracterización imprecisa del tráfico; b) son, por propia naturaleza, dinámicos, es decir, se adaptan a las variaciones en la intensidad y en las características temporales del tráfico; c) son simples y obtienen ganancias de multiplexación estadística muy significativas.

Cabe colegir que el diseño de métodos dinámicos de control de admisión eficaces depende de un conocimiento cuantitativo adecuado de la relación entre las variables que se miden y las de prestaciones, así como también de la interacción entre las diferentes escalas de tiempo de los fenómenos que se observan (en ATM, la transmisión de celdas y la duración de las sesiones).

Con el propósito de contribuir a identificar los mecanismos que influyen en las prestaciones de los esquemas de admisión basados en medidas, y de estudiar su viabilidad frente a esquemas más clásicos, en esta comunicación comparamos las prestaciones de dos métodos dinámicos de control de admisión de conexiones en una red ATM: uno en el que todos los parámetros de los procesos de tráfico son conocidos *a priori* y otro basado en medidas. La característica más distintiva de estos modelos es que, en la línea que proponen [4] y [5], en ambos se contemplan restricciones tanto en la pérdida de datos como en la probabilidad de bloqueo. Pese a que la mayoría de los trabajos sobre métodos de control de admisión se centran sólo en las primeras, parece claro que ignorar la dinámica de las sesiones conlleva un menor nivel de utilización. Además, la probabilidad de pérdida de celdas se evalúa de dos formas diferentes, el promediado "conservador" (es decir, la restricción se satisface en cualquier estado del sistema), y el promediado "agresivo" (la restricción se cumple sólo en el régimen estacionario). Mostramos, en particular, que el promediado conservador sobredimensiona el recurso. En cuanto al método basado en medidas, la diferencia más importante con respecto a otros modelos (por ejemplo, el de [6]) es la consideración de tráfico multiclase. En todos los casos, nuestro interés no es la estructura de una política óptima de admisión, sino el cálculo de la región admisible con ambos métodos, esto es, el cálculo de la capacidad del sistema.

El resto del trabajo se organiza de esta forma: en la sección 2 se presenta un modelo simple de admisión. En la sección 3 se analizan algunas de sus prestaciones, en concreto la forma y dimensión de la región admisible y la factibilidad de políticas de

admisión. La sección 4 contiene las líneas esenciales del modelo de admisión basado en medidas, que se analiza luego en la sección 5. La sección 6, por último, compara los resultados de los diversos experimentos numéricos y resume las conclusiones principales del estudio.

2. Admisión a un enlace ATM: el modelo básico

En esta sección planteamos el problema de la admisión a un nodo ATM con una única línea de salida de tasa de transmisión c . Consideramos un sistema sin búfer que sirve a S clases de tráfico. Las sesiones de la clase i se reciben según procesos de Poisson independientes de tasa λ_i , $i = 1, \dots, S$, y tienen una duración con distribución exponencial de media μ_i^{-1} . A partir del instante de admisión al sistema, una sesión de la clase i alterna períodos de actividad, en los cuales emite celdas a velocidad constante v_i , con períodos de inactividad. Los períodos activos (respectivamente, inactivos) tienen una duración aleatoria con distribución exponencial de parámetro β_i (resp. α_i). La emisión de celdas se modela como un proceso fluido continuo. Es conveniente señalar que ni la duración exponencial de los intervalos de permanencia en cada estado ni la duración exponencial de las sesiones son hipótesis esenciales del modelo; los resultados se pueden extender sin gran dificultad teórica a modelos con distribuciones de tipo fase, por ejemplo. Todo el exceso de tráfico ofrecido al sistema se pierde y, en principio, no existe ningún control de prioridad sobre el tráfico no cursado: el tráfico de clase i perdido es proporcional al tráfico de clase i ofrecido.

Nuestro modelo es idéntico al de [4], excepto en un aspecto clave: en lugar de plantearnos el control óptimo asignando *a priori* un factor de importancia relativo a cada clase de tráfico, buscamos calcular la región admisible máxima, bajo la condición de no superar una cierta proporción de pérdida de celdas para cada clase y no superar una probabilidad de bloqueo máxima para todas las sesiones.

Así formulado, el sistema queda completamente especificado por el vector

$$(\bar{k}, \bar{n}) = (k_1, \dots, k_S; n_1, \dots, n_S)$$

en donde k_i denota el número de sesiones de tipo i admitidas al sistema y n_i el número de sesiones activas de la clase i . Bajo cualquier política estacionaria de admisión, este vector de estado define un proceso de Markov irreducible y aperiódico en $2S$ dimensiones fácilmente analizable [8], [10]. Note, sin embargo, que aunque se considera un proceso de tráfico con memoria (dos estados) para cada fuente individual, el número de fuentes activas de cada clase se supone irrelevante para las decisiones de admisión. Se está suponiendo,

en realidad, que la parte (n_1, \dots, n_S) del vector de estado alcanza la distribución estacionaria entre dos transiciones consecutivas de la parte (k_1, \dots, k_S) . En [4] puede consultarse una justificación matemática rigurosa de esta hipótesis de separación o independencia entre escalas temporales que, básicamente, se justifica por dos razones: no afecta de manera importante a los resultados y reduce la dimensión del problema original $(2S)$ a la mitad (S) . Por el contrario, la tasa de pérdida de celdas sí depende de la distribución de probabilidad de la parte (n_1, \dots, n_S) del proceso, y se incluye explícitamente en él.

En las condiciones establecidas, el tráfico ofrecido al sistema en el estado $(k_1, \dots, k_S; n_1, \dots, n_S)$ es

$$\lambda(\bar{n}) = \sum_{i=1}^S n_i \nu_i$$

y el tráfico no admitido

$$(\lambda(\bar{n}) - c)^+ \equiv \max\{\lambda(\bar{n}) - c, 0\}$$

La distribución estacionaria de probabilidad del sistema viene dada por el producto de S binomiales

$$\pi(\bar{n} | \bar{k}) = \prod_{i=1}^S \binom{k_i}{n_i} p_i^{n_i} (1 - p_i)^{k_i - n_i}$$

con $p_i = \alpha_i / (\alpha_i + \beta_i)$ la probabilidad de que una fuente de tipo i esté en estado activo mientras permanece en el sistema. La tasa de tráfico perdido de la clase i , por unidad de tiempo, es entonces

$$L_i(\bar{k}) = \sum_{\bar{n}} \pi(\bar{n} | \bar{k}) (\lambda(\bar{n}) - c)^+ \frac{n_i \nu_i}{\lambda(\bar{n})}$$

Dada la independencia de todos los procesos de llegadas, la distribución estacionaria de probabilidad del número de sesiones admitidas se obtiene a partir de la expresión

$$p(\bar{k}) = G^{-1} \prod_{i=1}^S \frac{\left(\frac{\lambda_i}{\mu_i}\right)^{k_i}}{k_i!}$$

siendo G^{-1} una constante de normalización. En estos términos, la probabilidad de bloqueo de una sesión de la clase i es

$$B_i = G^{-1} \sum_{\bar{k} \in R_i} \prod_j \frac{(\lambda_j / \mu_j)^{k_j}}{k_j!}$$

expresión en la que

$$R_i \equiv \{\bar{k} \in R : \bar{k} + \bar{e}_i \in R\}$$

denota el conjunto de estados en los que se rechazan las sesiones de tipo i , y \bar{e}_i es un vector unitario con un uno en la coordenada i -ésima.

Tabla 1: parámetros de los procesos de tráfico.

Clase	p_i	ν_i	μ_i	f_i	ε_i
I	0.1	1.5	1	10^{-9}	10^{-2}
II	0.0375	4	1	10^{-9}	10^{-2}
III	0.01875	8	1	10^{-9}	10^{-2}

El objetivo que se pretende es el cálculo y la caracterización de las regiones acotadas R y Λ tales que:

- (1) $L_i(\bar{k}) \leq f_i k_i p_i \nu_i, \quad i = 1, \dots, S, \quad \forall \bar{k} \in R$
- (2) $B_i \leq \varepsilon_i, \quad i = 1, \dots, S, \quad \forall \bar{\lambda} \in \Lambda$

Tanto f_i como ε_i son números reales positivos; f_i representa un límite a la fracción de tráfico medio ofrecido de la clase i que se pierde; ε_i representa la probabilidad máxima de bloqueo para una conexión de clase i . En todos nuestros experimentos numéricos hemos fijado los valores $\varepsilon_i = 10^{-2}$ y f_i entre 10^{-9} y 10^{-6} .

Note que el primer conjunto de ecuaciones (1) impone un límite superior que afecta sólo a \mathbf{k} y obliga a que la restricción se mantenga en cualquier estado válido del sistema. Esta hipótesis es, según veremos, innecesariamente restrictiva, pues supone siempre una condición de peor caso posible. Note también que el segundo conjunto (2) de ecuaciones define una cota inferior para R (y superior para Λ), que depende no tan sólo de \mathbf{k} sino además de $\lambda_1, \dots, \lambda_S$.

3. Esquemas de control de admisión

El problema que definen las ecuaciones (1)-(2) (que son no lineales) requiere una carga de cómputo proporcional al número de estados, y por tanto, su complejidad aumenta exponencialmente con la dimensión S .

Sin embargo, para capacidades c moderadas y pocas clases de tráfico, es posible resolverlas directamente en un tiempo no muy grande. Para tratar de obtener una comprensión básica del modelo acerca de cuestiones como la linealidad de la región admisible, la existencia de un control de admisión y la ganancia de multiplexación estadística, se han resuelto varios casos de estudio numéricos. Los parámetros elegidos figuran en la Tabla 1.

La capacidad se fija en $c = 50$ y la duración de cualquier sesión se ha normalizado tomándola como la unidad de tiempo $\mu_i = 1$. Observe que para las tres clases de tráfico la tasa media es la misma (0.15). Los apartados que siguen explican los resultados que se han observado.



Fig. 1: región admisible, $f_i = 10^{-9}$

3.1 Linealidad

La figura 1 representa la frontera de la región admisible R (en las coordenadas (k_1, k_2, k_3)) para las tres clases de tráfico dadas. Todos los puntos situados por debajo de la superficie corresponden a configuraciones para las que se satisfacen los requisitos de calidad de servicio.

Excepto en los bordes de dos de los ejes coordenados (cuando el número de sesiones de un tipo determinado está próximo a la saturación), y despreciando las discontinuidades debidas a que las coordenadas deben ser números enteros, la superficie es lineal. El principio de linealidad conlleva importantes consecuencias. Primero, la región admisible queda determinada una vez se conocen sus puntos de intersección con cada uno de los ejes coordenados (máximo de 118 sesiones de clase I, 42 de clase II y 11 de clase III). El cálculo de esta región se reduce, así, a una sucesión de problemas en una sola dimensión. Segundo, hace posible definir un criterio de admisión lineal, asociado al concepto de ancho de banda efectivo o equivalente [9]. Este ancho de banda efectivo es, además, aproximadamente constante para cualquier configuración de tráfico admisible. Cabe señalar, sin embargo, que tal como se han definido las restricciones en la pérdida de celdas, este ancho de banda efectivo es independiente del proceso de llegadas y salidas de las sesiones, y se refiere únicamente a la región R y a las restricciones (1).

3.2 Existencia de un control de admisión

Si se fija la región R máxima en que la fracción de tráfico perdido no supera los umbrales establecidos, existe una región acotada en las coordenadas $(\lambda_1, \dots, \lambda_S)$ que hace que el problema (1)-(2) sea factible. Es inmediato deducir este hecho sin más que observar que B_i es una función continua estrictamente monótona creciente en $\lambda_1, \dots, \lambda_S$.

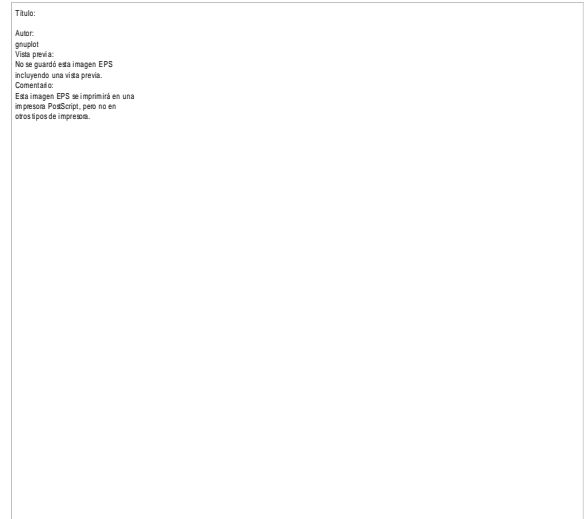


Fig. 2: región (λ_1, λ_2) -factible, probabilidad de bloqueo 10^{-2} . La recta formada por los puntos de menor cota corresponde a $f_1 = f_2 = 10^{-9}$; la otra, para $f_1 = 10^{-9}$, $f_2 = 10^{-6}$, pone de manifiesto la dependencia de la región λ -admisible de ambos tipos de garantías de servicio.

En la figura 2 se ha representado esta región para las clases de tráfico I y II. De nuevo, la característica básica es que la gráfica es lineal.

Conjuntamente, las figuras 1 y 2 implican que para el cumplimiento de ambas clases de garantías de servicio (la de pérdida de celdas y la de bloqueo de las sesiones) no es posible una definición simple de la noción de ancho de banda equivalente: éste depende ahora de las tasas (λ_1, λ_2) .

En contraste con el trabajo descrito en [4], donde en lugar de obtener la región admisible óptima se utiliza una región con estructura más sencilla, el diseño de esquemas de control de admisión simples es, en nuestro caso, más difícil si no se conocen las tasas $\lambda_1, \lambda_2, \dots, \lambda_S$.

3.3 Restricciones en régimen estacionario

Puesto que las llegadas se suponen poissonianas, la distribución de probabilidad del número de sesiones que observa cada sesión aceptada es $p(k)$. Por tanto, parece lógico sustituir las inecuaciones (1) por las S restricciones en donde

Es decir, la proporción entre el tráfico perdido y el ofrecido no debe ser, en régimen estacionario, superior a f_i . Vea que (1) implica (3). Por consiguiente, la región admisible del problema que definen las ecuaciones (2)-(3) es siempre mayor que

$$(3) \quad \sum_{\bar{k} \in R} g_i(\bar{k}) p(\bar{k}) \leq 0, \quad i = 1, \dots, S$$

la del dado por las ecuaciones (1)-(2).

$$g_i(\bar{k}) = L_i(\bar{k}) - f_i k_i v_i p_i.$$

Tabla 2: capacidad del multiplexor con restricciones en régimen estacionario.

Clase	k_{\max}	λ_{\max}	Δk_{\max}
I	129	111.8	9.3%
II	50	37.72	29%
III	15	8.1	36%

Señalemos, no obstante, que (3) depende ahora de \mathbf{k} y $\lambda_1, \dots, \lambda_s$, por lo que se precisa de algún criterio para definir qué solución de (2)-(3) es la óptima. En nuestro caso, hemos optado por tomar la región que maximiza el número de sesiones de una clase de tráfico.

La tabla 2 recoge, para cada tipo de tráfico, el número máximo de sesiones admisibles y la tasa máxima de llegadas al multiplexor, supuesto que sólo se atiende a un tipo de tráfico. La mejora relativa que se aprecia es tanto mayor cuanto menos regular sea el tráfico.

3.4 Prioridades

En escenarios con tráfico heterogéneo, es típico que las restricciones relativas a la pérdida de celdas no se alcancen simultáneamente, sino que sean estrictas sólo para una de las clases. Esto significa que priorizando la aceptación de tráfico de esta clase frente al de otras, es posible incrementar la utilización del nodo e incrementar también el tamaño de la región admisible.

Para verificar realmente este argumento, hemos repetido los cálculos para las clases I y II concediendo mayor prioridad a la clase I, con arreglo a la función de pérdidas

$$(\lambda_1(\bar{n}) - c)^+$$

para el tráfico de tipo I y

$$(\lambda(\bar{n}) - c)^+ - (\lambda_1(\bar{n}) - c)^+$$

para el tráfico de tipo II, con $\lambda_1(n) = n_1 v_1$. Lo que quiere decir que se acepta todo el tráfico posible de una clase (la I) y ninguno de la otra (la II) cuando la tasa global de entrada excede la capacidad. Desde luego, no hay razón alguna para suponer que una estrategia así sea la óptima², pero una vez más, nuestro interés no es tanto obtener un control óptimo como identificar los mecanismos que afectan a las prestaciones. Por otra parte, es bastante habitual limitar la búsqueda de políticas de admisión óptimas a alguna familia sencilla: en la mayoría de los casos, políticas que imitan el número máximo total de conexiones admitidas. En tal caso, los cálculos se simplifican considerablemente y no es demasiado laborioso deducir propiedades estructurales.

² La decisión de qué celdas descartar puede depender no sólo del tipo sino del estado del sistema.



Fig. 3: región (λ_i) -factible, probabilidad de bloqueo 10^{-2} .

Las cualidades más destacables de la figura 3, que representa los resultados obtenidos al ejercer tal control de prioridad sobre las celdas, son, por un lado, el mantenimiento de la linealidad; y por otro, la constatación de que las sesiones de tipo I sólo aprecian la mejora en el servicio cuando su tasa de llegadas es lo bastante intensa.

3.5 Comentarios

Los ejemplos anteriores muestran la existencia de soluciones al problema de control de admisión con dos clases de restricciones, ya sean en su forma conservadora (1)-(2) o en la forma que considera la dinámica de las sesiones (2)-(3). Aunque ya ha sido comentado, es oportuno insistir en que la consideración de ambas medidas de calidad persigue lograr un equilibrio explícito entre utilización y bloqueo sin recurrir para ello a ninguna función de coste elegida previamente.

La linealidad de la región admisible permite descomponer el problema en una serie de subproblemas análogos en una dimensión. Por último, el descarte selectivo de celdas, aunque beneficioso, proporciona sólo un incremento marginal del rendimiento.

4. Un modelo basado en medidas

Pese a la simplicidad con que se han modelado las fuentes en las secciones 2 y 3, parece claro que las prestaciones de los esquemas que allí se analizaron son bastante sensibles al conocimiento preciso de los parámetros p_i, v_i y, en particular, λ_i .

Una manera de evitar tener que ajustar un modelo paramétrico a las fuentes reales de tráfico es recurrir a medidas directas o indirectas de la utilización del recurso para adoptar las decisiones de

admisión. Una forma simple de proceder podría ser la siguiente: dadas las mismas condiciones para los procesos de tráfico y de llegada de las sesiones que en el modelo básico de la sección 2, y empleando la misma notación, una sesión se acepta, supuesto el nodo en el estado $\mathbf{k}=(k_1, \dots, k_S)$, si y sólo si el número de sesiones activas $\mathbf{n}=(n_1, \dots, n_S)$ pertenece a la región $A(\mathbf{k})$, en donde $A(\mathbf{k})$ es un cierto subconjunto que define las combinaciones admisibles en el estado \mathbf{k} . Note que, por considerar un recurso sin búfer, las únicas medidas posibles son las relativas a la carga actual de tráfico, \mathbf{n} .

El proceso estocástico $k(t)$ que representa a las sesiones en curso en el instante t es, por tanto, un proceso de nacimiento y muerte en S dimensiones, con transiciones:

$$\begin{aligned} &\text{de } \mathbf{k} \text{ a } \mathbf{k} + \mathbf{e}_j \text{ con tasa } \lambda_j a(\mathbf{k}) \\ &\text{de } \mathbf{k} \text{ a } \mathbf{k} - \mathbf{e}_j \text{ con tasa } k_j \mu_j \end{aligned}$$

siendo $a(\mathbf{k})$ la probabilidad de aceptar una sesión en el estado \mathbf{k} .

Como antes, el tráfico total perdido por unidad de tiempo viene dado por

$$\left(\sum_j n_j \nu_j - c \right)^+$$

Y la tasa media de tráfico perdido de la clase j en el estado \mathbf{k} por

$$L_j(\bar{\mathbf{k}}) = \sum_{\bar{\mathbf{n}}} \pi(\bar{\mathbf{n}} | \bar{\mathbf{k}}) \left(\sum_i n_i \nu_i - c \right)^+ \frac{n_j \nu_j}{\sum_i n_i \nu_i}$$

La probabilidad de bloqueo, finalmente, viene dada por

$$B = \sum_{\bar{\mathbf{k}}} p(\bar{\mathbf{k}}) (1 - a(\bar{\mathbf{k}}))$$

expresión en donde $p(\mathbf{k})$ denota la distribución estacionaria de probabilidad del proceso $k(t)$.

El problema queda entonces definido por las condiciones

$$(4) \quad \sum_{\bar{\mathbf{k}}} g_j(\bar{\mathbf{k}}) p(\bar{\mathbf{k}}) \leq 0, \quad j = 1, \dots, S$$

$$(5) \quad \sum_{\bar{\mathbf{k}}} p(\bar{\mathbf{k}}) (1 - a(\bar{\mathbf{k}})) < \varepsilon$$

con

$$g_j(\bar{\mathbf{k}}) = L_j(\bar{\mathbf{k}}) - f_j \nu_j k_j p_j$$

Note que todas las clases de sesiones observan ahora la misma restricción a la probabilidad de bloqueo.

5. Prestaciones

Para el examen de las prestaciones del modelo basado en medidas, se supondrán conocidos los parámetros μ_j y las componentes del vector de estado \mathbf{n} y \mathbf{k} ; las tasas máximas deben ser declaradas por las fuentes de tráfico en el momento del establecimiento de la conexión[1], y el vector de estado se obtiene por observación directa del sistema. Son, por lo tanto, desconocidos tanto el tráfico medio (p_j) como la intensidad de llegadas (λ_j).

5.1 Selección de sesiones

Es evidente que el cumplimiento de las restricciones (4)-(5) depende directamente de la elección de las regiones $A(\mathbf{k})$. Puesto que la simplicidad es un criterio básico a la hora de desarrollar esquemas de admisión, un método directo de estimar el tráfico medio de cada clase de sesiones es formar las ratios

$$P_j \equiv \frac{n_j}{k_j}$$

y utilizar el siguiente criterio de admisión: una sesión nueva se acepta si y sólo si

$$G_j(\bar{\mathbf{k}}) \leq 0, \quad j = 1, \dots, S$$

en donde G_j es la tasa de pérdidas de la clase j , en el estado \mathbf{k} , calculada a partir de los $\{p_j\}$. Observe que n_j/k_j se trata como el valor verdadero de p_j , despreciando la incertidumbre presente en todo proceso de estimación, y que además se lleva a cabo una estimación por cada decisión de admisión, lo que significa que no existe memoria en la extracción de los parámetros P_j . En la práctica, es de esperar que los resultados con este método se degraden en mayor medida cuanto mayor sea la varianza del proceso de tráfico (y, por consiguiente, la del estimador), esto es, cuanto menor sea p_j para un mismo valor de ν_j . En la tabla 3 se han recogido los resultados para las tres clases de tráfico analizadas de manera individual. Vea que, como se anticipaba, el valor de λ_{\max} para el cual se verifican ambas garantías de servicio es obviamente menor que el obtenido con el modelo de la sección 2, y que el decremento relativo es mayor para las clases con tráfico más variable.

Una manera de compensar esta pérdida de eficiencia, que se explora en [7], consiste en acotar linealmente las regiones de decisión $A(\mathbf{k})$ con una recta tangente a la función de ancho de banda efectivo de las fuentes de tráfico [9]. Sin embargo, el esquema de [7] presenta dos inconvenientes: depende de la elección empírica de un parámetro de control externo al método, y no proporciona un control inherente a la probabilidad de pérdida.

Tabla 3: prestaciones del modelo simple basado en medidas para cada clase de tráfico.

Clase	λ_{\max}	Eficiencia
I	77	69%
II	22	58.32%
III	4.48	55.3%

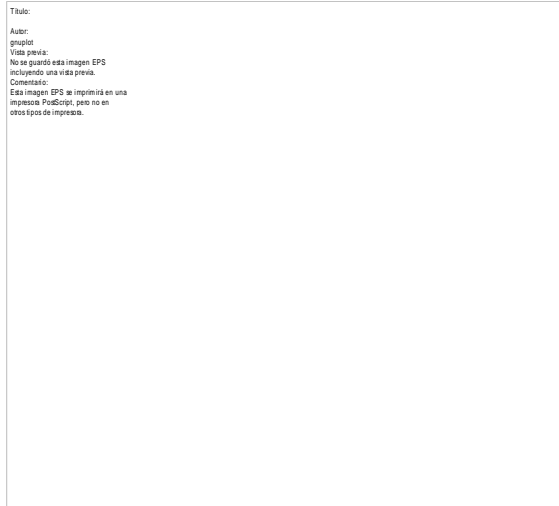


Fig. 4: Región admisible de los modelos con medidas (curva inferior) y completo (curva superior).

5.2 Región admisible

Para comprobar si se mantiene la linealidad aproximada de la región (λ_1, λ_2) -admisibile con el método basado en medidas, hemos resuelto las ecuaciones (4)-(5) simultáneamente para las clases de tráfico I y II, sin optar por ningún control de prioridad en cuanto al descarte de celdas y para una fracción objetivo de pérdidas de $f_i = 10^{-9}$.

Los resultados obtenidos se presentan en la figura 4, en la que se aprecian, por un lado, la menor eficiencia debida a la varianza de las medidas, y por otro lado, un ajuste peor que en el caso anterior a la condición de linealidad. No obstante, los resultados no parecen concluyentes, por lo que esta cuestión se está estudiando con mayor detalle.

En cualquier caso, resulta claro en la gráfica que la admisión basada en medidas consigue una parte significativa de la ganancia de multiplexación estadística del modelo completamente especificado (hasta casi un 70% para el tráfico de clase I) y que es capaz de controlar los dos parámetros de prestaciones en un amplio rango del espacio de coordenadas (λ_1, λ_2) .

5.3 Backoff

Cuando la intensidad de llegadas de las llamadas de una de las clases es elevada, puede no ser suficiente con acotar la probabilidad de bloqueo, sino que quizá pueda interesar limitar la frecuencia (número de sesiones por unidad de tiempo) que son rechazadas. Al mismo tiempo, en el modelo basado en medidas, ocurre que la mayoría de las decisiones de admisión erróneas se producen por la combinación de una alta intensidad de sesiones junto con errores en la estimación de los parámetros p_j .

Un procedimiento simple para contrarrestar este efecto es la introducción de un mecanismo de memoria cada vez que se rechaza una sesión. Así, por ejemplo, se puede modificar el modelo de la sección 4 en la misma línea que [11] de manera que, si se ha rechazado una sesión, la siguiente no sea admitida hasta que una de las actuales haya abandonado el sistema. Esta es una de muchas posibles alternativas similares, como la de esperar un tiempo aleatorio, un tiempo constante o esperar la salida de una sesión del mismo tipo.

Con los parámetros seleccionados, y haciendo uso de la primera de las alternativas mencionadas, se han obtenido los resultados de la tabla 4, que representan una ligera mejora de entre un 3% y un 5%.

6. Conclusiones

En este artículo se ha desarrollado un estudio preliminar de comparación entre dos modelos simples de admisión a un nodo/ enlace ATM con tráfico heterogéneo.

Nuestra motivación ha sido la de identificar empíricamente algunas propiedades básicas, con el objetivo de comprender qué factores afectan a las prestaciones cuando se fijan restricciones de calidad de servicio que actúan en dos escalas de tiempo diferentes, la de la probabilidad de pérdida de celdas y la probabilidad de bloqueo de las sesiones.

Las conclusiones fundamentales de la comunicación son, de una parte, la viabilidad de los métodos de admisión fundamentados en criterios lineales, tanto en el número de sesiones de cada clase como en las tasas de llegadas λ_i ; y, en segundo lugar, la posibilidad de sintetizar métodos de admisión basados en medidas que, explotando de forma óptima las observaciones de la red combinadas con los parámetros de tráfico suministrados por las fuentes de tráfico, dispongan de una región admisible capaz de satisfacer ambos tipos de garantías de servicio y de alcanzar una gran parte de la ganancia de multiplexación estadística que se tiene con el modelo completamente especificado.

Tabla 4: intensidad máxima de llegadas para cada clase de tráfico en el modelo basado en medidas.

Clase	λ_{\max}
I	80
II	22.8
III	4.65

Como la simplicidad de los modelos elegidos se justifica sólo por razones de tratabilidad analítica, parece conveniente considerar una generalización de la situación que aquí se ha presentado, en el sentido de analizar el efecto de fuentes de tráfico con correlación, sistemas con búfer, y métodos de admisión que, explícitamente, compensen la posibilidad de observaciones correlacionadas.

Referencias

- [1] “Recomendación ITU-I.371: Traffic Control and Congestion Control in B-ISDN”. ITU-T, Ginebra (Suiza), marzo 1994.
- [2] “Advances in the Fundamentals of Networking”, *IEEE J. on Selected Areas in Communications*, vol. 13, n° 5-6, 1995.
- [3] J. Roberts, U. Mocchi, J. Virtamo (editores). “*Broadband Network Teletraffic*”. Final Report of Action COST 242. Springer, 1996.
- [4] D. Mitra, M. I. Reiman, J. Wang. “Robust Dynamic Admission Control for Unified Cell and Call QoS in Statistical Multiplexers”. *IEEE J. on Selected Areas in Communications*, vol. 16, n° 5, pp. 692-707, 1998.
- [5] M. I. Reiman, A. Schwartz. “Call Admission: a New Approach”. Bell Labs Tech. Report, 1997.
- [6] R. J. Gibbens, F. P. Kelly, P. B. Key. “A Decision-Theoretic Approach to Call Admission Control in ATM Networks”. *IEEE J. on Selected Areas in Communications*, vol. 13. N° 6, pp. 1101-1114, 1995.
- [7] R. J. Gibbens, F. P. Kelly. “Measurement-based Admission Control”, en *Teletraffic Contributions for the Information Age*, actas del International Teletraffic Congress’ 15 (ITC’15) (V. Ramaswami, P. E. Wirth, editores), pp. 879-888, Elsevier 1997.
- [8] K. W. Ross. *Multiservice Loss Models for Broadband Communications Networks*. Springer, 1995
- [9] A. Elwalid, D. Mitra. “Effective Bandwidth of General Markovian Traffic Sources and

Admission Control of High Speed Networks”. *IEEE/ACM Trans. On Networking*, vol. 1, pp. 329-343, 1993.

- [10] S. M. Ross. *Stochastic Processes 2/e*. Wiley, 1996
- [11] N. G. Bean. “Robust Connection Acceptance Control for ATM Networks with Incomplete Source Information”. *Ann. Operations Research*, 1994.

Capítulo 5B: Sistemas Multimedia y Móviles

Evaluación del Rendimiento de un Protocolo de Acceso Múltiple, basado en un TDMA Dinámico.

M. García, A. Ferrándiz, L. Muñoz, R. Sanz (U. Cantabria)

Propuesta de un Protocolo de Acceso P-TDMA para Redes Inalámbricas de Banda Ancha.

L. Muñoz, J. Choque, M. García, R. Sanz (U. Cantabria)

Técnicas de gestión para la mejora del protocolo TCP en entornos radio.

R. Pérez Sánchez, M. Ruíz García (U. Málaga)

Evaluación del Rendimiento de un Protocolo de Acceso Múltiple basado en un TDMA Dinámico Bajo Tráfico de Naturaleza Autosimilar

Marta García, Alberto Ferrándiz, Luis Muñoz, Roberto Sanz
Area de Ingeniería Telemática Universidad de Cantabria
Avda. de los Castros 39005 Santander (CANTABRIA)
Telf: 942-201549, Fax: 942-201488
E-mail: marta@tlmat.unican.es

Abstract

In recent years, different media access protocols for sharing a high-speed channel in wireless ATM networks have been proposed. The performance evaluation of these protocols is typically obtained using Poisson processes for modelling input data traffic. However, a number of studies have shown that most of the traffic which will be supported by ATM networks have inherent self-similar characteristics. The authors shall simulate the behavior of a dynamic TDMA access protocol under different traffic categories, which require different kinds of services, such as CBR, VBR and ABR, this last one modeled as a self-similar process for interactive and file transfer data applications. Results in terms of throughput, delay and loss packet probability are included and some conclusions have been extracted from them.

1. Introducción

Los sistemas de comunicaciones móviles de primera y segunda generación han experimentado un enorme crecimiento en la última década. Estos sistemas se centran en la transmisión de la voz mientras que el servicio de datos se puede considerar un servicio de valor añadido. Sin embargo, está claro que los futuros sistemas de comunicaciones personales deberán soportar un amplio abanico de aplicaciones multimedia, con tráfico de distintas características y servicios con diferentes requerimientos. Uno de los principales aspectos de diseño, condicionante de la calidad de servicio proporcionada por el sistema, es la elección de un mecanismo que regule eficientemente el uso compartido del medio, lo que se conoce como protocolo de acceso múltiple o Media Access Control (MAC). Así, en los últimos años se han propuestos diferentes protocolos MAC para redes de banda ancha sin hilos con transporte ATM, [1]. Uno de los protocolos más relevantes, debido a sus implicaciones en el proceso de estandarización de este tipo de redes, es el protocolo Multiservice Dynamic Reservation (MDR) TDMA, propuesto por Raychadhuri & Wilson en [2]. El comportamiento de este protocolo fue evaluado asumiendo modelos de tráfico Poissoniano tanto para la voz como para los datos. Sin embargo, existen estudios basados en medidas de tráfico real que demuestran que la naturaleza del tráfico que soportan la mayoría de las redes actuales es de carácter autosimilar [3], [4], [5], es decir, presenta correlaciones sobre un amplio rango de escalas temporales (dependencia a largo plazo)

El objetivo de este trabajo consiste en evaluar el comportamiento del protocolo MDR-TDMA bajo un tráfico de característica autosimilar con objeto de compararlo con el obtenido en [2] en términos de utilización, retardo y probabilidad de pérdida de paquetes.

2. Descripción del Protocolo

Una de las líneas de investigación abiertas de mayor relevancia en el campo de las redes sin hilos de próxima generación se centra en definir la forma en que los terminales comparten el acceso a un canal de comunicaciones común para las transmisiones terminal-estación base. En los sistemas de comunicaciones móviles diseñados principalmente para la transmisión de voz con requerimientos de servicio CBR (Constant Bit Rate), el protocolo MAC que se implementa se basa típicamente en un esquema TDMA estático y se utiliza una modulación digital de banda estrecha en el nivel físico. Sin embargo, cuando el sistema tiene que soportar diferentes tipos de tráfico, este protocolo no parece suficiente para alcanzar los requerimientos de calidad de servicio adecuados. Por tanto, parece necesario modificar la técnica TDMA para mejorar el rendimiento mediante la asignación dinámica de los recursos. Este es el objetivo del protocolo MDR TDMA. Tal y como se muestra en la figura 1, la trama para el canal ascendente, asumiendo una disciplina FDD (Frequency Division Duplex), se divide en pequeñas ranuras temporales para el envío de las peticiones de los terminales y varias ranuras para la transmisión de los datos desde los terminales a la estación base.

Los terminales de voz y datos hacen sus reservas en las ranuras de petición siguiendo una técnica de contención de tipo Aloha ranurado. Una vez que un terminal transmite con éxito la petición de reserva, debe esperar hasta la próxima trama para conocer las ranuras temporales que le han sido asignadas, dado que la estación base asigna los recursos al final de cada trama.

Cada slot temporal de datos transporta un paquete con 48 bytes de información útil y unos pocos bytes de cabeceras de control.

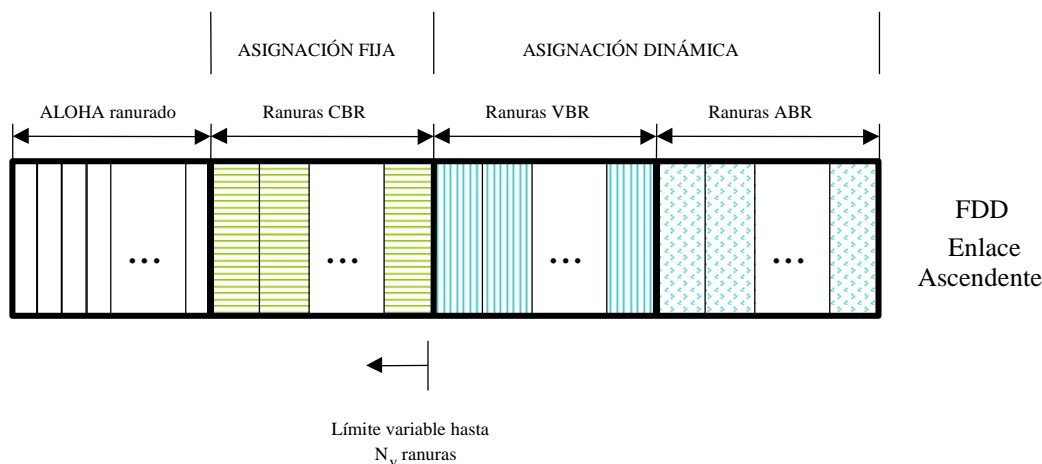


Fig. 1. Estructura de la trama del protocolo MDR

Esta parte de la trama se subdivide en dos ventanas separadas por un límite variable que separa dinámicamente los slots destinados a soportar tráfico con requerimientos CBR de la parte destinada al tráfico VBR (Variable Bit Rate) y ABR (Available Bit Rate). Este límite puede alcanzar un máximo de N_v slots y este parámetro debe ser menor que el número total de slots de datos en la trama. En el caso del tráfico CBR, una vez que una petición ha sido aceptada se garantiza el acceso a un slot fijo en cada trama hasta que finaliza la conexión. Por otro lado, en los slots posteriores al límite dinámico se acomodan los paquetes de los usuarios que han demandado un servicio VBR o ABR de forma que se les asigna dinámicamente uno o más slots de datos tras el último slot asignado para el tráfico CBR. Las largas ráfagas de datos, que no se pueden transmitir en una sola trama, se segmentan y se transmiten en las tramas siguientes. En este protocolo conviven dos filosofías de asignación de recursos radicalmente distintas: mientras que en la ventana anterior al límite dinámico la asignación del canal lógico de la comunicación es fija, es decir, una vez asignado el slot para una determinada conexión este no cambia hasta su finalización, en la ventana posterior a dicho límite las asignaciones de canales lógicos pueden cambiar en cada trama dependiendo de las necesidades de los terminales con tráfico VBR o ABR. Para este último, las políticas de servicio en la cola que se proponen en [2] son: FCFS (First Come, First Served), es decir, los paquetes de datos se sirven por orden de llegada, y TOE (smallest Time-Of-Expiry first served), en la que se da prioridad a los paquetes con requerimientos temporales más restrictivos o menor timeout. En este trabajo se ha aplicado únicamente la disciplina FCFS.

En cuanto al tráfico con requerimientos de servicio VBR, la asignación de ranuras se basa en un algoritmo mediante el cual la capacidad disponible es prorrateada entre las demandas de los usuarios. Las operaciones que se efectúan para realizar este

prorrateo no están definidas en [2]. Por tanto se pueden plantear y evaluar distintas estrategias de asignación de recursos.

Para las peticiones de voz se define un tiempo máximo de establecimiento de la conexión. Una vez transcurrido este tiempo sin haberle sido asignado ningún slot, la llamada es bloqueada. De forma similar, se define un tiempo máximo para los mensajes de datos. Si este timeout finaliza, el mensaje se pierde. Tanto para las peticiones de voz como para las de datos, cuando se produce una colisión, se retransmite en el intervalo de petición de la siguiente trama. Si la petición no colisiona, la estación base envía una confirmación y encola la petición hasta que se disponga de los recursos solicitados o se supere el timeout.

3. Caracterización del Tráfico

La evaluación del rendimiento de un protocolo de acceso múltiple es fuertemente dependiente del modelo de tráfico de fuente asumido. En los últimos años se ha hecho un enorme esfuerzo de investigación para caracterizar el tráfico principalmente en entornos LAN y WAN, [3], [4], [5]. De hecho, muchos han sido los estudios experimentales que, basándose en medidas de tráfico reales, han conducido a modelos más realistas como los modelos de tráfico autosimilar. De acuerdo con estos resultados parece necesario analizar el comportamiento de los protocolos de acceso caracterizando el tráfico con estos nuevos modelos. Existen ya estudios en los que se analiza el efecto de la autosimilitud en el rendimiento de la red y se encuentran diferencias importantes respecto de los resultados obtenidos considerando modelos Markovianos como los procesos de Poisson, [6], [7].

A continuación se detallan los modelos de caracterización de tráfico considerados en este trabajo, tanto para el tráfico de voz como para el tráfico de mensajes de datos.

3.1.1. Modelo de Tráfico para la Voz

Para la caracterización del tráfico de voz se ha considerado el mismo modelo que en [2], es decir un proceso Poissoniano. Un terminal de voz inactivo genera nuevas llamadas con una tasa media de llegadas de λ_v llamadas/sg con una distribución exponencial del tiempo entre llegadas. Cada llamada origina un secuencia continua de bits a una tasa constante con una duración distribuida exponencialmente de media T_v . Un terminal con una llamada pendiente se considera ocupado hasta que finaliza la transmisión o se supera el tiempo máximo de acceso y el terminal vuelve a estar inactivo.

3.1.2 Modelo de Tráfico para los Mensajes de Datos

En este trabajo se han considerado dos modelos diferentes para la caracterización del tráfico de mensajes de datos, que se asume solo con requerimientos de servicio ABR. En primer lugar, se ha aplicado el modelo tradicional de Poisson, al igual que con la voz, que supone una distribución exponencial para el tiempo entre llegadas y la longitud de los mensajes, tal y como proponen Raychaudhuri and Wilson en [2]. Sin embargo, existen numerosos artículos que han mostrado la naturaleza autosimilar del tráfico real de diferentes redes de paquetes, debida principalmente a las características de los sistemas de ficheros, al dinamismo de los protocolos de transporte de la red y al comportamiento del usuario. Estos estudios demuestran que la superposición de un gran número de fuentes ON/OFF independientes e idénticamente distribuidas con tiempos de ON y OFF que sigan una distribución *heavy-tailed* da lugar a un proceso autosimilar. Por tanto, el segundo modelo de caracterización del tráfico de datos considera que los periodos de actividad y silencio de cada terminal son dos variables aleatorias que siguen una distribución *heavy-tailed* como es la distribución de Pareto en lugar de una exponencial.

La función de distribución acumulada de Pareto viene dada por, [8]:

$$F_X(x) = 1 - \left(\frac{k}{x}\right)^a \quad k > 0, a > 0; x \geq k \quad (1)$$

Donde a es un parámetro que determina la forma de la cola de la distribución y k es el parámetro de localización que representa el valor más pequeño que puede tomar la variable aleatoria.

Cuando $1 < a < 2$, la distribución presenta varianza finita y media finita dada por:

$$\mu = \frac{ak}{a-1} \quad (2)$$

El grosor de la cola de la distribución de Pareto viene dado por el valor de a : cuanto más grande es a más estrecha es la cola de la distribución. Una

variable aleatoria distribuida según Pareto toma un valor más grande con mayor probabilidad que una variable aleatoria distribuida exponencialmente. Cuando el número de fuentes de Pareto es grande, su superposición da lugar a un proceso autosimilar con parámetro de Hurst dado por:

$$H = \frac{3-a}{2} \quad (3)$$

El parámetro de Hurst mide el grado de autosimilitud del proceso. Cuando $0,5 < H < 1$, el proceso es dependiente a largo plazo. Los procesos con un parámetro de Hurst cerca de 1 son altamente a ráfagas mientras que aquéllos con valor cercano a 0,5 presentan menos características de ráfagas. Por tanto, este parámetro es indicativo de la *burstiness* del tráfico de datos.

4. Modelo de Simulación

El escenario de simulación considerado para evaluar el rendimiento del protocolo MDR-TDMA, bajo las características de tráfico de fuente descritas en el apartado anterior, es similar al utilizado en [2]. Los parámetros de simulación se detallan en la tabla 1.

El canal ascendente es compartido por terminales que generan tráfico de voz a una velocidad constante de 32 Kbits/sg que demandan servicio CBR y terminales de paquetes de datos que transfieren ficheros de longitud media 5,12 Kbits con requerimientos de servicio ABR. Este tráfico de datos se subdivide a su vez en dos grupos, un grupo de mensajes críticos en tiempo con un time-out o tiempo de expiración más restrictivo y mensajes no críticos en tiempo con mayor time-out. El porcentaje de paquetes críticos sobre el total del tráfico de datos viene dado $\alpha=0.25$. La longitud de la trama es de 12 msec y el 5% de ésta se destina al intervalo de ranuras de petición. De acuerdo con los resultados obtenidos en [9] por Raychaudhuri *et al*, el número máximo de ranuras de datos para el tráfico CBR viene dado por la relación entre el tráfico de voz ofrecido respecto del total de tráfico ofrecido con objeto de alcanzar un buen balance entre la probabilidad de bloqueo y el retardo para los paquetes de datos. En este trabajo, la carga total ofrecida se divide al 50% entre la carga de voz y la carga de datos. Por tanto, el valor máximo de la ventana CBR N_v se ha fijado a la mitad del número total de ranuras de datos en una trama.

En cuanto al retardo de retransmisión al producirse una colisión en las ranuras de contienda, en [2] se asume que las peticiones de voz se retransmiten en la siguiente trama mientras que las peticiones de datos esperan un tiempo aleatorio de media 10 msg aunque no se especifica qué distribución sigue esta variable. En nuestro caso se ha supuesto que en ambos casos, voz y datos, se retransmite en la siguiente trama. Sin embargo, esta diferencia no afecta mucho a los resultados ya que el tiempo de trama es de 12 msg.

Tabla 1: Parámetros de simulación para el sistema MDR-TDMA

Capacidad del enlace, R	1,92 Mbits/sec
Duración de la trama, T	12 msec
Ranuras de petición por trama, N	20
Tamaño de la ranura voz/datos, bits	424 (384 payload)
Ranuras de datos por trama	52
Tamaño máximo de la ventana CBR, N_v	26
Tasa media de llegada de llamadas, λ_v	0.0005 calls/sec
Duración media de una llamada	180 sec
Velocidad binaria de la voz	32 Kbits/sec
Tiempo máximo de acceso para la voz	5 sec
Tasa media de llegada de mensajes, λ_d	0.0375 msg/sec
Longitud media del mensaje	5.12 Kbits
Time-out mensajes no críticos	1sec
Time-out mensajes críticos	50 msec
% mensajes críticos sobre el total, α	25%

Se han realizado dos simulaciones diferentes: una caracterizando el tráfico correspondiente a los mensajes de datos con un modelo Poissoniano, tal y como se aplica en [2], y la otra asumiendo que los periodos de ON/OFF de las fuentes que generan dichos mensajes siguen distribuciones *heavy-tailed* de Pareto, tomando el parámetro a igual a 1,2.

Para simular el sistema se ha utilizado el software BONEs (Block Oriented Network Simulator) DESIGNER 3.6 sobre estaciones de trabajo, modelos HP 712 and Sun Sparcstation 10, [10]. Cada iteración simula tres millones de tramas.

5. Resultados

En esta sección se presentan los resultados obtenidos de las simulaciones realizadas. En cada gráfica se incluyen las curvas tanto para el modelado del tráfico de datos con Poisson como con Pareto.

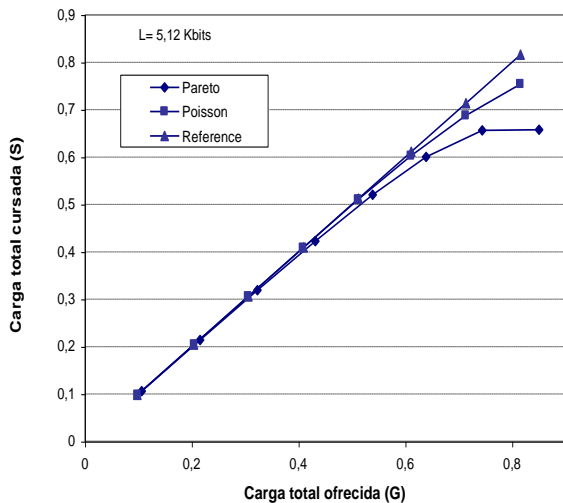


Fig.2 : Utilización del canal (S) vs. carga total ofrecida (G) (L=5,12 Kbits)

El retardo de los mensajes de datos que se representa en la figura 5 se define como el tiempo que transcurre desde que el terminal genera el mensaje hasta que consigue reservar ranuras para su transmisión.

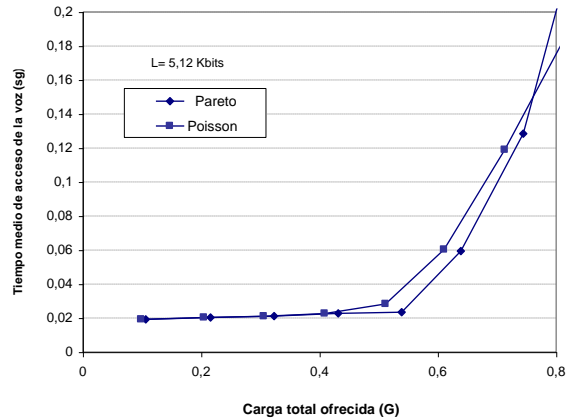


Fig. 3: Retardo medio de acceso de la voz vs. carga total ofrecida (L=5,12 Kbits)

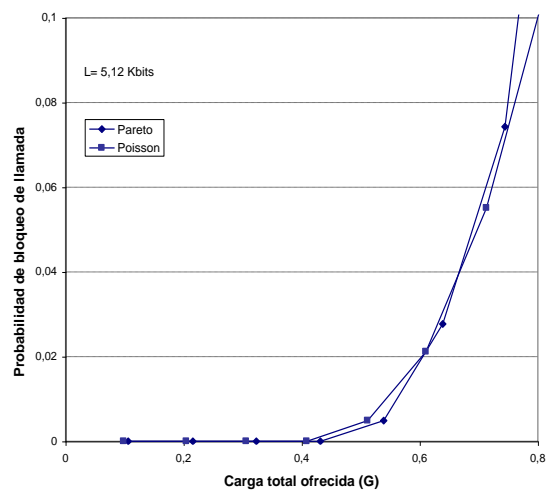


Fig. 4: Probabilidad de bloqueo de las llamadas de voz vs. carga total ofrecida (L=5,12 Kbits)

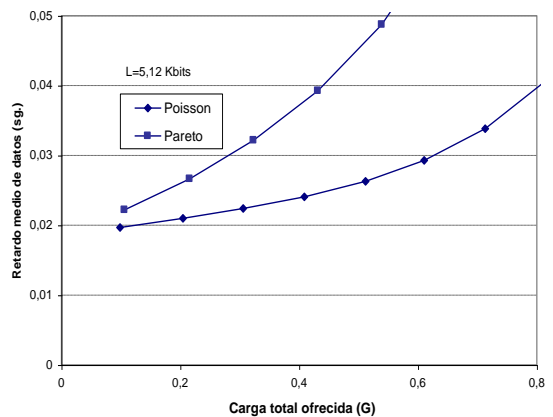


Fig. 5: Retardo medio de los mensajes de datos vs. carga total ofrecida (L=5,12 Kbits)

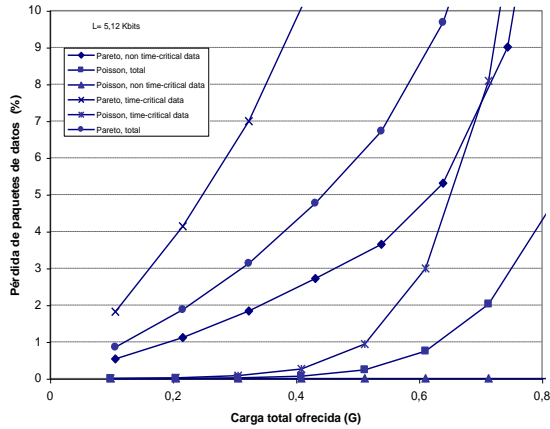


Fig. 6: Porcentaje de pérdida de mensajes de datos vs. carga ofrecida (L=5,12 Kbits).

El comportamiento del protocolo bajo tráfico de datos modelado con Pareto es aceptable en términos de probabilidad de bloqueo, pérdida de paquetes y retardo para cargas ofrecidas por debajo de 0,4, mientras que, en el caso Poissoniano se alcanzan las mismas prestaciones para cargas de 0,6.

En las fig. 8-11 se presentan los resultados obtenidos para una longitud media del mensaje de datos de 300 ATM-cells (L=115,2 Kbits). Esta longitud implica una nueva tasa media de llegadas de mensajes de $\lambda_d = 16,6 \cdot 10^{-4}$, con objeto de mantener el mismo número de terminales para cada valor de carga ofrecida que en el caso anterior.

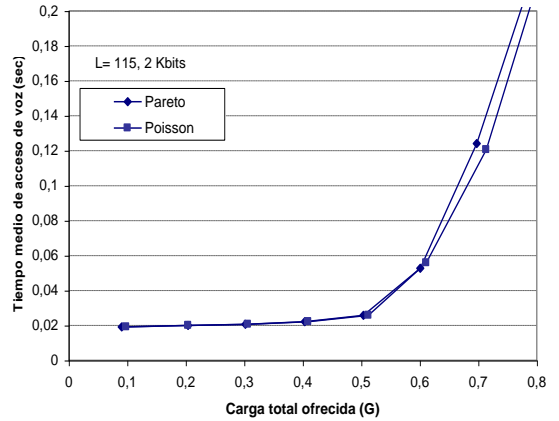


Fig. 8: Tiempo medio de acceso para las llamadas de voz vs. carga total ofrecida (L=115,2 Kbits)

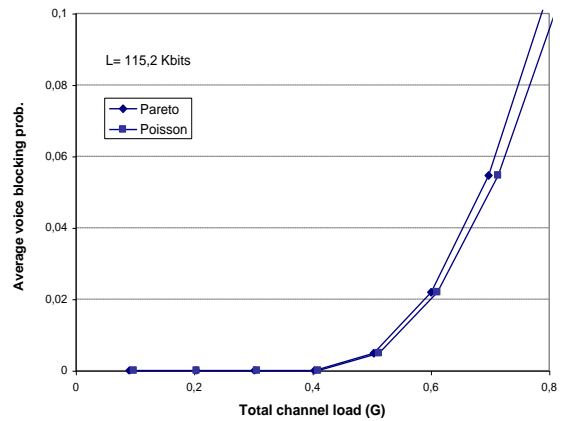


Fig. 9: Probabilidad de bloqueo de la voz vs. carga total ofrecida (L=115,2 Kbits)

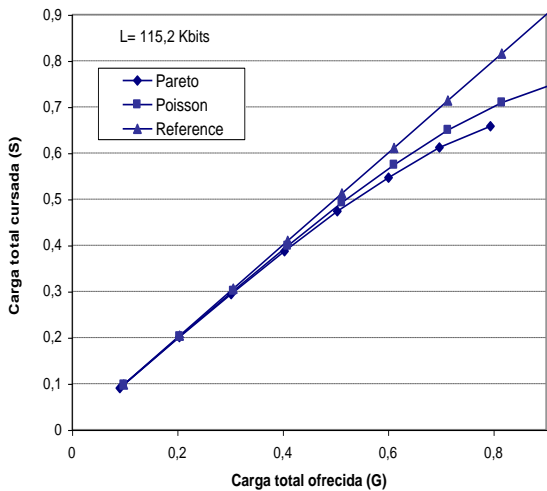


Fig. 7: Utilización del canal (S) vs. carga total ofrecida (G) (L=115,2 Kbits)

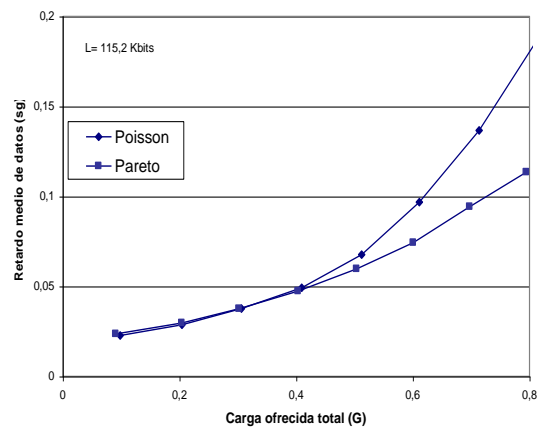


Fig. 10: Retardo de los mensajes de datos vs. carga total ofrecida (L=115,2 Kbits)

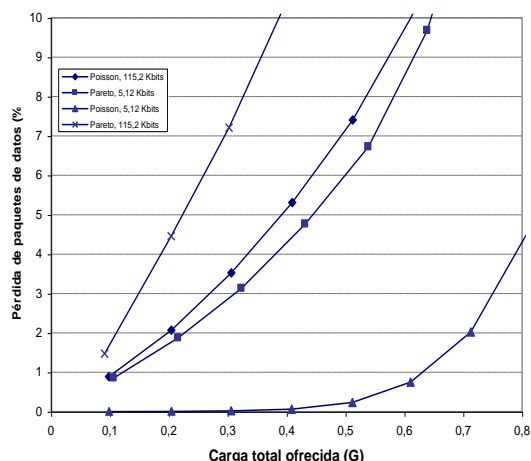


Fig. 11: Porcentaje total de pérdida de mensajes de datos vs. carga total ofrecida ($L=5,12$ Kbits, $L=115,2$ Kbits)

6. Conclusiones

A la vista de los resultados mostrados en la sección anterior, se puede concluir que el comportamiento del protocolo MDR-TDMA, analizado ya en [2] suponiendo tráfico de datos Poissoniano, se degrada considerablemente si se supone un modelo auto-similar para la caracterización de dicho tráfico con los parámetros considerados en las simulaciones realizadas. Por tanto, a lo hora de dimensionar el sistema, considerando por ejemplo, aplicaciones de datos tipo FTP o WWW, asumir modelos tradicionales no proporciona una medida realista de los recursos necesarios para asegurar una calidad de servicio adecuada.

Agradecimientos

Este trabajo ha sido financiado por la Comisión Interministerial de Ciencia y Tecnología, CICYT, proyecto TIC96-1074-C02-01.

Referencias

- [1] J.Sánchez, R. Martínez, M.W. Marcellin, "A Survey of MAC Protocols Proposed for Wireless ATM", IEEE Networks, Nov-Dec 1997, pp. 52-62
- [2] D. Raychaudhuri, N.D. Wilson, "ATM-Based Transport Architecture for Multiservices Wireless Personal Communications Networks", IEEE Journal on Selected Areas in Communications, vol. 12, No. 8, October 1994, pp. 1401-14.
- [3] W.E Leland, M. S. Taqqu, W. Willinger, D.V. Wilson, "On the Self-Similar Nature of Ethernet Traffic (Extended Version)", IEEE/ACM Transactions on Networking, vol 2, No. 1, February 1994, pp. 1-15
- [4] V. Paxson, S.Floyd, "Wide Area Traffic: The Failure of Poisson Modeling", IEEE/ACM Transactions on Networking, vol. 3, No. 3, June 1995, pp. 226-43.

- [5] R.G. Addie, M.Zukerman, T.D. Neame, "Broadband Traffic Modeling: Simple Solutions to Hard Problems", IEEE Communications Magazine, August 1998, pp. 88-95.
- [6] J. Aracil, L. Muñoz "Performance of Aloha Channels under Self-similar Input", Electronics Letters, vol. 33, No. 8, April 1997, pp. 659-60.
- [7] I. Rombogiannakis, M. Paterakis, "Voice/Data Integrated Wireless Channel Access in Third Generation Digital Cellular Networks: The Performance of Bursty Data Generated by Interactive Applications", International Journal of Wireless Information Networks, vol. 5, No. 1, 1998, pp. 1-12.
- [8] N.L. Johnson, S. Kotz, N. Balakrishnan, *Continuous Univariate Distributions*, Vol I, Wiley Interscience, Second Edition, ch. 19, pp. 233-249.
- [9] N.D. Wilson, R. Ganesh, K. Joseph, D. Raychaudhuri, "CDMA vs. dynamic TDMA for access control in an integrated voice/data PCN", in Proceedings of 1st International Conference Universal Personal Communications, Sep 1992. 12, pp. 267-272.
- [10] S.J. Schaffer, W.W. LaRue, "BONeS DESIGNER: "A Graphical Environment for Discrete-Event Modeling and Simulation", in Proceedings of the Second International Workshop on Modeling, Analysis and Simulation of Computer and Telecommunication System, 1994, pp. 371-374.

Propuesta de un Protocolo de Acceso P-DTDMA para Redes Inalámbricas de Banda Ancha

Luis Muñoz, Johnny Choque, Marta García, Roberto Sanz
Departamento de Ingeniería de Comunicaciones
ETSII y Telecomunicaciones - Universidad de CANTABRIA
Avda. Los Castros s/n, 39005 SANTANDER
Tel: 942-201497, Fax: 942-201488
E-mail: luis@tlmat.unican.es

Abstract

An integrated multimedia wireless network must be designed for transporting voice, video and data traffic, fulfilling the service quality requirements demanded by these types of traffic. The MAC protocols based on resource allocation on demand are the most suitable to fulfil the requirements of the integrated multimedia wireless network. In this paper, we propose a novel multiple access scheme denominated Polling-Dynamic Time Division Multiple Access, P-DTDMA which combines the best characteristics of the DTDMA proposed by D. Raychaudhuri and the modified token ring proposed by A. Acampora. The P-DTDMA is evaluated under CBR and ABR traffic. For the ABR services, the input traffic has been modelled as a self-similar process assuming Pareto distributions for ON/OFF period lengths.

1. Introducción

Una red inalámbrica con capacidad para soportar servicios multimedia debe estar diseñada para transportar tráfico de voz, vídeo y datos, cumpliendo los requisitos de calidad de servicio exigidos por dichos tipos de tráfico. Siendo ATM la tecnología que brinda mejores prestaciones para el transporte simultáneo de estos tipos de información, una de las posibilidades a tener en cuenta es que las redes inalámbricas para servicios de banda ancha adopten la tecnología ATM. De esta forma se combina la ventaja de la movilidad proporcionada por las redes inalámbricas con la multiplexación estadística y la calidad de servicio garantizada por las redes ATM de cable.

En este contexto, el protocolo MAC propuesto debe compensar las limitaciones físicas impuestas por el entorno radio, el cual es muy hostil debido principalmente a sus características de propagación multicamino y desvanecimiento selectivo en frecuencia. Estas limitaciones se hacen más patentes dado que el protocolo MAC propuesto debe proporcionar al tráfico multimedia un medio de comunicación con una probabilidad de error de bit o de pérdida de paquetes muy baja, puesto que la dicha tecnología está basada en el transporte de células en un ambiente con muy baja tasa de error. Para lograr esto existen muchas propuestas a nivel físico, entre ellas el uso de antenas adaptativas.

Los protocolos MAC basados en la asignación de recursos bajo demanda son los más apropiados para cumplir con los requisitos de tiempo real de determinados servicios. Esto es debido a que dichos esquemas asignan los recursos según la demanda, evitando así el desaprovechamiento de aquéllos cuando se utilizan métodos de asignación fija. Además, a diferencia de los esquemas de acceso por contienda, los protocolos de asignación de capacidad bajo demanda reducen el retardo de transmisión y la

ocupación del medio proporcionando un ancho de banda libre de contención durante los períodos de actividad.

Por otra parte, para evaluar la eficiencia de los protocolos MAC propuestos, se analiza mediante simulación el comportamiento que presentan ante tráfico multimedia. El modelado de este tipo de tráfico es un tema en constante investigación y en muchos casos se recurre a distribuciones estadísticas tradicionales, aunque éstas no se ajusten adecuadamente al tráfico real. Una de estas distribuciones es la de Poisson, utilizada adecuadamente para el estudio de tráfico telefónico, aunque se ha demostrado en recientes publicaciones que no se ajusta adecuadamente al modelo del tráfico de vídeo y datos.

2. Objetivos

Este trabajo tiene como objetivo principal presentar y evaluar una nueva propuesta de protocolo MAC para redes inalámbricas dedicadas a la transmisión de datos multimedia. El protocolo propuesto toma las principales ventajas de los protocolos presentados por D. Raychaudhuri [1] y A. Acampora [2] con el objeto de obtener un mejor rendimiento. Puesto que sólo existen curvas de prestaciones para el protocolo presentado por Raychaudhuri, los resultados de las simulaciones del protocolo propuesto serán comparados con éstos [3]. Se analizará el comportamiento del protocolo aplicando modelos de tráfico publicados en recientes estudios.

3. Descripción del Protocolo MAC

El protocolo presentado por Raychaudhuri denominado Multiservices Dynamic Reservation (MDR) TDMA, cuyo formato de trama se muestra en la Fig.1, proporciona acceso múltiple mediante el uso del protocolo ALOHA ranurado y comunicación bidireccional simultánea mediante FDD. En el enlace ascendente se aplica asignación

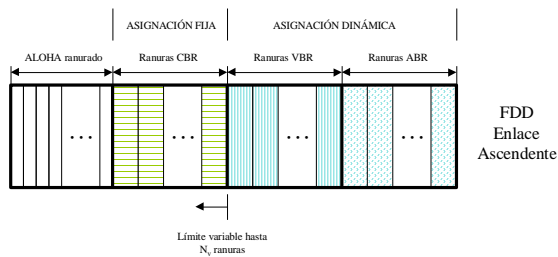


Figura 1. Formato de trama MDR-TDMA propuesta por D. Raychaudhuri

bajo demanda, distribuyendo los recursos entre todas las solicitudes en forma dinámica en función de dicha demanda y del tipo de tráfico a ser cursado (voz, vídeo o datos), brindando además la calidad de servicio requerida por estas fuentes de tráfico. Durante el enlace descendente la estación base distribuye el tráfico de forma programada a cada terminal destino.

El protocolo presentado por Acampora, Fig. 2, opera mediante la técnica TDD, utilizando por lo tanto una única portadora para los enlaces ascendente y descendente. Sin embargo, este esquema no tiene en cuenta la naturaleza del tráfico multimedia, sino que trabaja con el tráfico de datos, generado por consultas a bases de datos remotas. Las solicitudes para asignación del recurso de transmisión durante el enlace ascendente son resueltas mediante el método de sondeo. Para ello, el terminal móvil debe ser admitido previamente en el "árbol de conexión virtual" [4]. Este último es un método que asegura al terminal móvil correspondiente la calidad de servicio adecuada dentro de un grupo de estaciones base por las cuales puede desplazarse libremente. Con objeto de aumentar la eficiencia es posible transmitir más de una celda ATM por ranura de tiempo. Adicionalmente, con el fin de compensar los perjuicios que produce el entorno radio en la transmisión de la información, se emplean antenas adaptativas, las cuales se adaptan mediante la inserción de miniranuras en la trama.

El protocolo MAC propuesto en este trabajo está diseñado para operar en un ambiente pico/micro celular, sobre un único canal de alta velocidad usando un esquema TDMA/TDD. De esta forma, se realiza la transmisión de información mediante

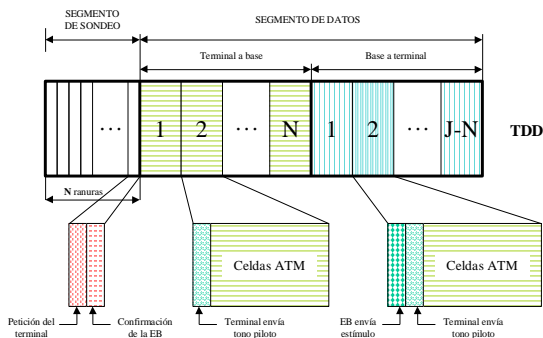


Figura 2. Formato de trama propuesta por A. Acampora

conmutación de paquetes entre cada terminal móvil y su estación base. Esta propuesta permitirá extender los servicios de banda ancha de las redes por cable a los usuarios que cuentan con terminales inalámbricos, ya sean portátiles (estáticos) o móviles (peatones o vehículos), proporcionando el ancho de banda demandado. Observar que esto no es más que la extensión de la tecnología ATM al entorno inalámbrico, actualmente denominado Wireless ATM (WATM).

Un diagrama de la asignación temporal de recursos del protocolo MAC propuesto aparece en la Fig. 3. La trama está constituida por dos segmentos: uno de sondeo y otro de datos, el cual a su vez se divide en dos partes: uno para la comunicación en sentido móvil a base (enlace ascendente), y otro para la comunicación en sentido base a móvil (enlace descendente). El segmento de sondeo contiene N ranuras, una por cada terminal remoto que haya sido admitido por la estación base. Asimismo, tanto el enlace ascendente como el descendente cuentan igualmente con N ranuras cada uno.

Cada ranura del segmento de sondeo se subdivide en dos partes. La primera parte, en sentido terminal a base, es utilizada por los terminales móviles para solicitar recursos a la estación base cuando tienen información que enviar; la segunda parte, en sentido base a terminal, es empleada por aquella para responder a los terminales móviles sobre los recursos que les han sido asignados. Puesto que esta porción de la trama no está protegida por la antena adaptativa se sugiere utilizar técnicas de espectro ensanchado como DS-SS (Direct Sequence Spread Spectrum) para proteger la información de interferencias y de la degradación por propagación multicamino.

En esta primera parte, el terminal móvil envía su petición utilizando su propio código de espectro ensanchado ocupando completamente el ancho de banda del canal. Notar que, durante el segmento de datos, la información es enviada a una velocidad binaria que consume todo el espectro asignado; por lo tanto la velocidad de chip durante el segmento de sondeo es igual a la tasa de bits durante el segmento de datos, con lo que un código de 200 a 300 chips sería adecuado para distinguir los terminales móviles, evitar la interferencia y disminuir el efecto del multicamino.

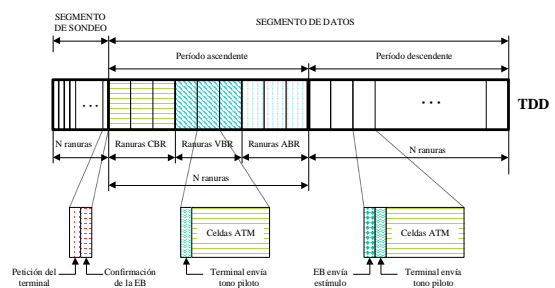


Figura 3. Formato de trama MAC propuesta en este trabajo

En la segunda parte, una vez que la estación base conoce todas las peticiones de los terminales móviles, determina los recursos que debe asignar en función del tipo de tráfico que el terminal móvil desea enviar y le informa a cada uno acerca de la posición de las ranuras asignadas, usando para ello la misma codificación de espectro ensanchado.

En cuanto al segmento de datos, al principio de cada ranura del período ascendente, el terminal móvil envía un breve tono piloto para permitir a la estación base calcular rápidamente los pesos adecuados de los elementos del array de la antena, posibilitando, de esta forma, la correcta recepción por parte de la estación base de las celdas ATM enviadas a continuación por dicho terminal móvil.

De manera similar, al principio de cada ranura del período descendente la estación base envía un estímulo al terminal móvil al cual desea enviar información, consistente de una secuencia de espectro ensanchado que lleva el código de identificación del terminal móvil. Este estímulo se denomina intervalo de sondeo. El terminal móvil responde a dicho estímulo enviando un breve tono piloto de radiofrecuencia para que la estación base calcule los pesos correctos del array de la antena, y a continuación envíe la información al terminal móvil.

Por tanto, con este esquema, cada conjunto de celdas ATM transmitido en cualquier sentido, va siempre precedido de un tono en sentido móvil a base, el cual es usado para adquirir rápidamente los pesos correspondientes de la antena adaptativa en la estación base. Esto permite que se pueda transportar más de una celda ATM por cada ranura.

Así pues, el protocolo propuesto utiliza la técnica de asignación bajo demanda con sondeo centralizado, en donde la estación base es la responsable de gestionar y controlar el acceso de los terminales a los recursos, tal y como propone Acampora.

Continuando con el segmento de datos de la trama, indicar que cada ranura de los enlaces ascendente y descendente contiene un número entero de celdas ATM. Dicho número será un parámetro importante en las simulaciones del protocolo. En general, el protocolo ha sido diseñado para cursar tráfico de tipo CBR (Constant Bit Rate), VBR (Variable Bit Rate) y ABR (Available Bit Rate), tal y como propone Raychaudhuri. En primer lugar se realiza la asignación de recursos a las peticiones para tráfico CBR mediante una asignación fija de una ranura sobre múltiples tramas, emulando de esta manera una conexión de voz por conmutación de circuitos. El número de conexiones CBR puede ser variable hasta llegar a un número máximo (ventana de asignación CBR) el cual es otro de los parámetros del protocolo. Las ranuras restantes son asignadas dinámicamente a las peticiones con requerimientos de servicio VBR o ABR. En el caso de VBR la asignación es realizada en base a un algoritmo de

multiplexación estadística en el cual los recursos disponibles son distribuidos en función de la cantidad de información que cada terminal VBR activo desee transmitir. Es decir, a mayor información a transmitir mayor número de ranuras se asignarán, asegurando que cada terminal VBR activo como mínimo disponga de una ranura para transmitir. Un terminal VBR se considera activo cuando, tras haber realizado una conexión con la estación base, tiene información que transmitir. Se considera inactivo cuando, a pesar de no tener información que transmitir, mantiene su conexión con la estación base.

Después de la asignación de recursos a los terminales con demanda de servicio del tipo CBR y VBR, las ranuras restantes, si hay alguna libre, son asignadas a los terminales móviles que solicitan cursar tráfico ABR. Los paquetes de datos ABR de gran longitud (ráfagas) que no pueden ser transmitidos en una sola trama pueden ser segmentados utilizando entonces múltiples tramas. Existen dos políticas de servicio en la cola para atender al tráfico ABR. La primera se denomina FCFS (First Come First Served), en el cual los paquetes son atendidos según su orden de llegada. Cuando a un paquete de un terminal le llega el turno de ser atendido se verifica que su *tiempo de expiración* no ha sido sobrepasado, en cuyo caso el paquete es cursado. La segunda política es denominada TOE (Time Of Expiry), según la cual los paquetes son almacenados en la cola según una cierta política de prioridades. Se establecen dos clases de paquetes: críticos, con un tiempo de expiración más pequeño, y no críticos con un tiempo de expiración comparativamente más grande. Se asigna una mayor prioridad a los paquetes críticos sobre los paquetes no críticos, por lo que los primeros son encolados delante de los no críticos. Cuando a un paquete le llega el turno de ser atendido, ya sea crítico o no crítico, se verifica que su tiempo de expiración no ha sido sobrepasado. Si es así el paquete es cursado, y en caso contrario el paquete es descartado.

4. Simulación

Para la evaluación del protocolo propuesto se utilizó un entorno gráfico de simulación de propósito general denominado BONEs (Block Oriented Network Simulator) DESIGNER [5]. Este es un software integrado que permite el modelado y la simulación de sistemas de transferencia de datos controlado por eventos.

4.1. Parámetros del Sistema

En principio, el diseño del protocolo se basa en asegurar la calidad de servicio para el tráfico CBR, ya que éste es muy sensible al *jitter*, esto es, a la variación en los valores de los retardos de cada paquete de información transmitido en la red para un terminal concreto. Por lo tanto, todo el tráfico generado durante la duración de una conexión CBR debe ser cursado, en el mejor caso, en un número

exacto de tramas. Esto evita tener que utilizar un mayor número de tramas para cursar dicho tráfico, lo que supondría una degradación, debida al jitter, en los servicios isócronos como es el caso de la voz.. El hecho de que se deba cumplir este requisito crea una dependencia entre los siguientes parámetros de la simulación: capacidad del enlace, número de bits por trama, velocidad binaria de la voz y número de bits útiles por ranura de datos. Por otra parte, con el objeto de tener como referencia los resultados del protocolo propuesto por Raychaudhuri, es necesario mantener los valores de los parámetros comunes entre ambos protocolos. Por lo tanto, otro punto de partida para el diseño es tomar la misma duración de la trama. Esto nos obliga a elevar la capacidad del enlace y, también, por consiguiente elegir una mayor velocidad binaria de la voz.

La Tabla I muestra los valores de los parámetros de la simulación del protocolo propuesto. La población de usuarios que se encuentran admitidos dentro de una estación base se asume que es de dos tipos: CBR y ABR. La carga total ofrecida se divide al 50% entre la carga de voz y la carga de datos. En base a estos valores de carga se obtiene el número de terminales de voz y datos que generan tráfico.

4.2. Modelo del Tráfico de Voz

Para modelar el tráfico CBR se han utilizado los modelos de tráfico telefónico, considerándose los eventos de inicio y finalización de las llamadas telefónicas como una variable aleatoria con distribución de Poisson. Dado que es necesario, desde el punto de vista del simulador, determinar la duración de cada llamada y el intervalo temporal entre la aparición de las llamadas, utilizamos la distribución exponencial negativa para obtener el valor de estas dos variables aleatorias. Por lo tanto, los usuarios CBR generan tráfico de voz a una tasa de bits constante de 64 Kbps, con una tasa de llegada λ de $5.0E-4$ llamadas/segundo (33.3 minutos entre llamadas), y una duración media de 180 segundos. El tiempo máximo de espera de una solicitud CBR es de 5 segundos; transcurrido este tiempo, si la llamada no ha sido atendida se considera llamada bloqueada. La ventana de conexiones CBR soporta un máximo del 50% del total de los recursos del sistema.

Tabla I. Parámetros de la simulación

Parámetro	Valor(es)
Num. Polling Slots	60
Num. Celulas/DataSlot	2
Capacidad del enlace	11.875 Mbits/seg
Duración de la trama	0.012 seg
Tasa de llegada de llamadas	$5.0E-4$ llamadas/seg
Duración media de una llamada	180.0 seg
Tiempo máximo de acceso para la voz	5.0 seg
Velocidad binaria de la voz	64 Kbits/seg
Número máximo de ranuras de voz	30
Tiempo medio OFF	4.31 seg
Tiempo medio ON	$4.31E-4$ seg
Longitud media del Paquete	5.120 Kbits
Factor de utilización ABR	$1.0E-4$
Timeout crítico	0.05 seg
Timeout no crítico	1.0 seg
α	0.25

4.3. Modelo del Tráfico de Datos

Recientes estudios sobre el modelado de tráfico de datos en redes de área local han desvelado nuevas consideraciones sobre sus características [6]. Estas investigaciones revelan que el tráfico de datos presenta dependencia a largo plazo y autosimilitud estadística, es decir, el tráfico exhibe comportamiento de ráfagas a través de un amplio rango de escalas de tiempo que van desde milisegundos hasta minutos u horas. Por lo tanto, para generar tráfico ABR se ha utilizado el modelo de fuentes ON-OFF con períodos de actividad (ON) e inactividad (OFF) de varianza infinita, haciendo necesario utilizar distribuciones del tipo *heavy-tailed* como por ejemplo la distribución de Pareto. Los períodos de actividad son muy breves comparados con los de inactividad. La relación entre ambos períodos se mide mediante el parámetro *factor de utilización ABR*. El valor de cada período es generado según una distribución de Pareto. Durante los períodos de actividad se generan paquetes con una longitud media determinada por el parámetro *longitud media del paquete*. Los datos que generan los terminales se dividen en dos grupos: datos de tiempo real o críticos, los cuales tienen un reducido tiempo de expiración determinado por el valor del parámetro *timeout crítico*, y datos que no son de tiempo real o datos no críticos, los cuales tienen un tiempo de expiración comparativamente grande determinado por el valor del parámetro *timeout no crítico*. El porcentaje de paquetes críticos respecto al total de paquetes (críticos más no críticos) es determinado por el parámetro α .

5. Resultados

En la Fig. 4 se observan las curvas que muestran la eficiencia del protocolo al cursar el tráfico ofrecido, tanto para el protocolo de Raychaudhuri como para el protocolo P-DTMA propuesto. Para este último se muestran los resultados obtenidos aplicando las dos políticas de servicio de cola, FCFS y TOE, para el tráfico de datos.

Comparando con la curva de referencia (en donde todo el tráfico ofrecido es cursado), se observa que hasta un tráfico ofrecido del 50% se cursa prácticamente el total de dicho tráfico. Para valores mayores, las curvas del protocolo propuesto tienden a un valor máximo del 60% frente a un 65% que se alcanza con el protocolo de Raychaudhuri. Esta diferencia se debe a la mayor probabilidad de bloqueo que soportan las conexiones de voz en el protocolo propuesto, lo que produce mayores pérdidas. Además, se observa que no existe una gran diferencia entre las curvas de las diversas políticas de servicio de la cola, apreciándose únicamente una leve mejora cuando el sistema trabaja con un servicio de colas TOE. La influencia de dicha política de servicio se podrá constatar en las curvas de calidad de servicio del tráfico de datos que viene dadas por el retardo de transmisión de los datos y la probabilidad de pérdida de paquetes.

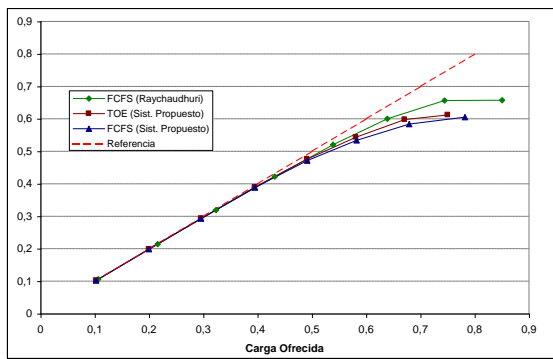


Figura 4. Tráfico ofrecido vs. tráfico cursado (voz y datos)

La Fig. 5 muestra las curvas de probabilidad de bloqueo para llamadas de voz (tráfico CBR). Con el protocolo propuesto se observa una subida abrupta en la región de 40%-50% frente a la región de 50%-60% del protocolo de Raychaudhuri. Esto es debido al que el número de terminales que generan tráfico de voz en el esquema PDTDMA es algo más del doble que en el caso de Raychaudhuri. Por lo tanto, para cargas altas, los recursos disponibles por el sistema para conexiones CBR se saturan rápidamente lo que produce un mayor número de conexiones bloqueadas. A la vez, también se produce un mayor tiempo de espera para acceder a un recurso hasta que se libere una conexión CBR. Este comportamiento se observa en la Fig. 6.

La Fig. 7 muestra las curvas de porcentaje de pérdida de paquetes de datos en función de la carga ofrecida. Se observa que para un valor de operación nominal del 50% las pérdidas de paquetes no críticos suponen alrededor del 2% para ambos protocolos

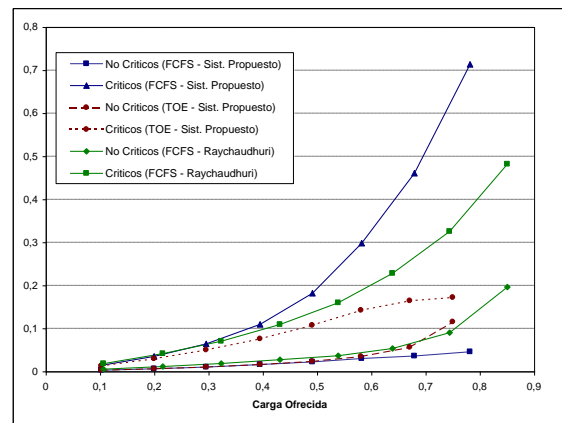


Figura 7. Carga ofrecida vs. porc. de pérdida de paq. de datos

mientras que para los paquetes críticos son de 20% para el protocolo propuesto y 15% para el protocolo de Raychaudhuri. El cálculo de los tiempos de ON y OFF se obtiene en función de la capacidad del enlace, y al tener el protocolo propuesto una velocidad binaria seis veces mayor que el protocolo de Raychaudhuri, se generan mayor número de peticiones de datos a pesar de contar con el mismo número de terminales de datos en ambos protocolos.

La política de servicio de la cola FCFS atiende las peticiones en el orden en que éstas llegan sin importar su nivel de prioridad, dependiente del timeout del paquete. Por lo tanto, al existir un mayor número de peticiones de datos los recursos del sistema se saturan rápidamente, provocando un aumento en el tiempo de espera de los paquetes encolados. Este hecho puede que no signifique un gran problema para los paquetes no críticos, pero sí que lo es para los paquetes críticos, de ahí la gran diferencia que muestran las curvas del protocolo propuesto frente al de Raychaudhuri para valores altos de carga ofrecida.

Por otro lado, se observa que las curvas muestran una notable mejora al utilizar TOE como política de colas para atender peticiones de datos. Esta mejora supera incluso las curvas del protocolo de Raychaudhuri con servicio de colas FCFS, lo que permitiría transportar tráfico de datos multimedia, que son muy sensibles al retardo, de forma eficiente y con una alta calidad de servicio.

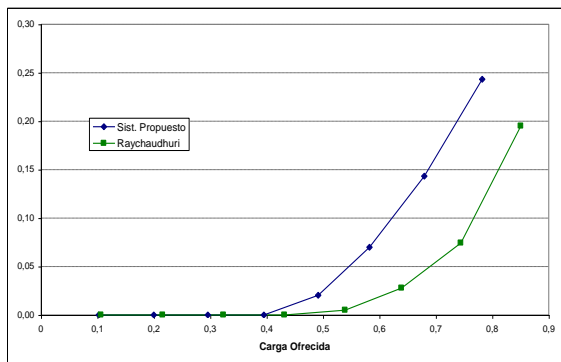


Figura 5. Carga ofrecida vs. probabilidad de bloqueo de llamada

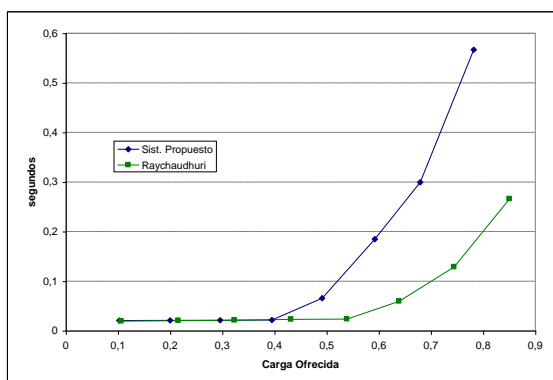


Figura 6. Tiempo promedio de acceso de voz vs. carga ofrecida

El mayor tiempo de espera que soportan las peticiones de datos ante una elevada carga ofrecida se ve plasmado en las curvas de la Fig. 8, en donde se muestra el promedio total de retardo de transmisión de todos los paquetes (críticos y no críticos). Para una carga ofrecida del 50%, los retardos superan los 50 milisegundos que es el valor límite para transmitir un paquete crítico, por lo tanto para cargas mayores del 50% el aumento del retardo de transmisión es debido principalmente al retardo producido por los paquetes no críticos. Este hecho se hace mucho más notorio aún cuando se utiliza una política de colas TOE, ya que ésta obliga a los paquetes no críticos a colocarse al final de la cola y

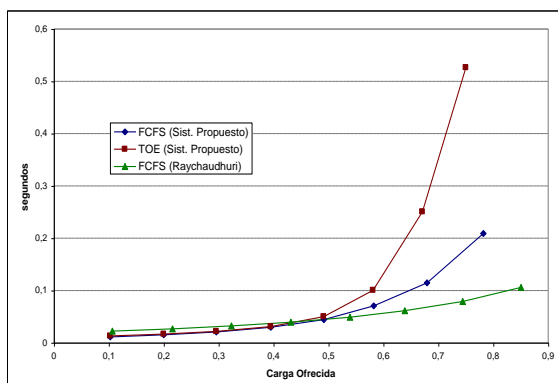


Figura 8. Carga ofrecida vs. ret. medio de transmisión de datos

por consiguiente son atendidos siempre cuando no exista ningún paquete crítico en la cola, lo que produce mayores retardos.

Las curvas de retardo de transmisión de datos merecen un comentario adicional cuando la carga ofrecida es menor del 50%, razón por la cual en la Fig. 9 se muestra ampliada la zona de la gráfica para este rango. Se observa que el protocolo propuesto presenta menores retardos de transmisión de datos para cargas más bajas al 50%, sin importar de forma excesiva la política de colas que se utilice.

6. Conclusiones

En el presente artículo se ha presentado una propuesta de protocolo MAC denominado *Polling-Dynamic Time Division Multiple Access (P-DTDMA)* para redes inalámbricas dedicadas a la transmisión de datos multimedia de banda ancha. En dicha propuesta se emplea la técnica de asignación bajo demanda con sondeo centralizado, basada en la diferenciación del tráfico ofrecido teniendo en cuenta el tipo de servicio demandado por el usuario. Asimismo, en el diseño del protocolo se han tomado en consideración las características hostiles del medio radio, compensándolas mediante el uso de antenas adaptativas.

Por otra parte, la evaluación de las prestaciones del protocolo se ha realizado mediante modelos de tráfico más relacionados con el tráfico real como son los modelos de fuentes ON-OFF con distribuciones de Pareto en lugar de la tradicional distribución de

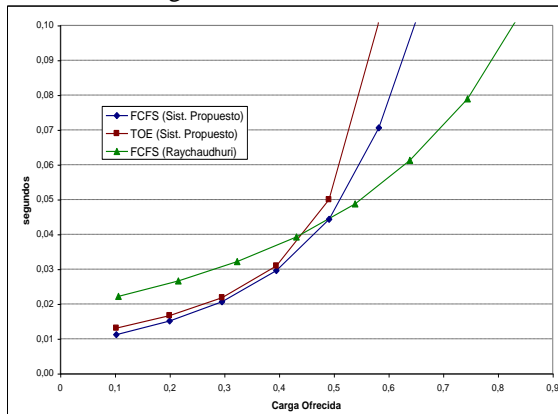


Figura 9. Vista ampliada de las curvas de retardo de datos

Poisson, la cual no se ajusta a la realidad respecto a la generación del tráfico de datos.

Agradecimientos

Parte de este trabajo ha sido financiado por la Comisión Interministerial de Ciencia y Tecnología, CICYT, proyecto TIC96-1074-C02-01.

Referencias

- [1] D. Raychaudhuri and N. D. Wilson, "ATM-Based Transport Architecture for Multiservice Wireless Personal Communication Network", IEEE JSAC, vol. 12, no. 8, Oct. 1994, pp. 1401-14.
- [2] Anthony S. Acampora, Srikanth V. Krishnamurthy, "A New Adaptive MAC Layer Protocol for Wireless ATM Networks in Harsh Fading and Interference Environments", IEEE 6th International Conference on Universal Personal Communications Record, 12-16 October 1997, San Diego, CA USA.
- [3] M. García, L. Muñoz, R. Sanz, "Performance of a Dynamic TDMA Access Protocol under Self-Similar Traffic Input", ICT'99, 15-18 June 1999, Cheju, Korea.
- [4] A. S. Acampora, M. Naghshineh, An Architecture and Methodology for Mobile-Executed Handoff in Cellular ATM Networks, IEEE JSAC, vol. 12, no. 8, october 1994
- [5] S. J. Schaffer, W.W. LaRue, "BONeS DESIGNER: A Graphical Environment for Discrete-Event Modeling and Simulation", MASCOTS'94, Proceeding of the Second International Workshop on Modeling, Analysis, and Simulation of Computer and Telecommunication System 1994
- [6] W. Leland, "On the Self-Similar Nature of Ethernet Traffic (Extended Version)", IEEE/ACM Trans. Networking, Feb. 1994, pp. 1-15.

Técnicas de gestión para la mejora del protocolo TCP en entornos radio

Ricardo Pérez Sánchez, Margarita Ruíz García
Departamento de Tecnología Electrónica Universidad de Málaga
E.T.S.I de Telecomunicación Campus de Teatinos s/n, 29071 (Málaga)
Telf: 952-132843, Fax: 952-131447
E-mail: ricar@dte.uma.es, marga@dte.uma.es

Abstract

In recent years, there is a growing interest in wireless local area networks as a consequence of the large-scale utilization of personal computers and mobile communications. These computers use applications and protocols, originally developed for wired desktop hosts. The performance disparity between wireline and wireless networks is very large, aspects like throughput rates and error control must be considered to achieve good results. Unlike wired networks, packets transmitted on wireless channels are often subject to burst error which cause packet losses. In this paper we study the effect of burst packet errors and error recovery mechanisms employed in wireless MAC protocols on the performance of transport protocols such as TCP. This leads to new scheduling method that considers the channel characteristics, CSDP (channel state dependent packet scheduling).

1. Introducción

El crecimiento incesante en la demanda de servicios, velocidades de acceso mayores y la movilidad de los usuarios, hace que se haya acrecentado en los últimos años el interés por las redes de área local inalámbricas, WLANs (*Wireless Local Area Networks*). Hasta el momento, las redes de área local clásicas, LANs (*Local Area Networks*), limitan a los usuarios a permanecer físicamente en los edificios que albergan las infraestructuras, para la comunicación y el acceso exterior. Sin embargo con las WLANs el escenario que se plantea es diferente, y no es otro que el de servir como medio de acceso a una red principal (*backbone*), utilizando para ello una estación base, a través de la cual los usuarios móviles podrán enviar y recibir información.

Por tanto, dos son los beneficios que nos proporciona esta configuración. El primero es la movilidad, sin estar limitados por ésta, en el acceso a la información, y el segundo es la reducción de costes que supone en algunos casos, la instalación de una red *wireless* que ofrece la conectividad y conveniencia de una clásica LAN, sin la necesidad del gasto en la sustitución o remodelación de las antiguas infraestructuras de cable.

En la sección 2 se comentarán los aspectos más relevantes de las WLANs y la justificación del uso, en la estación base, de un sistema de gestión CSDP. En 3 se propone un modelo para el canal radio, y en 4 mostraremos la reacción que TCP presenta con este tipo de canales. En 5 veremos los problemas planteados en una estación base con una sola cola (FIFO), y en la sección 6 describiremos CSDP, estudiando su comportamiento y comparando con el anterior. En la sección 7 se presentarán resultados con diferentes escenarios y en 8 las conclusiones y líneas futuras.

2. Redes Wireless

2.1. Características principales

Las WLANs proporcionan un servicio de envío de paquetes sin calidad de servicio, es lo que denomina tráfico *best-effort*. Sin embargo, como veremos más adelante, difiere en muchos aspectos de la concepción clásica que se tiene de éste en las redes de cable convencionales. Otra característica es que los ordenadores que se usan en las WLANs, utilizan aplicaciones basadas en protocolos que fueron diseñados originalmente para las redes de cable. Por tanto no contemplan las características propias de un medio de transmisión tan hostil como el aire. Mientras que en el cable los errores que se producen son de naturaleza aleatoria, en un canal *wireless* tenemos ráfagas de errores (*burst errors*). Esto es debido a que en la transmisión vía radio aparecen fenómenos como la propagación multirayecto, los desvanecimientos (*fading*) y la variabilidad temporal del canal. Acrecentándose los efectos, si añadimos la movilidad del usuario y los trasposos de una estación base a otra (*handoff*).

El rango de frecuencias en las que operan la generación actual de WLANs, depende del país. En Estados Unidos el responsable para esta regulación es el *Federal Communications Commission (FCC)* que designó en 1985, dentro del radio espectro de la ISM (*Industrial, Scientific, and Medical Band*), las siguientes bandas de uso libre, 902-928 MHz, 2400-2483.5 MHz, y 5725-5850 MHz. En Europa, el organismo equivalente al FCC, *Conference European des Administrations des Postes et des Telegraphes (CEPT)* reservó 150 MHz de espectro libre en 5.1 GHz para este propósito [1]. Como consecuencia de esta división aparecen, el IEEE 802.11 que es el estándar para WLANs en la banda ISM y desarrollado por el *Institute of Electrical and Electronics Engineers (IEEE)* y el subgrupo creado por el ETSI (*Europe Telecommunications Standard*),

HIPERLAN (*High Performance Radio LAN*) para la banda de 5 GHz en Europa [2]. Para el primero las velocidades de transmisión de datos oscilan entre los 1 y 2 Mbps (Mega bits/segundo), mientras que para el estándar europeo llegan hasta los 23.5 Mbps coincidiendo ambos en un rango de aplicación de entre 50 y 100 metros.

Bajo estas condiciones, las prestaciones de un protocolo de transmisión como es TCP (*Transmission Control Protocol*) se degradan considerablemente.

La mayor parte de las soluciones propuestas en la literatura [3], [4], [5], [6] para mejorar la baja eficiencia del TCP en entornos radio, operan en el nivel de transporte. En este artículo se propone un mecanismo para mejorar las prestaciones en WLANs, pero en el nivel de enlace. La posibilidad de llevar esto a cabo radica en el uso por parte del TCP de un reloj con una granularidad de 500 ms (en la mayoría de las implementaciones), que hace que la latencia en la detección de las pérdidas sea grande. Esto permitiría introducir en el nivel de enlace un mecanismo basado en retransmisiones que nos permitiera recuperar errores sin interferir en el esquema propio del TCP.

2.2. Justificación del uso de CSDP

Normalmente, múltiples sesiones de TCP comparten un mismo enlace radio para acceder a los usuarios móviles. Todo este tráfico es gestionado por una estación base, que es el nexo de unión entre la *backbone* y el entorno móvil. En este escenario, y utilizando en la estación base una sola cola de transmisión, con una política de servicio FIFO, el ancho de banda del enlace *wireless* se degrada, al tener que retransmitir sucesivas veces los paquetes de la cabeza de la cola, HOL (*head of line*), que sufren los errores de ráfaga propios del canal (*burst errors*). Esto provoca un bloqueo en la transmisión de paquetes a otros receptores, donde sus canales respectivos pudieran estar en buenas condiciones. La base de esto radica en que los enlaces de radio son estadísticamente independientes, y durante el intervalo en el que uno se encuentra en mal estado, el resto podrían transmitir sin problemas. Como vemos es indispensable aprovechar el conocimiento del canal *wireless*, para obtener una mayor eficiencia en la transmisión y mejorar la calidad de servicio de las aplicaciones que se apoyan en él.

La propuesta de un gestor de paquetes basado en las condiciones del canal, mejoraría los problemas derivados de compartir un enlace radio. Con el CSDP (*channel state dependent packet*) tendríamos para cada usuario móvil asociado a la estación base un cola independiente. Dentro de cada cola la disciplina de servicio sería FIFO, y las colas se servirían según el estado de su canal radio asociado. Con esta configuración se aumentará la velocidad de transmisión (*throughput*), y se mejorará la eficiencia en el reparto del enlace.

El método de gestión basado en CSDP se implementa en la estación base, esto tiene la ventaja de que aplicaciones basadas tanto en TCP como en UDP (*User Datagram Protocol*) puedan beneficiarse de sus efectos, pensemos en RPC (*Remote Procedure Call*) o NFS (*Network File System*).

3. Canales Wireless LAN

3.1. Características

La pérdida de paquetes en un medio cableado es bastante rara y tiene una naturaleza aleatoria. En cambio, para un medio *wireless* los errores se producen por ráfagas. Como los usuarios se mueven, la potencia de la señal recibida varía, además hay efectos como el desvanecimiento y la interferencia de otros usuarios que degradan las prestaciones del canal. Estos problemas se acrecientan al usar redes WLANs, ya que una de las posibles capas físicas que pueden tener, está basada en el salto de frecuencias FH (*frequency hopping*), por tanto si nos situamos cerca de un objeto que transmita a la misma frecuencia podemos tener problemas, aunque en el siguiente salto no nos afecte, hay que tener en cuenta que trabajamos con velocidades de transmisión muy elevadas y la escasez de técnicas sofisticadas como el control adaptativo de potencia, métodos de corrección de errores o *interleaving*, hacen al sistema vulnerable. No obstante en el estándar europeo (HIPERLAN), se usan algunas de estos sistemas para obtener un buen comportamiento a velocidades de 23.5 Mbps.

Otro aspecto importante es la limitación en la cobertura (50-100 m). Medidas realizadas en entornos WLAN [7] demuestran que la tasa de paquetes erróneos depende de la distancia entre el transmisor y el receptor. De manera que el canal varía para cada usuario, dependiendo de su localización respecto a la estación base.

3.2. Modelado del canal

Los estudios en este tema [8] [9] establecen que para modelar el patrón de error que presentan estos tipos de canales hay que usar diagramas de estados de Markov. Cada estado está asociado con un valor de tasa de error de bit BER (*bit error rate*). La permanencia media en los estados de alto BER, es mayor que el tiempo de transmisión de un paquete, lo que causará pérdidas de retransmisión. Por ello como vemos en la Fig. 1 modelamos el canal con 2 estados, uno bueno (B) y otro malo (M).

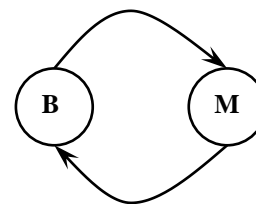


Fig. 1: Modelo de error del canal

El estado bueno indica que la recepción del paquete es correcta y el malo lo contrario (hay bits erróneos). Usaremos p^E para indicar la probabilidad de pérdida que hay en los estados, $E \in \{B, M\}$. Así se pueden perder paquetes estando en un estado bueno si p^B es alta y completar con éxito una transmisión en un estado malo si p^M es bajo. Normalmente $p^M \gg p^B$. El tiempo en el que se permanece en cada estado (bueno y malo) está distribuido exponencialmente, pero con valores medios diferentes.

4. Comportamiento de TCP con las pérdidas en ráfaga

La herramienta de simulación usada para efectuar este análisis ha sido PTOLEMY [10], [11], [12]. La aplicación usada ha sido una transferencia de ficheros FTP (*file transfer protocol*), en dos escenarios diferentes Fig. 2.a y Fig. 2.b, con los parámetros de la Tabla. 1. De esta forma podremos detectar los problemas que presenta TCP cuando trabaja con canales de radio. Los resultados se presentan en la Fig. 3 y la Tabla. 2.

Tabla. 1: Parámetros de simulación para los 2 escenarios

Transmisor (Tx)	Velocidad	5 Mb/s
	Tráfico	FTP
	Fichero	800 Kbytes
	T. Simulación	20 sg
TCP	Ventana Tx	4 Kbytes
	MSS ¹	1024 bytes
	RTT ²	20 ms
	Granularidad	500 ms
Canal Wireless	p^B	0.0
	p^M	1.0
	Media período B	5 sg
	Media período M	0.1 sg
Receptor (Rx)	Ventana Rx	4Kbytes

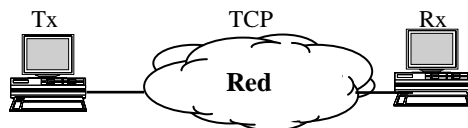
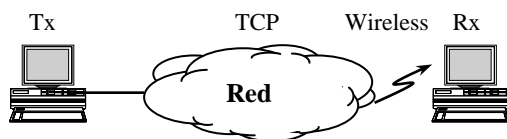


Fig. 2.a: Configuración TCP clásica



¹ MSS (Maximum Segment Size), es el tamaño máximo del segmento TCP utilizado.

² RTT (Round Trip Time), tiempo que transcurre desde que se envía un paquete hasta que se recibe su confirmación.

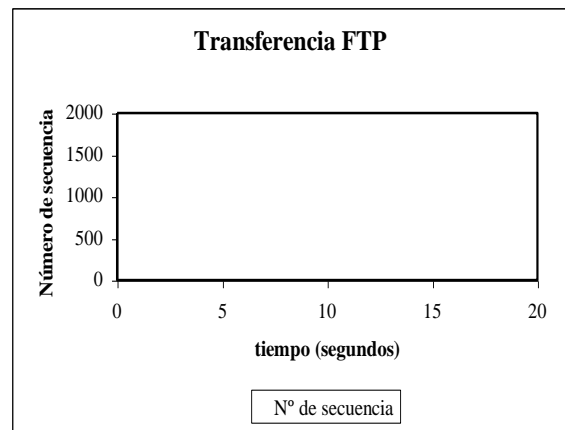


Fig. 2.b: Configuración TCP-Wireless

Fig. 3: Traza de FTP para TCP-Wireless

Tabla. 2: Resultados de las simulaciones para TCP

	TCP Clásico	TCP-Wireless
Throughput	1.38 Mbps	1.15 Mbps
Retardo	4.67 sg	10.29 sg

4.1. Latencia en la detección

Como vemos en Fig. 3 el canal *wireless* está en un estado malo 3 veces, $t = 0$ sg, $t = 4.23$ sg y $t = 11.75$ sg. En el primer *burst* sólo se pierde un paquete, y como el sistema está arrancado, la fase de *slow start* [13] se activa rápidamente volviendo a retransmitir de inmediato. Para los otros dos el retardo es mucho mayor, debido a la latencia en la detección de los paquetes perdidos, repercutiendo negativamente en la tasa de llegada (*throughput*) y en retardo de la aplicación FTP. En estos últimos casos se pierden los 4 paquetes de la ventana de transmisión.

4.2. Baja eficiencia de los protocolos TCP

Algunos de los protocolos que TCP usa para realizar el control de flujo y retransmitir adecuadamente los paquetes perdidos, pierden parte de su efectividad cuando trabajan en un medio radio. Por ejemplo *Fast Retransmit* que fue un añadido a TCP para reducir la latencia en la detección de pérdidas, entra en funcionamiento si se reciben 3 ACKs duplicados. Esto indica que un paquete se ha perdido y debe ser retransmitido sin esperar a que se cumpla su *time-out*. El problema es que al trabajar con un ancho de banda de 1 Mbps, que es pequeño en comparación a los 10 Mbps de las redes Ethernet, las ventanas de transmisión no pueden ser grandes, a veces menores que 3 paquetes. Otro inconveniente es que los ACKs pueden perderse en el canal ascendente (móvil-estación base), impidiendo por tanto la activación del mecanismo.

Por otra parte, si una retransmisión tiene lugar cuando el canal está en un estado malo, el *time-out* es doblado según el algoritmo de Karn [13], y el canal no es utilizado.

5. Estación base con una sola cola

Como hemos visto en la sección anterior, es necesario introducir algún mecanismo que mejore la eficiencia de TCP y evite la pérdida de datos por ráfagas. Cualquier modificación en el nivel de transporte implica que no se puede reutilizar los beneficios para otros protocolos no orientados a conexión como UDP, por eso lo ideal es hacerlo en el nivel de enlace. Además, los *time-outs* usados en los protocolos de enlace, son mucho más pequeños que en los mecanismos del nivel de transporte.

5.1. Recuperación de errores en el nivel de enlace

Debido a la alta tasa de paquetes perdidos en el medio radio, la capa MAC (*medium access control*) de los sistemas WLANs proporciona un mecanismo de confirmación para cada paquete. El estándar para esta capa es del IEEE 802.11. Una de sus recomendaciones [14] es el uso de CSMA/CA + ACK prioritario (*Carrier Sense Multiple Access with Collision Avoidance*). El CSMA/CA trata de reducir la probabilidad de colisión entre múltiples estaciones que acceden a un mismo medio. El ACK prioritario intenta recuperar los paquetes perdidos mediante retransmisiones. Para detectar la pérdida de un paquete, un ACK es retornado por la estación destino inmediatamente después de recibirlo. Los ACKs tienen prioridad para acceder al medio sobre otras estaciones que esperaban a que estuviera libre para transmitir. Si un ACK no es recibido tras la transmisión de un paquete, en la fuente se cumple el *time-out* y se retransmite tras un tiempo aleatorio. Así sucesivamente hasta que se producen R_{max} (número máximo de retransmisiones permitidas) retransmisiones, tras lo cual el paquete se elimina.

5.2. Estación base con una FIFO

Esta configuración consiste en canalizar todos los tráficos entrantes a la estación base, BS (*base station*), a través de una sola cola, que se sirve utilizando una disciplina FIFO. Suponemos que el espacio en ella es elevado, para evitar que se produzca un *overflow*, porque en este estudio no interesan los problemas de congestión en los nodos intermedios, sino el efecto que los canales radio tienen sobre el sistema.

Cuando el medio está libre de errores, el funcionamiento es correcto, pero cuando trabajamos sobre un medio *wireless* con errores de ráfaga, comienzan los problemas. En la sección 7 se presentarán resultados con diferentes escenarios. La topología de red usada consiste en una estación base, con la que se comunican un número de fuentes que generan tráfico a través de canales de cable clásicos. En el otro extremo de la estación base tenemos los receptores con enlaces radio, a los que va dirigida la información. La velocidad de las fuentes es de varios Mbps, mientras que en la estación base sólo disponemos de 1 Mbps. A medida que el número de

fuentes aumente los problemas en la BS irán aumentando.

Algunos de esos problemas son perfectamente predecibles.

5.3. Problemas en la estación base con una FIFO

5.3.1. Bloqueo

Este bloqueo se conoce como HOL (*Head of Line*). Se debe a que la comunicación entre la estación base y los receptores se realiza a través de canales radio estadísticamente independientes. Si el receptor al que va destinado el paquete HOL está en una zona en la que no es capaz de recibir con claridad (*fade zone*), está retrasando la transmisión de paquetes a otros receptores. El efecto que tiene este bloqueo es una disminución del *throughput* y un aumento del retardo.

5.3.2. Disminución de la eficiencia del canal

Si el paquete que vamos a transmitir en un momento dado tiene su canal en un estado de error, intentaremos inútilmente transmitirlo y retransmitirlo hasta R_{max} ocasiones, para al final tener que desistir y eliminarlo. En este espacio de tiempo habríamos podido enviar otros paquetes cuyos canales estuvieran en buenas condiciones. Esto causa una degradación en la eficiencia de transmisión del canal radio.

5.3.3. Desigualdad en el reparto del canal

El tiempo del que dispone cada fuente para usar el canal *wireless*, es función de las características de pérdida del canal de cada receptor. Esta desigualdad en el uso del canal se conoce como *fairness*, y un buen diseño sería aquel, que proporcionase a todas las sesiones una equidad en el reparto de su ancho de banda del canal.

Por tanto vemos como una disciplina FIFO, proporciona bajas prestaciones, provocando aumentos en los tiempos de transmisión y una eficiencia de transmisión pobre.

6. Estación base con varias colas

En esta sección propondremos un sistema de gestión que trate de mejorar en lo posible los problemas que el escenario anterior presentaba. Apoyándonos en el conocimiento de las características del canal *wireless*, utilizaremos un mecanismo de gestión CSDP, (ver Fig. 4). Estas políticas de gestión tratan de aprovechar que los canales son estadísticamente independientes, de manera que si uno se encuentra en mal estado, aplaza la transmisión del paquete y lo intenta con otro que vaya a un receptor distinto. El problema que pueden plantear este tipo de disciplinas es que el tiempo que aplazamos la transmisión sea mayor que el *time-out* asociado en nivel de transporte de TCP, pudiendo cargar innecesariamente el sistema con copias de paquetes. Como ya comentamos con anterioridad, los períodos

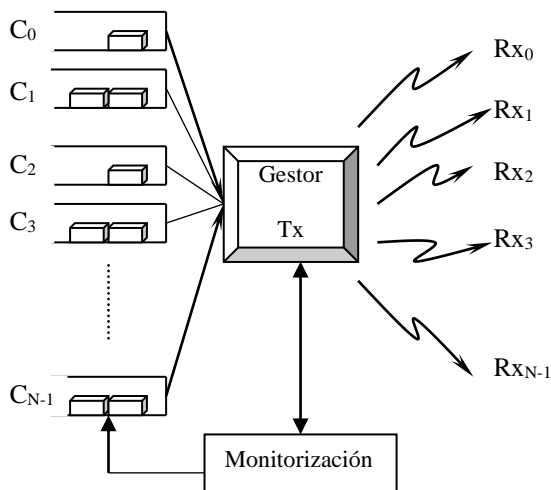


Fig. 4: Configuración CSDP

de *time-out* de TCP son del orden de segundos, mientras que la duración de los *burst* son de milisegundos. Esta diferencia es suficiente para que la retransmisiones del nivel de enlace no alteren las del nivel de transporte.

6.1. Sistema de gestión CSDP

Como vemos en la Fig. 4, el sistema tiene tres componentes.

- Un conjunto de colas, una para cada receptor dado de alta en ese momento en la red.
- Un sistema para monitorizar los estados de los canales de transmisión radio.
- Un gestor de paquetes, que será el encargado de seleccionar la cola de la que transmitir.

En un principio podríamos pensar que sería muy costoso mantener una cola por cada usuario que estuviera dentro del área de cobertura de la estación base. Pero realmente los sistemas WLANs permiten hasta un máximo de 10-15 usuarios, por lo que no es ningún problema.

6.2. Mecanismo de funcionamiento de CSDP

Para cada móvil hay una cola en la estación base. Dentro de ella la disciplina de servicio es FIFO. Sin embargo el servicio a través de las diferentes colas puede ser RR (*Round Robin*), ETF (*Earliest Timestamp First*), LQF (*Longest Queue First*) o cualquier otra. El sistema de monitorización es el que comprueba el estado de los canales de radio. Si el canal entre la estación base y el móvil está en mal estado, marca la cola correspondiente al receptor. Esta marca se realiza, si tras la transmisión de un paquete no recibimos el ACK correspondiente. A la cola se le quita la marca, tras un período estimado de lo que puede durar el *burst*.

De esta forma el gestor CSDP, opera eligiendo un paquete de una de las colas no marcadas. Si están vacías, elige las marcadas. Transmite, y si no recibe un ACK, devuelve el paquete a su cola correspondiente, incrementado los intentos de

transmisión. A continuación selecciona otro paquete en función del tipo de servicio a través de las diferentes colas y vuelve a empezar. Si el número total de intentos de transmisión excede de R_{max} , elimina el paquete.

Este sistema, evita el HOL y mejora la eficiencia del canal de transmisión radio. En cuanto al *fairness*, dependerá del tipo de servicio que usemos entre las colas. El que mejor encaja en estas exigencias es RR, y por ello es el usado en las simulaciones de la siguiente sección.

7. Simulaciones

La herramienta de simulación utilizada es el PTOLEMY. En el primer escenario, tenemos una estación base con una sola cola para 3 y 10 fuentes de tráfico FTP, y en el segundo hacemos uso de la estación base con varias colas, utilizando las mismas fuentes de tráfico que en el caso anterior. Los parámetros usados se muestran en la Tabla. 4.

Hemos elegido un tamaño de paquete de 552 bytes, 512 de información y 40 de cabecera. Se recomienda que el máximo tamaño del segmento TCP (MSS) esté en torno a los 536 bytes (campo de información). De esta forma se busca que el datagrama IP, lleve los segmentos tan grandes como sea posible sin necesidad de fragmentación IP a lo largo del camino entre la fuente y el destino. La elección de los parámetros del enlace cable aseguran la interconexión sin problemas de redes Ethernet con velocidades de 10 Mbps. Y un retardo de 1.6 ms es un valor típico para entornos LAN.

Elijiendo una duración media del periodo bueno, 20 veces mayor que la del malo, y con las probabilidades de pérdida mostradas en la Tabla. 4, modelamos un canal radio en condiciones nominales.

Como se comentó con anterioridad el cuello de botella del sistema se encuentra en la estación base, por eso todas las estadísticas se realizarán en ella y con un número constante de paquetes. Las más importantes van a ser la probabilidad de pérdida y la eficiencia del canal radio. En la última parte de la sección veremos una comparativa de los tiempos de transferencia conseguidos en los dos escenarios.

Tabla. 4: Parámetros de simulación para los 2 escenarios

Transmisor	Velocidad	0.5 Mb/s
	Tráfico	FTP
	Fichero	100 Kbytes
TCP	MSS	512 bytes
	P^B	0.0
Canal Wireless	P^M	0.8
	Media B	1 sg
	Media M	50 msg
	Velocidad	10 Mbps
Canal de cable	Retardo	1.6 msg

7.1. Resultados con 3 fuentes

Para los dos gestores usados, identificaremos como FIFO al escenario con una sola cola y CSDP-RR (*Round Robin*), al otro. Se van a estudiar la probabilidad de pérdida y la eficiencia del canal en función del valor de R_{max} . Éste es un parámetro importante en el protocolo de acceso al medio radio. Cuando a la estación base llega un paquete TCP, el transmisor intenta mandarlo y espera una confirmación por parte del receptor o a que expire el *time-out*. Si sucede esto último, volverá a transmitirlo, incrementando el número de intentos, de modo que cuando se alcanza R_{max} , el paquete se da por perdido y se saca otro de la FIFO.

Como vemos en la Fig. 5.a, conforme R_{max} aumenta las pérdidas reales en el canal son menores. Mientras que para FIFO conseguimos probabilidades de pérdida menores del 1% en R_{max} igual a 8, en CSDP-RR bastaría con un valor de 2. Esto se debe a que las pérdidas en CSDP-RR son mucho menores, al no reintentar la transmisión cuando el canal radio está en mal estado. En la Fig. 5.b se muestra el efecto de esta disminución en el número de retransmisiones. Mientras que para FIFO, la transmisión de una paquete a un receptor, con el enlace en mal estado, se traduce en un uso repetido del canal y por tanto una pérdida de eficiencia, para CSDP-RR el aplazar la retransmisión, permite servir a otra cola con más probabilidades de que su canal esté en buen estado, y así no desperdiciar ancho de banda y tiempo de transmisión, manteniendo una alta eficiencia.

Otro parámetro importante a la hora de elegir un buen sistema de gestión, es estudiar el retardo que éste produce sobre las aplicaciones, tanto si son de tiempo real como si son de tráfico constante. En la Fig. 5.c podemos ver una comparativa de los tiempos de transferencia para un fichero (*FTP*). En este caso se ha fijado el valor de R_{max} a 8, que era el que proporcionaba un buen rendimiento para el compromiso *probabilidad de pérdida vs eficiencia de transmisión*. Como era de esperar, con un valor de R_{max} tan alto, FIFO no obtiene buenos resultados, ya que aunque un solo canal radio, de los que conectan la estación base con alguno de los receptores, esté en mal estado, repercute en el sistema global, aumentando las retransmisiones para ese canal en particular y afectando a los tiempos de transferencia de todas las fuentes.

De esta comparativa que hemos realizado entre FIFO y CSDP-RR extraemos algunas conclusiones.

- Para obtener en FIFO una baja probabilidad de pérdidas hay que sacrificar ancho de banda (eficiencia del canal), cosa que con CSDP-RR no ocurre, ya que además de ser baja para valores de R_{max} más pequeños, la eficiencia se mantiene alta y constante para todos ellos.
- En FIFO, la elección de R_{max} alto (que es lo que se extrae de la conclusión anterior), implica

tiempos de transferencia mayores, al bloquear un canal en mal estado la transmisión a otros receptores.

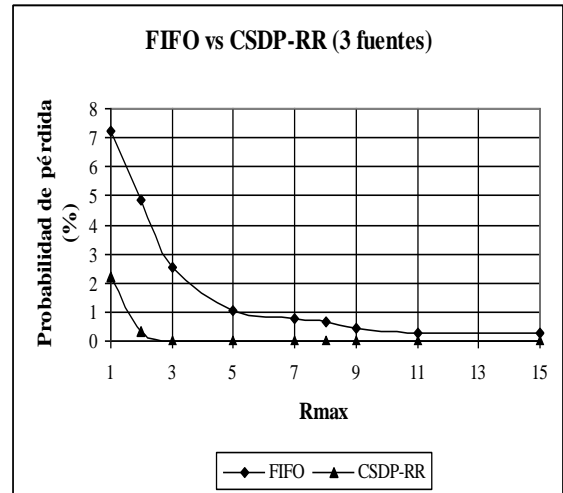


Fig. 5.a: Comparativa de la probabilidad de pérdida para FIFO y CSDP-RR, con 3 fuentes

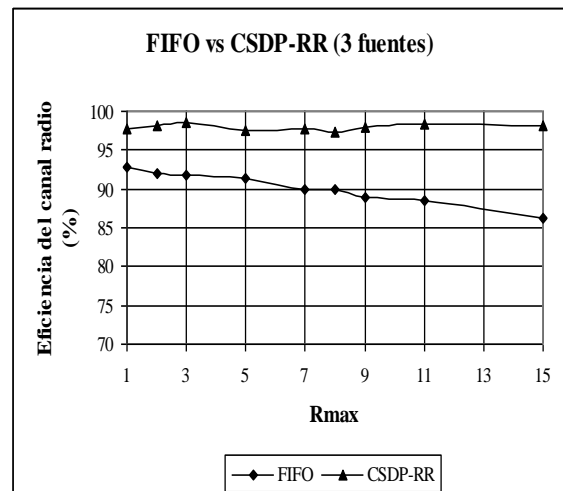


Fig. 5.b: Comparativa de la eficiencia del canal radio para FIFO y CSDP-RR, con 3 fuentes

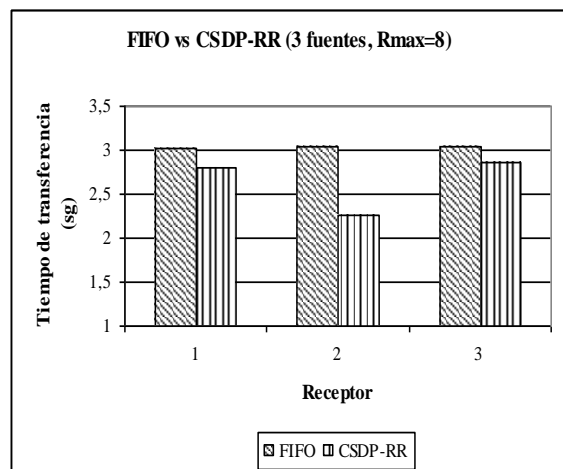


Fig. 5.c: Comparativa de los tiempos de transferencia de un fichero para FIFO y CSDP-RR, con 3 fuentes y $R_{max}=8$

7.2. Resultados con 10 fuentes

Para los mismos parámetros de simulación, a excepción del número de fuentes, presentamos en las Fig. 6.a, Fig. 6.b y Fig. 6.c la comparativa entre FIFO y CSDP-RR.

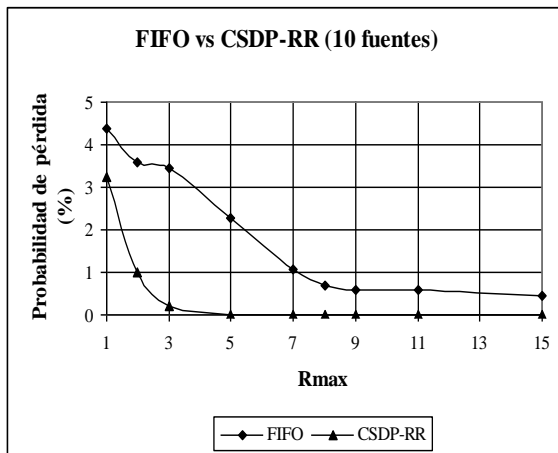


Fig. 6.a: Comparativa de la probabilidad de pérdida para FIFO y CSDP-RR, con 10 fuentes

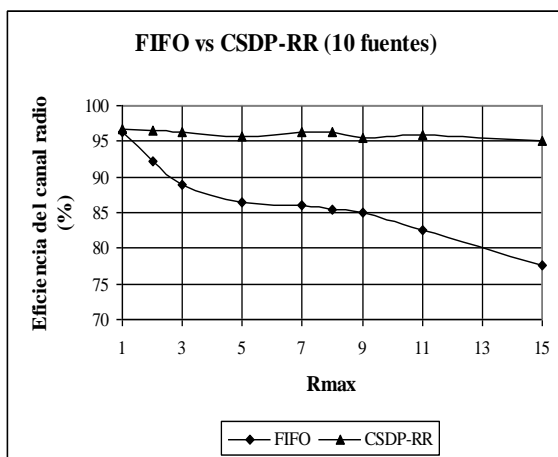


Fig. 6.b: Comparativa de la eficiencia del canal radio para FIFO y CSDP-RR, con 10 fuentes

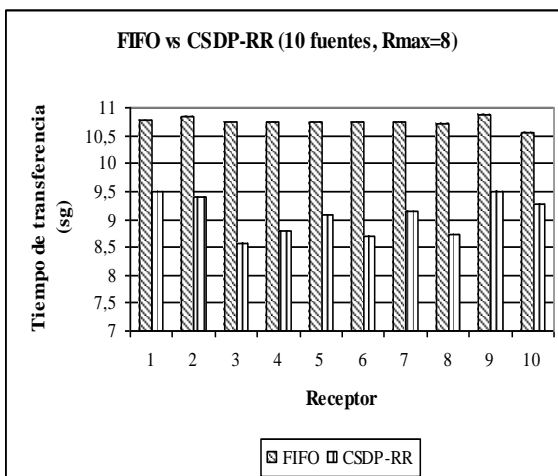


Fig. 6.c: Comparativa de los tiempos de transferencia de un fichero para FIFO y CSDP-RR, con 10 fuentes y Rmax=8

Para la probabilidad de pérdida y la eficiencia del canal obtenemos resultados similares a los anteriores. El caso más interesante es el de los tiempos de transferencia. Usando un gestor FIFO y con un número de fuentes suficientemente grande, los retardos aumentan, al tener todos los paquetes que esperar, en la cola de transmisión, a que finalice el envío y las posibles retransmisiones de todos y cada uno de los paquetes que le preceden. En cambio, con CSDP-RR, al disponer de colas individuales y una gestión diferente, nos aseguramos de que la mayoría de las transmisiones se realizan cuando los canales radio están en buen estado. Así, las retransmisiones disminuyen y los retardos que aparecen, son prácticamente los derivados del sistema que selecciona la cola de la que vamos a transmitir, en nuestro caso RR.

Como consecuencia de todo esto, el tiempo medio de transferencia para FIFO es de 10.749 segundos y para CSDP-RR es de 9.06 segundos. Ganamos en cada aplicación FTP más de 1.5 segundos y con un tamaño de fichero pequeño.

8. Conclusiones

Con este sistema de recuperación de errores a nivel de enlace, basado en el conocimiento de las características del canal *wireless* conseguimos, no sólo mejorar la utilización y el reparto del canal, sino también que las aplicaciones se beneficien de forma directa. Tras una comparativa de los sistemas de gestión FIFO y CSDP-RR concluimos, que parámetros como la probabilidad de pérdida, eficiencia del canal, *throughput* y retardo mejoran sustancialmente sin añadir un exceso de complejidad.

En el futuro, serán motivo de estudio los efectos que diferentes tipos de tráfico, voz, datos y vídeo tienen sobre los sistemas *wireless*, así como la repercusión en la garantía de sus calidades de servicio.

Referencias

- [1] CEPT Recommendation T/R 22-06. "Relating to the harmonized radio frequency bands for HI Performance Radio Local Area Networks (HIPERLANs) in the 5 GHz and 17 GHz frequency range."
- [2] Various authors. Europe Telecommunications Standard (ETSI), functional specification of the High Performance Radio LAN (HIPERLAN) system. January 1995.
- [3] A. Bakre and B. R. Badrinath, "I-TCP: indirect TCP for mobile hosts," in Proceedings of the International Conference on Distributed Computing Systems, (Vancouver, Canada), IEEE Press, 1995.
- [4] R. Caceres and L. Lftode, "The effects of mobility on reliable transport protocols," in Proceedings of the International Conference on Distributed Computing Systems, (Poznan, Poland), IEEE Press, June 1994, pp. 12-20.

- [5] A. DeSimone, M. C. Chuah, and O.-C. Yue, "Throughput performance of transport-layer protocols over wireless lans," in Proceedings of the Conference on Global Communications (GLOBECOM), IEEE Press, 1993, pp. 542-549.
- [6] E. Amir, H. Balakrishnan, S. Seshan, and R. Katz, "Efficient TCP over networks with wireless links," in The First International ACM Conference on Mobile Computing and Networking 1995, (Berkeley, CA), Dec. 1995.
- [7] D. Duchamp and N. Reynolds, "Measured performance of a wireless lan," in 17th Conference on Local Computer Networks, (Minneapolis, MN), IEEE Press, Sept. 1992, pp. 494-499.
- [8] F. Swarts and H. Ferreira, "Markov characterization of digital fading mobile VHF channels," IEEE Trans. Vehic. Tech., Nov 1994, 977-985.
- [9] H. Wang and N. Moayeri, "Finite state markov channel a useful model for radio communications channels," IEEE Trans. Vehic. Tech., Feb. 1995, pp. 163-171.
- [10] "Ptolemy 0.7 User's Manual", Vol. I, University of California, Berkeley 1997.
- [11] "Ptolemy 0.7 Programmer's Manual", Vol. II, University of California, Berkeley 1997.
- [12] "Ptolemy 0.7 Kernel Manual", Vol. III University of California, Berkeley 1997.
- [13] Douglas E. Comer, David L. Stevens, "InternetWorking with TCP/IP" Volume I, Prentice-Hall 1991
- [14] W. Diepstraten, "IEEE 802.11: Wireless Access Method and Physicl Specification, " May 1993. Doc:IEEE P802.11-93/70.

Resumen de Posters

Compresión/Descompresión y Conversión de Archivos de Datos Binarios Phigitales

J.Ramón Glez-Regueral Valdés
Universidad Politécnica de Madrid - ETSI Telecomunicación
Departamento de Matemática Aplicada a las Tecnologías de la Información
E-mail: regueral@qualia.es

Abstract

The special features of the Phigital system numbers are applied here for the substantial compression of canonical alphanumeric expressions in order to provide file structures fitted for transmission, reception and decompression, as well as for conversions from/into 2-based conventional binaries or any other number systems. The Phigital system is based on the irrational number Phi (Φ). It is a positional binary number system that provides a valid alternative in many present and future applications. The algorithms developed from this theoretical context for all the necessary mathematical operations obtain absolutely accurate results using only natural numbers, in spite of the irrational nature of the basis of the system. This system furthermore provides new resources of applications in the techniques designed for improving cryptography and safety in communications.

Herramienta de gestión de proyectos software mediante técnicas hipertexto

Alvaro Suárez, Elsa M^a Macías, Laura Viera
GAC (Grupo de Arquitectura de Computadores)
Dpto. de Ingeniería Telemática Universidad de Las Palmas de G.C.
Campus Universitario de Tafira (Aula Magna), 35017 Las Palmas de G.C.
Telf: 928-45-12-39, Fax: 928-45-12-43
email: {alvaro, elsa}@cic.teleco.ulpgc.es

Abstract

In general, the development of a software design requires the participation of several people who usually work in different places independently. The software design should be well-documented to make easier its upkeep. In this paper we present a tool for the management of software designs using hypertext techniques (HTML). The tool permits: the generation of documentation in any stage of the software design, generation of documentation about an ended project, the exchange of information through communication networks and software reutilization.

Propuesta de un Protocolo de Sincronización de Flujos Multimedia sobre RTP (*Real-time Transport Protocol*)

Fernando Boronat Seguí, Juan Carlos Guerri Cebollada, Manuel Esteve Domingo, Carlos Palau Salvador
Area de Ingeniería Telemática, Departamento de Comunicaciones
Universidad Politécnica de Valencia - Escuela Universitaria de Gandía
Ctra. Nazaret-Oliva S/N, 46730 Grao de Gandía (VALENCIA)
Telf: 96 284 93 41, Fax: 96 284 93 13
E-mail: fboronat@ocom.upv.es,

Abstract

The Distributed Real-Time Systems Research Group, in which we are involved works in synchronisation protocols for multimedia systems. A new method for multimedia flows synchronisation is presented in this paper. It is based in the combination of RTP (Real-time Transport Protocol) and a protocol developed within the group, called Feedback Global Synchronisation Protocol, based in Global Time and feedback techniques. The protocol has been evaluated in two different real environments: WAN and LAN environments. Moreover, we have evaluated the synchronisation quality using subjective methods.

Diseño de un entorno de Outsourcing para la Gestión de Redes TCP/IP basado en la plataforma Spel 2.1 de Cabletron

Julio Miró Borrás, Francesc Xavier Blanes Molina
Departamento de Comunicaciones
Escuela Politécnica Superior de Alcoy
Universidad Politécnica de Valencia
Pza. Ferrándiz Carbonell s/n, 03801
Alcoy (Alicante)

Abstract

Nowadays most companies need network infrastructures for their development and productivity objectives, but they are not telecommunications companies, so they do not have specialized technicians to carry out the administration of their networks. Keeping in mind this, Outsourcing arises with the idea of solving this question. The network management Outsourcing lets a telecommunications company to take care of the network management of their client companies. In this report we present the design of an Outsourcing framework for Network Management based on Cabletron Spel 2.1 platform and an application developed for the transmission of event information between the Outsourcing company and its client enterprises.

Evaluación de las prestaciones de una red acceso radio punto multipunto para distintos servicios de datos

C. Téllez Labao, E. González Parada, A. Reyes Lecuona y A. Díaz Estrella
Dpto. Tecnología Electrónica, ETSI Telecomunicación, Universidad de Málaga.
Campus Teatinos 29071, Málaga
Telf: 952132844, Fax: 952131447
E-mail: tellez@dte.uma.es

Abstract

This work presents a method to distribute the radio resources remaining after allocating all guaranteed services among competing best-effort applications traffic sources. Our goal is to optimise the resources allocation to non- guaranteed services by using message size-based resource scheduling algorithms, which has proven to be a very efficient method.

Metodología para el Desarrollo de Bancos de Pruebas para Sistemas de Comunicaciones

Javier Poncela González, José Tomás Entrambasaguas Muñoz
Dpto. de Ingeniería de Comunicaciones Universidad de Málaga
Campus de Teatinos, s/n, 29071 Málaga
Telf: 95-2132759, Fax: 95-2132027
E-mail: poncela@ic.uma.es, jtem@ic.uma.es

Abstract

Conformance testing plays an important issue in the development of telecommunication equipment. In this paper we introduce a methodology covering the whole development cycle for the efficient implementation, both in time and cost, of test benches for communication systems. It is shown how this can be achieved having as input one or more TTCN test suites. The SDL formal language will be used for the implementation of the lower layers in the test system as well as an emulator of the system under test. An example of implementation using a reduced functionality of DECT is also shown.

Análisis de DIS y VRML para un Entorno de Simulación Distribuida Aplicada a la Navegación de Sistemas Autónomos Móviles.

M. Sánchez Alonso, J.M. García Carrasco Antonio F. Gómez Skarmeta, H. Martínez Barberá
Dept. Ingeniería y Tecnología de Computadores Dept. Informática, Inteligencia Artificial y Electrónica
Universidad de Murcia
Apartado Correos 4021, 30001 Murcia
Telf: 968-364607, Fax: 968-364151
E-mail: {skarmeta,msanchez}@dif.um.es

Abstract

In this paper we present the problems associated to the design of simulator for complex models, and the solutions that the Distributed Simulations Environments offer to this kind of activities. We propose the development of tool that consider not only the use of the Distributed Interactive Simulation protocol (DIS) but also its interaction with VRML, and the semi-automatic generation of distributed simulators starting from a simulation language. In order to analyse its possibilities we present the application of approach to the simulation of the movement of an autonomous robots, where different distributed component as the navigation, or the environment interact through a virtual world.

Arquitectura de publicación automatizada de contenidos educativos supervisados en Internet.

¹A. Arin, ²E. Magaña, ³J.J. Astrain, ⁴Jesús Villadangos, ⁵J.R González de Mendivil
Departamento de Automática y Computación
Universidad Pública de Navarra
Campus Arrosadia s/n. 31006 Pamplona (Navarra)
Telf: 948-168904, Fax: 948-169281
E-mail: {¹[ma10747](mailto:ma10747@zurron.upna.es), ³[ja11066](mailto:ja11066@zurron.upna.es)}@zurron.upna.es, {²eduardo.magana, ⁴jesusv, ⁵[mendivil](mailto:mendivil@upna.es)}@upna.es

Abstract

This paper presents an automated publishing architecture of educational and supervised contents over the Internet. The system makes easier the job of publishing educational courses over the network using specific tools that automates the access control (CGI -- Common Gateway Interface), encrypts the information that goes through the net for not being accessed by strangers (SSL – Secure Sockets Layer), gives an statistical control of the usage of the system, and tutors the course. This project is multiplatform, i.e. all the components that integrate the system are available for any operating system (Windows 9X, Solaris, Linux,...) and it is based on the Web.

Facultad virtual con servicios autoperpersonalizados y tecnología push

Juanjo Unzilla, Eduardo Jacob, Luis Zabala, Sergio Oslé
Grupo de Ingeniería Telemática. E.T.S.I.I. e I.T. de Bilbao (UPV/EHU)
Alda Urquijo s/n, 48013 - Bilbao
Telf: 94 601 4206, Fax: 94 601 4259
{[jtpungaj](mailto:jtpungaj@bi.ehu.es), [jtpjatae](mailto:jtpjatae@bi.ehu.es), [jtpzaall](mailto:jtpzaall@bi.ehu.es)}@bi.ehu.es | jtaosvas@aintel.bi.ehu.es

Abstract

This paper presents the design and implementation of a virtual campus whose purpose is to reinforce the traditional educational system. This campus offers a number of services, with a central common kernel, all supported by a Linux server and accessible remotely through a common web browser. The system main distinctive features are its user-oriented approach and its security policy, guaranteed by the SSL protocol and the use of certificates. To optimize the network and machines resources, several protocols and languages are employed, from TCP to UDP and from classic CGI scripts to JAVA and JavaScript.

Sistema Distribuido para la Captura y Reproducción Sincronizada de información Multimedia en un entorno Médico

Juan Carlos Guerri, Manuel Esteve, Manuel Zaera, Carlos Palau
Departamento de Comunicaciones
Universidad Politécnica de Valencia
46022 Camino de Vera, S/N
Telf: 96 387 7743, Fax: 96 387 7309
E-mail: jcguerri@dcom.upv.es

Abstract

Frequently, the assessment of the physical condition of a person depends on the evaluation of different tests. Different devices perform the data acquisition in these tests. All this information must be analysed together and simultaneously in order to extract conclusions, so a temporal relationship has to be established between the data sources. In this work, the synchronisation process of the sources of information is obtained by synchronising the clocks of the computers that are connected by means of a Local Area Network (LAN), using a clock synchronisation protocol and stamping the global time in each information data flow. During the analysis, the aforementioned time-stamp is used to playback the data in a synchronised way. The proposed system is valid for any system in which data synchronisation was needed. At present, it has been used in a distributed medical environment for the analysis of electrical muscular activity and lumbar movement, with the aim of detecting injuries and planning the sportsmen training. The easy management of the application and the precision of the synchronisation protocol, have allowed the extraction of interesting medical conclusions.

Herramientas para la generación de cursos de calidad a través de Internet para formación continua

Manuel Esteve, Carlos Palau, Juan Carlos Guerri, Manuel Zaera
Área de Ingeniería Telemática. Departamento de Comunicaciones
Universidad Politécnica de Valencia
Con. de Vera s.n. Valencia 46071
Tel. 96 387 73 05 Fax. 96 387 73 09
E-mail: mesteve@dcom.upv.es

Abstract

Continuous education is an important target in the lifelong professional activity development, especially, in engineering field. Telematic technology gives a good way to develop this target. This paper presents a complete process for generating multimedia teaching materials and on line courses generation for continuous training over the Internet. The process is based on three elements: a tool for teachers – called the IFTA or InterFad Teacher Assistant – for automatically generating the courses and showing the relationships between the various multimedia objects included in the course; an application that uses a database generated by the teacher's tool to make a WWW application; and thirdly, an information and multimedia object database that supplies the web application and can be accessed by students using a standard navigator over the Internet. The paper describes also the user interface of the teachers tool and the general system configuration. The entire process, with supporting tools, is currently being used at the University Polytechnic of Valencia as part of the Interfad project.

Un nuevo modelo de Seguridad para Sistemas Distribuidos y herramienta de Diseño, Gestión y Simulación de políticas basado en el mismo.

Eduardo Jacob, Juanjo Unzilla, Alejandro Muñoz
Departamento de Electrónica y Telecomunicaciones – Area de Ingeniería Telemática
ETSIT e IT de Bilbao, Euskal Herriko Unibertsitatea – Universidad del País Vasco
Alameda de Urquijo S/N, 48013 Bilbao
Telf: 946014214, Fax: 944414041
{jtpjatae,jtpungaj}@bi.ehu.es, jtamumaa@bipt106.bi.ehu.es

Abstract

Designing and evaluating security policies is one of the main concerns of people involved in the management of local area and corporate networks. This article introduces a new security model that gives the possibility to represent and study complex interrelations between elements and how they modify the level of security of a given infrastructure. We also present a tool that we are developing that based in this model is useful in designing, managing and simulating security policies over distributed environments. The tool offers the possibility for simulating small distributed systems, showing present security level and possible improvements by means of changes in its elements.

Proyecto EDAO: Evaluación Docente Asistida por Ordenador

Eduardo Villoslada de la Torre, Juan J. Villacorta
e-mail: eviltor@cantosantel.uva.es
Grupo de Tecnologías Multimedia
E.T.S. de Ingenieros de Telecomunicación
Universidad de Valladolid

Abstract

The improvement of telecommunication resources and infrastructures must complete with the improvement of services. It is our mission to approach people to the new technologies, by means of the creation of new services and the spreading of the rest. We have chosen the tele-education to develop a new application that make fast and easy the evaluation of the knowledge of students or people. First, this article relates the tele-education topic, with some comments about its future. Then, it is described the system and the developed applications with their main features.

Implementación de un Servidor de Información en Internet bajo Situaciones de Carga Elevada

Fidel Cacheda Seijo, Alberto Pan Bermúdez, Lucía Ardao Rodríguez, Ángel Viña Castiñeiras
Departamento de Electrónica y Sistemas
Facultad de Informática, Universidad de A Coruña
Campus de Elviña s/n, 15.071 A Coruña
E-mail: {fidel, alberto, lucia, avc}@des.fi.udc.es

Abstract

Internet information systems are being widely used nowadays, which produces high workload situations and easily changeable requirements. In this way, information systems should be built using a dynamic, secure and robust architecture. In this paper we present an architecture suitable for Internet information systems, distributed in three layers, which let create a more flexible and robust system. Finally, we describe the implementation details of an information system in a real highly workloaded environment. We have used new technologies to develop our system, such as Java, which allow us to create multiplatform information servers. We use a relational database management system to store the high amount of information of our system, in order to achieve a better performance.

The system is nowadays working in Internet, and can be visited at: <http://www.biwe.es>

Software de Simulación de Sistemas de Comunicación

Javier Santana Estupiñán, Itziar Alonso González
Dpto. de Ingeniería Telemática
35017 - Campus de Tafira
Universidad de Las Palmas de Gran Canaria
Email: ialonso@cic.teleco.ulpgc.es

Abstract

Nowadays, there a lot of tools of simulation which are in the most of labs. Our goal has been to develop our own tool for an easy apprenticeship about communication systems. This software application is a visually programmed object oriented simulation software. By arranging and connecting icons, signal processing algorithms as well as many other types of simulations can be programmed, adjusted and easily tested. This application works in a graphical environment.

Sistema de medidas remotas usando Internet

M. Cervera, F.J.Gómez, J. Martínez
Lab. de Microelectrónica. E.T.S. de Informática y
Dpto. de Física Aplicada. Universidad Autónoma de Madrid.
Campus de Cantoblanco, 28049 Madrid
Telf: 91-3978607, Fax: 91-3974895
E-mail: Javier.Martinez@ii.uam.es

Abstract

In this paper we present a vehicle for a complete engineering and scientists practical formation. It is made possible by the use of virtual instrumentation, not in the sense of a simulation tool, but as the remote access to facilities located in related research laboratories. Internet World Wide Web provides a best way to materialize this remote access, reducing the requirements in the educational laboratory to workstations (WS) or personal computers (PC) with the habitual Internet access.

Agentes Móviles Basados en Java para la Gestión de una Plataforma de Comunicaciones Telefónicas

Arribi J.
Dpt. de Electrónica y Sistemas de la Universidad de A Coruña
Campus de Elviña. 15071. Coruña. España
Teléfono: 981-167000
Fax: 981-167160
jesus@gris.des.fi.udc.es

Abstract

Thanks to the popularity of the WWW and due to the increasing aceptation of Java as language for the development on Internet, the number of distributed applications on this important platform is increased. The advantages of an intuitive graphic user interface, like web pages, and the facilities that Java gives to make distributed programming via RMI, and to send classes with the serialization, have changed the monolithic and traditional applications which have given way for more advanced and distributed architectures. In this paper is showed the convergence of Web and Java technologies in order to develop a distributed architecture based on mobile agents, that provides telephonic services. These services are, on the one hand, remote and automatic programming of PBX (Private Branch Exchange) and, on the other hand, storing and querying of telephonic costs via Web.

Diseño de un Sistema de Recepción Digital Multimedia

José Juan Pazos Arias, Manuel Ramos Cabrer y Cándido López García
Area de Ingeniería Telemática Universidad de Vigo
ETSI Telecomunicación. Campus Universitario s/n. 36200 Vigo (PONTEVEDRA)
Telf: 986-812186, Fax: 986-812116
E-mail: {jose,mramos,candido}@ait.uvigo.es

Abstract

The aim of this article is to present the design of a digital multimedia reception system which, apart from the classical functions of digital signal reception and processing, includes other which are supposed to be much more attractive for final users: interactive (pay per view, software downloading through the broadcasting signal, choosing the position of the camera in sports events, etc.) and communication facilities (Internet access, teleshopping, electronic bank services, reception of commercial messages, database access, etc.) . The ultimate goal of this system is the gradual transformation of conventional TV receivers into personal computers.

Telemonitorización de la “Pérgola Fotovoltaica” vía Internet. Primeros resultados de operación.

S.G. Galán, J. Cañada Bago, P.J. Pérez, G. Almonacid
GRUPO JAÉN DE TÉCNICA APLICADA
Alfonso X El Sabio, 28. 23700. Linares (Jaén)
Tel.: 953.649556. Fax: 953.649508
E-mail: sgalan@ujaen.es, jcbago@ujaen.es, pjperez@ujaen.es, galmona@ujaen.es

Abstract

Communication networks, Internet to be precise, allow the implementation of monitoring systems and remote controls of photovoltaic installations by means of different communication technologies. This paper describes the monitoring system structure of the "Photovoltaic Pergola", a 2kWp grid connected photovoltaic system, which is located in the Solar Energy Laboratory at the High Polytechnic School in Jaen University. It also presents the different tools that have been developed that make possible the remote monitoring of the photovoltaic installation, and the features that they offer, showing the working results obtained up to the moment. Besides, the paper presents the changes that are being carried out in the structure of the monitoring system, to adapt it to a new installation of higher magnitude, "UNIVER", which is among the very first ten in Europe.

Modelo de Dimensionado de Centros de Certificación

Albert Hurtado Blasco, Jordi Forné Muñoz
Dpto. Matemática Aplicada y Telemática
Universitat Politècnica de Catalunya
C/Jordi Girona, 1-3
08034 Barcelona
E-mail: jforne@mat.upc.es, ahurtado@mat.upc.es

Abstract

The problem of secure communications over insecure networks has been deeply studied over the last decade. Public-key cryptography and certification authorities are necessary to provide security services over a large open network, like Internet. In this paper we present a model to dimension the certification centres. This is a simple model based on queuing theory that can be used to choose the suitable server for a certification centre at a given security level.

Acceso al Adaptador de Red desde Win32 (Windows NT y 9X): Arquitectura y Aplicaciones

Francisco Gómez Marcos, José Ignacio Moreno Novella
Area de Ingeniería Telemática, Universidad CARLOS III de Madrid
Avda. de la Universidad 30, 28911 Leganés (MADRID)
Telf.: 91-6249431, Fax: 91-6249430
E-mail: gomez@beasys.com, jmoreno@it.uc3m.es

Abstract

Network access in the Win32 environment is usually provided through standard high-level APIs like Sockets and NetBIOS. A low-level access API, by contrast, fits certain application requirements like network protocol design, network management and network engineering training at university degree. This work provides such a low-level access library in the Win32 environment, supporting both Windows NT 4.0 and Windows 95&98, in spite of their different implementations. Two platform-specific kernel-mode NDIS drivers and a portable user-mode library (for both dynamic and static linking) support the API, giving the developer an easy-to-use language-independent network card access toolkit, similar to the well known ancient MS-DOS Packet-driver, but much more robust and powerful. A set of network applications developed in C, C++ and JAVA using the API is also provided, complexity ranging from lab exercises to real world network managing programs.

Análisis de una Metodología de Diseño de Redes Conmutadas de Área Local.

Esteve Pallarés Segarra, Luis J. De la Cruz Llopis y Joan García Haro
Departamento de Matemática Aplicada y Telemática
Universitat Politècnica de Catalunya (UPC)
C/ Jordi Girona 1 y 3. 08034 Barcelona
Telf: 934016835, Fax: 934015981
E-mail: esteve@mat.upc.es

Abstract

The deployment of new telematic services requiring high bandwidth is impacting on the conventional telecommunication networks. Traditional networks are not well-suited to support this growing demand of bandwidth. This fact is first manifested in Local Area Networks and the solution was the increase of link capacity as well as the segmentation in several LANs. Consequently, LANs that originally were shared-medium, shared-bandwidth ones are becoming switched networks. In this paper, a method to design these new topologies are examined and carefully analyzed after implementing it efficiently in software.

Definición de una misión aeroespacial en el País Vasco con microsátélites de órbita baja y su aplicación telemática

Armando Ferro, Marivi Higuero, Luis Zabala, Eduardo Jacob, Juanjo Unzilla
{jtpfevaa|jtpphiapm|jtpzaall|jtpjatae|jtpungaj}@bi.ehu.es
Grupo de Ingeniería Telemática
Escuela Técnica Superior de Ingenieros Industriales y de Telecomunicaciones de Bilbao

Abstract

This paper presents a project studying the viability for the building of a microsatellite in the CAPV enviroment. Because of the importance of the systems based on satellites are acquiring, this project arose as the result of the interest of several working groups from different sectors, and with the objective of develop the technologies related to these systems in our area. It has been defined an strategy based on the following key elements: definition of a reasonable mission, development of the basic platform for that mission, and economical financing program for the proyect. The most important areas of telematic interest we have considered are 'Store & Forward' services, communications protocolos used between the terrestrial control station and the satellite itself, software components for the control station, and several areas related with satellite constellations like routing algorithms, or management of satellites networks.

Estimación de carga de tráfico aplicada al encaminamiento de llamadas.

A. Ariza Quintana, M^a Carmen Oria, F. Sandoval
Departamento de Tecnología Electrónica, E.T.S.I. Telecomunicación
Universidad de Málaga, Campus de Teatinos S/N 29013 (Málaga)
Telf: 952132728, Fax: 952131447
E-mail: alfonso@dte.uma.es

Abstract

In this work we present two modifications to the routing algorithm ASDR that allow improving its operation by means of the improvement of the estimate of the offered traffic. The first modification consists on improving the traffic measure by means of the employment of estimate windows overlapped in the time. The second one, more advanced and complex, consists on the employment of the prediction of traffic load being based on the recurrent behavior that presents the traffic offered to the nets of current communications. If the traffic presents a periodic behavior it is possible to predict which will be the traffic in the net. These modifications of the basic algorithm are also expandable to other routing algorithms and to the assignment of virtual path bandwidth.

Herramientas y Protocolos para Edición Cooperativa en Red

Carlos Ley, Alvaro Suárez, José L. Pérez, José Hernández
GAC (Grupo de Arquitectura y Concurrencia)
Departamento de Ingeniería Telemática, Universidad de Las Palmas de Gran Canaria
Campus de Tafira S/N, 35017 Las Palmas
Telf: 928-45 89 70, Fax: 928-45 12 43
E-mail: {alvaro, cley}@cic.teleco.ulpgc.es

Abstract

Cooperative work using networked computers is an important topic in Telematics. The main idea behind Cooperative work is that a group of users can achieve a task collaborating in order to accomplish a common goal. There are many fields covered by these so-called cooperative applications, and in this paper we deal specifically with document editors that let users access and modify the same file concurrently. We present some solutions to the problem of collaborative writing in a computer network environment. Two different tools have been developed for implementing these solutions. Also a new edition tool under development is presented that is planned to work using the World Wide Web interface.

Estudio e Implementación de un Modem Eficiente para Comunicaciones PowerLine.

Noelia Cabezón Pellejero <noelia@sunul-1.ita.es>
Francisco J. Cortés Santaolalla <fcortes@sunsp-3.ita.es>
Departamento de Tecnologías Avanzadas en Telecomunicaciones.
Area de Electrónica y Nuevas Tecnologías,
María de Luna 7, 50015 Zaragoza
E-mail: telematica@sunul-1.ita.es

Abstract

This paper presents AC power wiring like a Communications Network. The sources of attenuation, noise and distortion encountered when communicating over Power Lines, and a model suitable to describe that noise using measurement results are reviewed. A lineal and a constant envelope efficient modulations are studied, and the bit error rate performance in the PowerLine environment is calculated by a simulation. Nowadays, PowerLine Communications are very slow. The results show they are possible with a higher bit rate. A $\pi/4$ -DQPSK modem is implemented in a Digital Signal Processor (DSP), and finally, a complete prototype is proposed.

Generación de escenarios: Planificación de la Estructura Lógica y de las topologías de Acceso y Backbone (SWINET)

Alberto Eloy García Gutiérrez, Klaus D. Hackbarth
Grupo de Ingeniería Telemática de la Universidad de Cantabria
Avda. los Castros s/n, 39005 Santander (Cantabria)
Telf: 942-201494, Fax: 942-201488
E-mail: agarcia@tmat.unican.es, klaus@tmat.unican.es

Abstract

SWINET is a strategic planning tool which considers a specific list of client access nodes characterised by corresponding parameter values for each access area. Nodes are classified by a heuristic algorithm based on the total node traffic demand and the geographical distance allowing that the level of some nodes are already predetermined and resulting a two level network composed by concentrators in the access layer and switching unit in the backbone one. The node traffic demand of each service class is then distributed over this network topology obtaining corresponding list of links between each set of nodes, both access and backbone ones.. As a result of this calculation SWINET gets all access and backbone links considering a star, mesh, tree or ring structure for the access network and a meshed/star structure for the backbone one, allowing generation of sets of different network scenarios will be used by other planification tools.

Herramienta para la mejora de distribución de tareas PVM sobre una LAN

Domigo Marrero, Alvaro Suárez
Area de Ingeniería Telemática. Universidad de Las Palmas de Gran Canaria
Campus Universitario de Tafira s/n. Edificios de Telecomunicación.
35017 Las Palmas de Gran Canaria
Tfno. (928)451223, Fax: (928)451243
E-mail : dmarrero@det.ulpgc.es

Abstract

This paper presents an application based in the Client-Server Model. The application polls status report of target hosts on the network. The client is a Graphic Interface for Xwindows and Xview that its requests status information to servers. The responses will be displayed and tested for the program and the user. The comparative received from all the servers can be compared and decide which host has the best performances. One of the servers is made up of four remote procedure call (RPC) with external Data Representation (XDR) (Sun Microsystems) and the other one can execute commands or receive files from the clients.



COIT/AEIT



Asociación de
Técnicos de
Informática



patrocinadores
oficiales



FUNDACION



CICYT

La pública diferencia



THE ENTERPRISE MIDDLEWARE SOLUTION

ISBN: 43554366456464

Depósito Legal: M-435445