TESIS DOCTORAL

Contribución al soporte de la clase de servicio ABR en redes ATM mediante técnicas de planificación equitativa

TESIS DOCTORAL

Contribución al soporte de la clase de servicio ABR en redes ATM mediante técnicas de planificación equitativa

Luis Alejandro Guijarro Coloma
Ingeniero de Telecomunicación

Director:
Jorge Martínez Bauset
Doctor Ingeniero de Telecomunicación

Area de Ingeniería Telemática
Departamento de Comunicaciones
Universidad Politécnica de Valencia

Impreso en mayo 1998

Este texto ha sido compuesto en Palatino y Helvética por el autor, mediante LaTeX2$_\varepsilon$.

A Marga

If a man will begin with certainties, he shall end in doubts; but if he will be content to begin with doubts, he shall end in certainties.

Francis BACON
The Advancement of Learning 1605

Agradecimientos

Durante la elaboración de esta Tesis he tenido la inmensa fortuna de contar con la colaboración de muchas personas. Quisiera dar las gracias de modo especial a Jorge Martínez Bauset y a José Ramón Vidal Catalá por haber sido mis compañeros en el grupo de investigación InterATM y a la vez mis tutores en la investigación que he llevado a cabo en esta Tesis. Además, ellos han leído la memoria y han hecho multitud de sugerencias.

Quisiera mostrar mi agradecimiento a Jorge Mataix Oltra, a Antonio Alabau Muñoz y a Vicente Casares Giner, quienes han ocupado sucesivamente el cargo de Coordinador del Área de Ingeniería Telemática desde que inicié mis estudios de Doctorado en el Departamento, por haber apoyado siempre la investigación de los doctorandos en el Área.

Además, quiero dar las gracias a Manolo Tomás Valero, por los frecuentes y fructíferos intercambios de impresiones que hemos tenido, al respecto de la investigación doctoral y de otros muchos temas.

Finalmente, me gustaría agradecer a Marga su infinita paciencia durante las incontables horas que he pasado absorto en la elaboración de esta Tesis, especialmente durante la confección de la memoria. Sin su comprensión y apoyo no hubiera sido capaz de llevar a cabo esta empresa.

TESIS DOCTORAL
Contribución al soporte de la clase de servicio ABR en redes ATM mediante técnicas de planificación equitativa

TRIBUNAL CALIFICADOR

Presidente:	**Dr. Vicente Casares Giner**
Vocal:	**Dr. Jorge García Vidal**
Vocal:	**Dr. Javier Aracil**
Vocal:	**Dr. Jordi Mata Díaz**
Secretario:	**Dr. Jorge Mataix Oltra**
Vocal suplente:	**Dra. Olga Casals Torres**
Vocal suplente:	**Dr. Josep Solé Pareta**
Vocal suplente:	**Dr. Joan García Haro**

Realizado el acto de lectura y defensa de la tesis doctoral el día 22 de mayo de 1998.

Se otorga la CALIFICACIÓN de:

El Presidente Los Vocales El Secretario

Índice General

Índice de Tablas

Índice de Figuras

Capítulo 1

Introducción

Esta Tesis doctoral aborda el estudio de los mecanismos de red utilizados para la provisión del tipo de servicio denominado de "de buenas intenciones", en adelante, servicio de tipo *best-effort*, en redes ATM. La contribución de esta Tesis es un algoritmo de conmutador para el soporte de la clase de servicio *Available Bit Rate* (ABR), que es una de las clases de servicio *best-effort* definidas por el ATM Forum para la capa ATM. La provisión de esta clase de servicio se basa en un control de flujo por realimentación de tasa desde la red a las fuentes. En este control intervienen tanto los sistemas finales —fuente y destino— como los nodos de la red —conmutadores—. Un algoritmo de conmutador para el soporte de ABR tiene la función de generar el valor de tasa al que debe emitir cada una de las conexiones que atraviesan el conmutador. El algoritmo debe ser capaz de generar un valor de tasa óptimo en términos de eficiencia y de equidad.

1.1 Planteamiento de la Tesis

En las redes ATM se consigue proveer de una manera integrada distintos servicios sobre una única plataforma de red. Las redes ATM operan en modo orientado a conexión y segmentan toda la información en paquetes cortos de longitud fija denominados *células*. El procesado de las células en el interior de la red se ha minimizado, de modo que la adaptación del servicio de capa ATM a cada una de las aplicaciones que se soportarán sobre la red ATM se lleva a cabo en los sistemas finales, mediante el protocolo de adaptación a ATM adecuado.

Las redes ATM pueden proveer servicios a aplicaciones con muy distintos requisitos de calidad de servicio. Esta calidad de servicio se suele concretar en términos de retardo máximo de transferencia y de tasa máxima de pérdida de células. Para que ello sea posible, a nivel ATM deben operar distintos mecanismos de control de congestión[1], tales como control de admisión, control de parámetros de usuario, control de flujo, descarte selectivo de células, planificación de células, etc.

[1] Aunque el UIT-T en la Rec. I.371 distingue entre los términos *control de tráfico* y *control de congestión*, en la mayoría de la bibliografía consultada se emplean indistintamente. Nosotros emplearemos el término *control de congestión*.

Algunas aplicaciones requieren del servicio de capa ATM un retardo de transferencia máximo garantizado. Para ello, la red ATM debe reservar recursos durante la duración de la conexión. Por su parte, el usuario debe comprometerse en tiempo de establecimiento de conexión a respetar un patrón determinado de tráfico. Si la red no puede reservar los recursos necesarios para garantizar los parámetros de calidad de servicio que requiere el usuario, no aceptará el establecimiento de la conexión. Existen aplicaciones, no obstante, que no precisan una calidad garantizada de servicio, por lo que sería deseable un servicio más económico, esto es, que no reservara recursos. Incluso otras aplicaciones, aun precisando cierta calidad de servicio, no pueden prever las características del tráfico de células que van a generar durante el tiempo de vida de la conexión, por lo que se verían obligadas a sobredimensionar el patrón de tráfico que declaran, dando lugar a una infrautilización de los recursos que se reservaran.

Para este último tipo de aplicaciones, se planteó la necesidad de proporcionar un servicio en la Red Digital de Servicios Integrados de Banda Ancha (RDSI-BA) con características similares al servicio ofrecido por las redes de área local IEEE 802 o por el protocolo IP. Para ello se procedió a la definición de un servicio ATM de tipo *best-effort*. Tradicionalmente el servicio *best-effort* se ha asimilado al que proporciona el protocolo IP en la Internet. En ésta, la red ofrece el mejor servicio posible en cada momento, en cuanto que pone todos los recursos de la red a disposición de los usuarios, principalmente ancho de banda en el enlace y espacio de almacenamiento en los nodos. En las redes ATM, los recursos no reservados para el soporte de aplicaciones de servicio garantizado podrían ofrecerse según este paradigma *best-effort* a la aplicaciones con requisitos débiles de calidad de servicio. Para permitir esta posibilidad, el ATM Forum ha definido la categoría de servicio *Unspecified Bit Rate* (UBR). En UBR, cuando se detecta congestión, la red descarta células en los nodos.

No obstante, en las redes ATM se dan dos circunstancias novedosas que permiten ofrecer un servicio *best-effort* más potente.

En primer lugar, las redes ATM son redes con una tecnología homogénea, a diferencia de la Internet, que está constituida por subredes de distinta tecnología. Ello permite que los nodos de la red ATM puedan modificar el contenido de las células de cada conexión, a diferencia de la Internet, en donde los nodos de las subredes no pueden procesar los datagramas. Esta potencialidad se ha materializado en la definición de la clase de servicio ABR, cuyo mecanismo de soporte es un esquema de control de flujo por realimentación explícita del valor de tasa que la red permite a cada conexión. En ABR, la calidad de servicio que recibe en cada momento el usuario puede, por tanto, ser percibida explícitamente, mientras que en UBR sólo puede ser percibida implícitamente a través de la medida del retardo experimentado por las células en la red y a través de la detección de la pérdida de células. Así, el servicio ABR, que es de tipo *best-effort*, ofrece una calidad de servicio que la red comunica al usuario en cada momento.

En segundo lugar, las redes ATM se han concebido desde un principio como redes públicas, en donde la provisión de servicios tiene un carácter comercial, a diferencia de la Internet, cuyo mantenimiento estaba financiado inicialmente por organismos públicos. Ello obliga al operador de la red ATM a ofrecer a cada usuario la garantía de que la calidad de servicio que está recibiendo, aun en el caso de servicio *best-effort*, no va a quedar

perjudicada por un comportamiento no cooperativo por parte de otros usuarios. Esta garantía no es factible en UBR, que traslada directamente el paradigma *best-effort* tradicional. En cambio, el control de flujo de ABR define de hecho un comportamiento de referencia, respecto del cual la red puede determinar si un usuario es cooperativo o no. Falta, no obstante, incorporar en ABR algún mecanismo que garantice que el uso de los recursos por parte de cualquier usuario del servicio ABR no va a exceder la asignación que le corresponde según la realimentación que le entrega la red, puesto que el control de flujo no puede garantizar la asignación por sí mismo.

En ABR, la red debe distribuir los recursos disponibles entre los usuarios de acuerdo con un criterio de equidad determinado. El criterio *max-min* es el criterio más ampliamente utilizado en el diseño de algoritmos de soporte de servicio ABR. En esencia, el criterio *max-min* consiste en asignar a cada una de las conexiones estranguladas en un enlace de cuello de botella una fracción idéntica. Este criterio asume en su formulación que todas las aplicaciones usuarias tienen el mismo derecho a la obtención de recursos. Esta suposición es válida en el caso de las aplicaciones transaccionales de datos que se utilizan sobre la Internet, tales como la transferencia de ficheros o el correo electrónico, para las cuales la calidad de servicio depende del tiempo de tranferencia global de la unidad de información. Sin embargo, existen diversas aplicaciones con requisitos de tiempo real, cuyos requisitos de ancho de banda no son idénticos; por ejemplo, una aplicación de videoconferencia precisa mayor ancho de banda que una aplicación de voz, o que una aplicación transaccional de datos. En tales casos, el criterio *max-min* no es apropiado para el reparto de los recursos. Podrían ser más apropiados otros criterios que tengan en cuenta la prioridad de cada aplicación en la distribución de ancho de banda. Por otro lado, algunas aplicaciones necesitarían que la red garantizase cierto ancho de banda mínimo, el cual les permitiría mantener una interacción mínima imprescindible para su funcionamiento en caso de congestión de la red. Aunque la especificación del servicio ABR contempla que el usuario especifique una valor de tasa mínima que la red debe garantizar, los algoritmos de conmutador existentes no permiten soportar esta posibilidad.

1.2 Objetivos y contribuciones

En esta Tesis, se parte del análisis del concepto de servicio *best-effort* en redes ATM, y se analiza en qué grado cada una de las clases de servicio normalizadas por los organismos ATM Forum y UIT-T, se adecua a esta definición. Tales clases son, aparte de las ya mencionadas UBR y ABR, las clases *non-real-time variable bit rate*, *ATM block transfer* y *guaranteed frame rate*. A continuación se realiza un estudio sobre los mecanismos de soporte del servicio ABR, a saber: la planificación de la transmisión de las células a través de los enlaces, la gestión del espacio de almacenamiento en los conmutadores, la generación y envío de la realimentación por parte de la red y el ajuste de tasa por parte de la fuente en función de esta realimentación. En particular, se hace énfasis en aquellos mecanismos que no están sujetos a normalización, esto es, el algoritmo de planificación, el de gestión del espacio de almacenamiento y el de generación de la realimentación. En cuanto a aquellos mecanismos normalizados por el ATM Forum, esto es, el envío de la realimentación y el

ajuste de tasa en la fuente, son analizados en detalle para justificar cada una de las decisiones tomadas durante el proceso de normalización, que finalizó con la aprobación de *ATM Forum Traffic Management 4.0* (The ATM Forum Committee, 1996) en abril de 1996.

El control de flujo en ABR se basa en la generación por parte de los conmutadores de una señal de realimentación que indica a las fuentes de las conexiones el valor de tasa de emisión al que deben ajustarse. Se han propuesto diversos algoritmos para la generación de la señal de realimentación, también denominados algoritmos de conmutador, tales como ERICA (Jain y otros, 1995) o EPRCA (Roberts, 1994b). Estos algoritmos tratan de estimar localmente cuál es la fracción de ancho de banda que les corresponde a cada una de las conexiones que atraviesan el conmutador a partir de medidas de comportamiento de las conexiones. El criterio de referencia más usado para esta estimación es el criterio *max-min*. Antes de generar el valor de tasa que realimentarán a la fuente, estos algoritmos tienen en cuenta el grado de congestión que experimenta el conmutador. Además, estos algoritmos asumen que la planificación de las células en el conmutador es del tipo FCFS.

La contribución principal de esta Tesis es un algoritmo de conmutador para el control de flujo en ABR. El algoritmo presenta la novedad siguiente: asume que la planificación de las células en los puertos es del tipo equitativo, concretamente según las técnicas *Weighted Fair Queueing* (WFQ) (Demers y otros, 1989) o *Self-Clocked Fair Queueing* (SCFQ) (Golestani, 1994). Básicamente, el algoritmo estima el valor de tasa que realimentar como la fracción de ancho de banda que asigna el algoritmo de planificación a aquellas conexiones estranguladas en su cuello de botella. El grado de congestión en el puerto se controla fijando un valor objetivo de utilización del enlace menor que la unidad. Esta contribución es original: no se han diseñado, hasta la fecha, algoritmos de conmutador de soporte ABR basados en algoritmos de planificación equitativa.

Además de servir de soporte al algoritmo de conmutador, al utilizar algoritmos de planificación equitativa en los nodos se consigue satisfacer las dos deficiencias observadas en el soporte de la clase de servicio ABR y planteadas en la sección 1.1. En primer lugar, los algoritmos de planificación equitativa consiguen proteger la asignación de ancho de banda de cada conexión. De este modo, permiten que el servicio ABR pueda garantizar de forma efectiva que los recursos estén disponibles según indica la realimentación que entrega al usuario, independientemente del comportamiento del resto de los usuarios del servicio. En segundo lugar, los algoritmos de planificación equitativa pueden asignar el ancho de banda según un criterio más general que el criterio *max-min*, a saber, el criterio *max-min* ponderado. Según este criterio, a cada conexión se le asigna una fracción de ancho de banda que, una vez normalizado al peso de la conexión, es igual al del resto de conexiones que están estranguladas en el mismo cuello de botella. Este criterio *max-min* ponderado permitiría que el conjunto de aplicaciones que pueden soportarse sobre ABR se amplie a las aplicaciones de vídeo y de audio, dado que el criterio *max-min* ponderado concede la prioridad adecuada a cada conexión durante la asignación de ancho de banda. Además, mediante la modificación dinámica de los pesos de cada conexión, se puede conseguir que el algoritmo de planificación equitativa garantice un ancho de banda mínimo a cada conexión.

La evaluación del algoritmo propuesto se efectúa mediante simulación por eventos

discretos. Se evalúan, para diversas configuraciones de red ATM, los parámetros de mérito en régimen estacionario. Estos son: la eficiencia de uso de los recursos de red, el grado de equidad en la asignación de tasas y el grado de invariabilidad de prestaciones respecto al tamaño de la red. Asimismo, se evalúan las prestaciones en régimen transitorio, esto es, la estabilidad frente a perturbaciones y la resistencia frente a usuarios no cooperativos.

1.3 Antecedentes

Esta Tesis doctoral se ubica dentro de la línea de investigación de redes de banda ancha existente en el Área de Ingeniería Telemática de la Unversidad Politécnica de Valencia.

El embrión del grupo de trabajo se gestó a raíz de la participación del Área de Ingeniería Telemática en el "Plan de Acción Nacional para la I+D en Comunicaciones Integradas de Banda Ancha" (PlanBA) entre 1994 y 1996. El grupo participó en el proyecto UNICORN ("UNidad de InterCOnexión para Redes IEEE 802.3 e IEEE 802.6 con Nodo ATM"), cuyo objetivo era la especificación, diseño e implementación de una unidad de interfuncionamiento (*internetworking unit*) que permitiera la interconexión de redes LAN IEEE 802.3 y MAN IEEE 802.6 remotas a través de la RDSI-BA basada en ATM.

En años recientes se han leído dos Tesis doctorales sobre la evaluación de prestaciones de mecanismos de capa ATM, en el Área de Ingeniería Telemática, con los títulos "Provisión de servicios de datos sin conexión en la RDSI-BA" (Martínez, 1997) y "Evaluación de prestaciones mediante técnicas de descripción formal de la emulación de red local sobre ATM" (Vidal, 1997).

1.4 Descripción de los contenidos

En el capítulo 2, se analiza el modelo de servicio de capa ATM. Se estudian las definiciones de cada una de las clases de servicio normalizadas por el ATM Forum y por el UIT-T. Se centra el estudio en aquellas clases de servicio que proveen un servicio *best-effort*, definiendo las características de cada clase de servicio, así como los mecanismos que se han diseñado para el soporte de cada una de ellas. Como resultado de este análisis, se establece una definición de qué debería ser un servicio *best-effort* que se proveyese sobre una red ATM y se determina en qué grado cumple cada una de las clases de servicio anteriores esta caracterización.

En el capítulo 3, se estudia la función de los algoritmos de planificación en el soporte de un servicio *best-effort*, concretamente, el papel que desempeñan cuando el servicio se soporta sobre un esquema de control de flujo por realimentación. Se estudian las propiedades de los algoritmos de planificación equitativa, así como su implementación en ATM. Asimismo, se analizan la función y las alternativas en la gestión del espacio de almacenamiento, como parte necesaria del soporte de un servicio *best-effort*.

En el capítulo 4, se centra el análisis en el control de flujo, que es uno de los mecanismos de control de la congestión en servicios *best-effort*. Para cada uno de los mecanismos involucrados en el control de flujo se analizan las alternativas de diseño existentes y se

describen razonadamente cada una de las opciones tomadas en la clase de servicio ABR, definida en *ATM Forum Traffic Management 4.0*. Asimismo, se estudian algunos de los algoritmos de conmutador más conocidos.

En el capítulo 5, se describe el algoritmo de conmutador que se propone en esta Tesis. El algoritmo de conmutador consta de dos mecanismos: un mecanismo que estima la tasa equitativa a partir de la planificación que efectúa un algoritmo tal como WFQ o SCFQ y un mecanismo complementario que controla el grado de congestión en el nodo. Se presentan asimismo los antecedentes conocidos del algoritmo de conmutador presentado. A continuación, se estudian las propiedades del algoritmo de conmutador y la complejidad de implementación del mismo. Finalmente, se describen las posibilidades del algoritmo de conmutador para soportar una distribución de ancho de banda *max-min* ponderada y para garantizar un valor de ancho de banda mínimo.

En el capítulo 6, se evaluan las prestaciones del algoritmo de conmutador. Se justifica la elección de un modelo de simulación, se describe las características del modelo elaborado para la simulación y finalmente se presentan los resultados de las distintas pruebas llevadas a cabo.

Finalmente, en el capítulo 7, se extraen las conclusiones de la Tesis y se apuntan algunas líneas de trabajo futuro.

1.5 Publicaciones

Hasta la fecha, esta Tesis ha dado como fruto dos publicaciones en congresos internacionales, a saber:

GUIJARRO, L., J. R. VIDAL y J. MARTÍNEZ, «A new switch algorithm for generalised Available Bit Rate (ABR) service provision for multimedia applications », *Proceedings of the European Conference on Networks & Optical Communications (NOC'98)* (Manchester) (junio 1998), aceptado para su presentación.

GUIJARRO, L., «Influencia de las disciplinas de servicio en conmutadores ATM sobrel el control de tráfico en servicio ABR », *Proceedings of the 4th International Youth Forum in Computer Science and Engineering (YUFORIC'97)* (Barcelona) (abril 1997), 25–32.

A partir de la experiencia extraída de esta Tesis y de las Tesis citadas en 1.3, se ha generado una tercera publicación en congreso internacional:

GUIJARRO, L., V. PLA, J. R. VIDAL y J. MARTÍNEZ, «Multi-layer simulation approach for evaluation of data service support in ATM networks », *Proceedings of the 1st International Conference on ATM (ICATM'98)* (Colmar) (junio 1998), IEEE, aceptado para su presentación.

Capítulo 2

Servicios *best-effort* en redes ATM

Las redes ATM han sido concebidas desde el principio para proveer integradamente una diversidad de servicios a un gran número de aplicaciones, tanto existentes como por venir. Para abordar la complejidad de tal tarea, los organismos normalizadores en telecomunicaciones, en particular el ATM Forum y el Sector de Normalización de la Unión Internacional de las Telecomunicaciones (UIT-T), han optado por definir un modelo de servicio para la capa ATM que agrupa las aplicaciones por similitud en sus requisitos de calidad de servicio, al tiempo que determina qué mecanismos son los apropiados para proveer qué parámetros de calidad de servicio. El resultado del modelo propuesto por el ATM Forum ha sido una arquitectura de cinco categorías de servicio, mientras que el UIT-T propone cinco capacidades de transferencia.

En este capítulo, se elabora una definición precisa y razonada de servicio de tipo *best-effort*. A partir de esta definición, se estudia el grado en el que cada una de las clases de servicio definidas por ATM Forum/UIT-T se aproxima a ella. Finalmente, se extraen conclusiones al respecto de cuál es el paradigma de servicio *best-effort* sobre redes ATM y cuáles son los mecanismos de soporte necesarios para su provisión.

2.1 La integración de servicios en redes ATM

Es ampliamente aceptado que las redes integradas de banda ancha van a soportarse sobre la técnica de multiplexación y de conmutación conocida como Modo de Transferencia Asíncrono (*Asynchronous Transfer Mode*, ATM), según estableció el (UIT-T, 1993b) en la Recomendación I.150.

La técnica ATM se caracteriza por:

1. operar en modo orientado a conexión;

2. segmentar toda la información que se entrega a la red para su transmisión, en paquetes de longitud fija e igual a 48 bytes de carga útil más 5 bytes de cabecera, denominados células;

3. que la funcionalidad de la cabecera se ha reducido fundamentalmente a la identificación de la conexión virtual;

4. no aplicar control de errores ni de flujo a nivel de enlace, esto es, de conmutador a conmutador, o bien de terminal a conmutador;

5. que la transmisión y la conmutación siguen un esquema de multiplexación asíncrona, mucho más flexible que otros como el esquema de multiplexación síncrona empleado en la RDSI-BE.

Gracias a las características 1, 2, 3 y 4, en las redes ATM la conmutación se realiza eficientemente en *hardware*, lo cual permite obtener grandes capacidades de conmutación[1].

Las redes ATM permiten dar servicio a usuarios con diversos requisitos de servicio. Para ello, cada usuario indica explícitamente la calidad de servicio que precisa durante el establecimiento de la conexión; la red, por su parte, gestiona sus recursos mediante las funciones de control de admisión, control de policía y planificación de transmisión. De este modo, ATM ha conseguido la integración de servicios sobre una misma plataforma de red, a diferencia de lo que ha venido ocurriendo hasta la fecha. El tráfico de voz ha sido transportado tradicionalmente sobre redes telefónicas, el tráfico de vídeo sobre redes de radiodifusión de televisión y el tráfico de datos en redes de comunicación de datos. Las tres redes estaban diferenciadas pues transportaban tipos diferentes de tráfico con requisitos de calidad de servicio también diferentes. De hecho, la unificación de servicios ha sido una motivación importante para el diseño de las redes ATM, pues tal unificación produce, por un lado, economías de escala y, por otro, la aparición de nuevos servicios que no eran en principio factibles o económicamente viables.

Podemos afirmar que las redes de banda ancha soportarán tráfico generado por dos tipos principales de aplicaciones (Keshav, 1997). Por un lado, aquellas aplicaciones que sí precisan de unos márgenes mínimos de calidad. Por ejemplo, una aplicación que genera voz en forma de flujo continuo de 64 kbit/s dejaría de ser operativa si la red proporcionase menos de 64 kbit/s en algún punto del trayecto de un extremo a otro. Además, si la aplicación es bidireccional e interactiva, desde el punto de vista de la ergonomía se requiere que el retardo de ida y vuelta sea menor que unos 150 ms. Por tanto, si la red quisiera soportar una aplicación bidireccional de voz con unos mínimos de calidad, debería garantizar, además de un ancho de banda de 64 kbps, un retardo de ida y vuelta de unos 150 ms. Este tipo de aplicaciones de denominan de servicio garantizado y requieren que se les reserve recursos en la red.

Por otro lado, se encuentran aquellas aplicaciones comunes actualmente en la Internet. Estas son relativamente insensibles a las prestaciones que le ofrece la red, pues aceptan cualquier nivel de calidad de servicio que les proporcione la red. En otras palabras, los requisitos de calidad de servicio de tales aplicaciones son elásticos: la aplicación se adapta a los recursos disponibles en cada momento. Por ejemplo, una aplicación de transferencia

[1]Como ejemplos de productos comerciales, *Fore Systems* ofrece el conmutador *Forerunner ASX-1000*, con una capacidad de 10 Gbit/s, mientras que *Lucent Technologies*, el conmutador *Globeview 2000*, con una capacidad de 20 Gbit/s, segun la revista Data Communications (http://www.data.com)

de ficheros preferiría un ancho de banda infinito y un retardo nulo extremo a extremo; no obstante, funcionaría correctamente, aunque no de forma óptima, aunque el ancho de banda disponible disminuyese y el retardo extremo a extremo aumentase. Tales aplicaciones reciben el nombre de aplicaciones *best-effort* , pues la red se compromete con el usuario en el intento de entregar los paquetes, pero sin establecer unos márgenes mínimos de calidad. Nótese que, en principio, el servicio *best-effort*, que es el que se proporciona a aplicaciones *best-effort*, no requiere que la red reserve recursos durante el establecimiento de la conexión.

Más detalladamente, presentamos en la tabla 2.1 una tabla que muestra una clasificación de aplicaciones elaborada desde el punto de vista del usuario. Se ha tomado de la recomendación UIT-T I.211 (UIT-T, 1993a):

Se ha establecido una diferenciación en función del medio —video, voz, imagen y datos— y del modo de acceso —conversacional o interactivo, de mensajería, de consulta , de distribución—:

Conversacional Los servicios conversacionales proporcionan en general los medios para una comunicación bidireccional con transferencia de información en tiempo real (sin almacenamiento ni retransmisión) de extremo a extremo, entre usuarios o entre un usuario y un ordenador principal (por ejemplo, para tratamiento de datos). El flujo de la información de usuario puede ser bidireccional simétrico, bidireccional asimétrico y, en ciertos casos concretos (por ejemplo, en la vigilancia por vídeo), unidireccional. La información es producida por el usuario o usuarios emisores y se dirige a uno o más copartícipes de la comunicación situados en el lado receptor. Son ejemplos de servicios conversacionales de banda ancha la videotelefonía, la videoconferencia y la transmisión de datos a alta velocidad.

De mensajería Los servicios de mensajería ofrecen la comunicación de usuario a usuario entre usuarios individuales por medio de unidades de almacenamiento con funciones de almacenamiento y retransmisión, de buzón electrónico y/o tratamiento de mensajes (por ejemplo, edición, tratamiento y conversión de información). Son ejemplos de servicios de mensajería de banda ancha los servicios de tratamiento de mensajes y los servicios de correo electrónico para imágenes en movimiento (películas), imágenes de alta resolución e información audio.

De consulta El usuario de los servicios de consulta puede consultar la información almacenada en centros de información para uso público. Esta información se enviará al usuario solamente si la solicita. La información puede consultarse individualmente. Además, el usuario controla el instante en que debe comenzar una secuencia de información. Como ejemplos pueden mencionarse los servicios de consulta de banda ancha para películas, imágenes de alta resolución, información audio e información de archivos.

De distribución Estos servicios abarcan los servicios de difusión. Proporcionan un flujo continuo de información que es distribuido desde una fuente central a un número ilimita-

Clase de Aplicación	*Ejemplo*
Vídeo interactivo	Videoconferencia
Voz interactiva	Telefonía
Texto interactivo	Transacción bancaria
Imagen interactiva	Conferencia multimedia
Mensajería de vídeo	Correo multimedia
Mensajería de voz	Buzón de voz
Mensajería de texto/datos	E-mail, telex, fax
Mensajería de imagen	
Distribución de vídeo	Televisión, Teleenseñanza
Distribución de voz	Radio
Distribución de texto	Tablón de anuncios
Distribución de imagen	Previsión meterorológica
Consulta de vídeo	Vídeo bajo demanda
Consulta de voz	Audioteca
Consulta de texto/datos	Transferencia de ficheros
Consulta de imagen	Catálogo de biblioteca
LAN agregadas	Interconexión de LANs
Terminal remoto	telnet
Llamadas remotas a procedimiento	
Servicio distribuido de ficheros	
Computer process swap	

Tabla 2.1. CLASES Y EJEMPLOS DE APLICACIONES

do de receptores autorizados conectados a la red. El usuario puede acceder a este flujo de información, sin la posibilidad de determinar en qué instante debe comenzar la difusión de la cadena de información. El usuario no puede controlar el comienzo ni el orden de presentación de la información difundida. Dependiendo del momento en el que se produce el acceso del usuario, puede que la información no sea presentada desde el comienzo. Son ejemplos de estos servicios los servicios de radiodifusión de programas de televisión y de audio.

Finalmente la clase de aplicaciones de **comunicación entre ordenadores** comprende interacciones entre máquinas. Las tres últimas clases de la tabla 2.1 son de comunicación entre ordenadores.

Podemos caracterizar inicialmente las aplicaciones clasificadas anteriormente en términos del tipo de tráfico que generan y de la calidad de servicio que requiren.

En primer lugar, podemos identificar distrintos grados de regularidad en el patrón de tráfico que generan. El audio y el video se han codificado habitualmente utilizando codifi-

cación de tasa constante para su transmisión a través de redes de conmutación de circuitos; incluso si se utiliza codificación de tasa variable para su transmisión a través de redes de conmutación de paquetes la razón tasa de pico a tasa media está limitada entre 3 y 10, lo cual no permite ganancias significativas por multiplexación estadística. Esta razón de tasas de emisión se denomina coeficiente de ráfaga. Por otro lado, aquellas aplicaciones de comunicación entre ordenadores así como aplicaciones de audio/video/imagen que transportan unidades discretas de información, esto es, las aplicaciones de mensajería y de consulta, suelen presentar un patrón de tráfico muy esporádico, lo cual permite obtener ganancias por multiplexación estadística. Además, estas aplicaciones, que denominaremos transaccionales, no llevan asociadas intrínsecamente una tasa natural de generación.

En segundo lugar, la calidad de servicio viene descrita básicamente por la sensibilidad de la aplicación al retardo y por su tolerancia frente a pérdidas. En cuanto a la sensibilidad al retardo, para una aplicación conversacional de vídeo/voz, una célula tardía es una célula perdida. Por su parte, las aplicaciones de distribución pueden tolerar retardos significativos, pero necesitan que la variación en el retardo esté acotada, pues disponen de un *buffer* de tamaño limitado en recepción para re-sincronizar la reproducción. Para una aplicación transaccional, en cambio, una transacción se ha completado cuando todos los datos han llegado; cuanto antes se complete la transacción, mejor calidad se ofrece. Además, el usuario percibe la transferencia como un único objeto; por tanto, en la calidad que el usuario percibe no influyen los detalles de temporización de la transferencia de cada paquete en particular. Finalmente, existe un tiempo mínimo de transferencia debajo del cual el usuario no percibe mejora en la calidad. Dentro de las aplicaciones transaccionales, las aplicaciones de mensajería son poco sensibles al retardo, mientras que en las aplicaciones de consulta el usuario es sensible únicamente a retardos de más de unos pocos segundos.

En cuanto a la tolerancia frente a pérdidas, las aplicaciones de comunicación entre ordenadores y en general todas las transaccionales de datos son generalmente sensibles a las pérdidas porque todos los datos que se transfieren deben recibirse correctamente, por lo que los datos que no se reciben deben recuperarse mediante retransmisión. El coste o la inconveniencia de la retransmisión depende de la sensibilidad de la aplicación al retardo. En cambio, la voz y el video conversacionales pueden ser tolerantes frente a pérdidas si se codifican adecuadamente: la pérdida de información de sincronización de trama puede ocasionar una importante degradación de la calidad, mientras que la pérdida de resolución en una imagen o de una muestra de voz puede pasar inadvertida. Además, la codificación multiresolución o estructurada (*layered*) permite crear flujos de prioridad alta, protegidos de forma robusta frente a pérdidas, y flujos de prioridad baja, cuya pérdida no ocasiona merma en la calidad global

2.1.1 El modelo de servicio de capa ATM

Como se ha mostrado en la sección 2.1, existe un gran número de aplicaciones actuales y futuras que deberán soportarse sobre redes ATM, cada una de las cuales puede caracterizarse en términos de patrón de generación de células y de calidad de servicio —retardo y pérdida—. Asimismo hay una variedad de mecanismos a disposición de los diseñado-

res de conmutadores ATM para soportar el transporte del amplio espectro de aplicaciones mencionado.

Un Modelo de Servicio es una abstracción que permite hacer corresponder las aplicaciones a los mecanismos. Para ello, por un lado agrupa a las aplicaciones por similitud en su comportamiento y en sus requisitos de calidad de servicio; por otro, identifica aquellos mecanismos de la red que pueden proporcionar unos determinados parámetros de calidad de servicio. El resultado esta agrupación y correspondencia a nnivel de la capa ATM es una enumeración de clases de servicio. El ATM Forum, en *ATM Forum Traffic Management 4.0*, especifica cinco Categorías de Servicio; para cada una, define un conjunto de parámetros para describir el tráfico que se presenta a la red, y otro para describir la calidad de servicio correspondiente que se exige de la red. Por su parte, el UIT-T, en I.371 (UIT-T, 1996), especifica cinco Capacidades de Transferencia (*ATM Transfer Capabilities*, ATC), algunas de las cuales son equivalentes a las categorías de servicio del ATM Forum pero reciben denominaciones diferentes.

A continuación, describiremos sucintamente cuáles son los parámetros de calidad de servicio que se emplean en el Modelo de Servicio de la capa ATM y cuáles son los mecanismos básicos a emplear para la provisión de las clases de servicio a nivel de capa ATM.

2.1.1.1 La Calidad de Servicio en capa ATM

Los parámetros de Calidad de Servicio sirven para cuantificar las prestaciones de una conexión de capa ATM. A continuación, describiremos los tres parámetros de calidad de servicio que pueden ser negociados entre el usuario y la red en la fase de establecimiento de la conexión correspondiente. Tales parámetros son de naturaleza estadística

El retardo de transferencia de célula (*Cell Transfer Delay*, CTD) se define como el tiempo transcurrido entre la transmisión de una célula en el punto de medida 1 (p.ej., en el interfaz UNI del terminal de origen) y su correspondiente recepción en el punto de medida 2 (p.ej., en el interfaz UNI del terminal de destino) para una conexión determinada. El CTD incluye componentes de retardo fijos, debidos a propagación en el medio, transmisión y procesado en los conmutadores, y componentes de retardo variable, debidos a tiempo de espera en las colas de los conmutadores. El parámetro de calidad de servicio CTD Máximo se define como el cuantil (1-α) del CTD.

El parámetro Variación de pico de Retardo de Célula (*Cell Delay Variation*, CDV) se define como el cuantil (1-α) del CTD menos el CTD fijo que puede experimentar cualquier célula transmitida a través de una conexión establecida.

Por último, el parámetro Tasa de Pérdida de Células (*Cell Loss Ratio*, CLR) se define, para cada conexión, como el cociente entre células perdidas y el total de células transmitidas a través de la conexión establecida.

El compromiso que adopta la red en cuanto a los parámetros de calidad de servicio es el siguiente: la red se compromete a satisfacer los parámetros de calidad de servicio negociados siempre que el usuario cumpla su parte del Contrato de Tráfico. El Contrato de Tráfico es una especificación de las características susceptibles de ser negociadas durante el establecimiento de una conexión. Consiste en un descriptor de tráfico de la conexión,

un conjunto de parámetros de calidad de servicio para cada sentido de la conexión; y una definición de conexión conforme, esto es, bajo qué condiciones concretas la red se compromete a garantizar la calidad de servicio negociada para la conexión. El descriptor de tráfico de la conexión comprende, a su vez, los siguientes elementos:

1. El descriptor de tráfico de la fuente, es decir, los parámetros de tráfico de la fuente ATM, i.e., PCR, SCR, MBS o MCR, que describiremos enla sección 2.1.2.

2. Su Tolerancia de CDV (*Cell Delay Variation Tolerance*, CDVT), por el que se fija un valor máximo a la variabilidad de retardo que la fuente puede ocasionar, debido a multiplexado de células a nivel de capa ATM o capa física; esta variabilidad altera las características del tráfico de la conexión declaradas en el descriptor de tráfico de la fuente.

3. Una definición de conformidad, es decir, un algoritmo que permita a la red decidir si las características del tráfico de la conexión respetan el descriptor declarado de tráfico de la fuente; en *ATM Forum Traffic Management 4.0*, el ATM Forum propone el *Generic Cell Rate Algorithm* (GCRA).

2.1.1.2 Procedimientos de control de tráfico en capa ATM

En *ATM Forum Traffic Management 4.0* especifica una serie de funciones y procedimientos para el control de tráfico y de la congestión a nivel de capa ATM. Además, indica que las redes ATM pueden implementar cualquier combinación de tales funciones con el propósito de satisfacer determinados objetivos de calidad de servicio de aquellas conexiones identificadas como conformes. Las tres funciones más importantes son: el control de flujo especificado para la Categoría de Servicio ABR, que se describirá en el capítulo 4, y las funciones CAC y UPC.

La función de Control de Admisión de Conexiones (*Connection Admission Control*, CAC) se define como el conjunto de acciones que toma la red cuando se establece una conexión con el fin de determinar si la conexión puede ser establecida o debe ser rechazada. En particular, a partir del contrato de tráfico de la conexión, la función CAC debe determinar si existen suficientes recursos en cada uno de los elementos de la red para establecer tal conexión; además debe asegurarse de que la calidad de servicio negociada para las conexiones ya establecidas puede mantenerse. Finalmente, su implementación está sujeta a diferenciación por parte del fabricante y del operador.

La función de Control de Parámetros de Usuario (*Usage Parameter Control*, UPC) se define como el conjunto de acciones que toma la red para monitorizar y controlar el tráfico de cada una de las conexiones establecidas. En particular, debe detectar las posibles violaciones del contrato de tráfico por parte de cada conexion, para lo cual debe verificar que el tráfico de la conexión respeta los valores especificados por su descriptor de tráfico de fuente, según la definición de conformidad dada y teniendo en cuenta el CDVT negociado. Las células para las que UPC verifica que no son conformes deben descartarse, aunque se contempla la opción de marcar tales células con baja prioridad para el descarte.

2.1.2 El modelo de Categorías de Servicio del ATM Forum

La definición de Categorías de Servicio por parte del ATM Forum ha seguido un proceso de crecimiento por refinamiento progresivo. Efectivamente, cuando se detecta que una aplicación tiene un comportamiento y/o unos requisitos de Calidad de Servicio que no pueden ser satisfechos eficiente o eficazmente por una Categoría existente, se plantea la definición de una nueva Categoría.

Según *ATM Forum Traffic Management 4.0* la arquitectura de servicios que se proveen a nivel de la capa ATM consta de las siguientes 5 categorías de servicio: categoría de tasa de bit constante (*Constant Bit Rate*, CBR), categoría de tasa de bit variable con requisitos de tiempo real (*real-time Variable Bit Rate*, rt-VBR), categoría de tasa de bit variable sin requisitos de tiempo real (*non-real-time Variable Bit Rate*, nrt-VBR), categoría de tasa de bit disponible (*Available Bit Rate*, ABR) y categoría de tasa de bit sin especificar (*Unspecified Bit Rate*, UBR).

2.1.2.1 Categoría CBR

Está destinada a aplicaciones de tiempo real, que tienen unas restricciones de retardo y que no tienen un patrón muy esporádico (*bursty*) de tráfico; en otras palabras, el coeficiente de ráfaga suele ser bajo.

El descriptor de tráfico de fuente contiene el parámetro de tráfico Tasa de Pico (*Peak Cell Rate*, PCR), tasa que la aplicación se compromete a no superar. Por otro lado, en CBR se especifica una calidad de servicio en términos de tasa CLR aceptable, de un CTD máximo y de un CDV de pico.

En CBR, la red reserva con exclusividad los recursos, de modo que un ancho de banda constante es puesto a disposición de la conexión durante su duración

2.1.2.2 Categoría UBR

Esta Categoría de Servicio está destinada a aplicaciones transaccionales, que suelen generar tráfico muy esporádico y que además toleran retardos no acotados.

El descriptor de tráfico de fuente contiene el parámetro PCR, que opcionalmente puede ser utilizado por la red para sus funciones CAC y UPC. No especifica tasa de pérdidas aceptable, ni un retardo máximo ni un CDV máximo; por tanto, la red no se compromete a garantizar ningún tipo de calidad de servicio.

La red puede dimensionar globalmente los recursos, con el objeto de acomodar el tráfico generado por el conjunto de las conexiones UBR

2.1.2.3 Categoría rt-VBR

Está destinada a soportar aplicaciones de tiempo real con codificación de tasa variable de forma más eficiente que CBR.

El descriptor de tráfico de fuente consiste en una tasa máxima PCR que la aplicación se compromete a no superar; además, una tasa media, que se denomina *Sustainable Cell Rate*

(SCR) y un tamaño máximo de ráfaga, denominado *Maximum Burst Size* (MBS). En cuanto a calidad de servicio, especifica una tasa de pérdidas aceptable, un retardo máximo y un CDV de pico.

La red asigna los recursos necesarios en el establecimiento de la conexión, pero aquellos no utilizados en cada momento pueden ser empleados por servicios con prioridad menor, tales como UBR o ABR.

2.1.2.4 Categoría nrt-VBR

Está destinada a mejorar la calidad del servicio que perciben las aplicaciones transaccionales, frente a UBR.

Se define un descriptor de tráfico de fuente con una tasa máxima PCR, una tasa media SCR y un tamaño máximo de ráfaga MBS. Sí especifica tasa de pérdidas aceptable, pero no un retardo máximo ni un CDV de pico.

La red reserva los recursos para garantizar las prestaciones y permitir multiplexado estadístico

2.1.2.5 Categoría ABR

Al igual que nrt-VBR, ABR está destinada a mejorar la calidad de servicio que perciben las aplicaciones transaccionales, frente a UBR.

Sin embargo, en ABR se define un comportamiento de referencia que permite modular la tasa de emisión de células según un mecanismo de control de flujo por ajuste de tasa. Se establecen garantías relativas y procedimentales sobre pérdidas, pero no se especifica un retardo máximo ni un CDV de pico.

2.1.3 Las Capacidades de Transferencia del UIT-T

En general, es posible una correspondencia entre las Categorías de Servicio del ATM Forum y las Capacidades de Transferencia del UIT-T:

- la categoría de servicio CBR se denomina *Deterministic Bit Rate* (DBR);
- la categoría de servicio UBR no tiene Capacidad de Transferencia equivalente;
- la categoría de servicio VBR se denomina *Statistical Bit Rate* (SBR), aunque sólo se especifica *non-real-time* SBR;
- la categoría de servicio ABR se denomina de manera idéntica;
- se especifica una nueva Capacidad de Transferencia, denominada *ATM Block Transfer*(ABT).

En la Capacidad de Transferencia ABT, las características de transferencia de capa ATM para una conexión se negocian a nivel de bloque ATM, donde se define bloque como un grupo de células de una conexión. Para cada bloque se especifica un valor de tasa máxima

de generación, denominada *Block Cell Rate* (BCR). Una vez aceptado el bloque por parte de la red, la calidad de servicio que recibe el bloque ATM es equivalente al de una conexión DBR cuya tasa PCR fuese igual a BCR.

Durante el establecimiento de la conexión, se especifica un valor de tasa máximo PCR, así como un valor de tasa media SCR y un valor MBS; finalmente se negocia un valor máximo de frecuencia para las transacciones de renegociación de BCR. Se especifica como parámetros de calidad de servicio, CLR, CTD máximo y CDV de pico.

2.2 Los servicios *best-effort* en redes ATM

El concepto de servicio *best-effort* se empleó para caracterizar el servicio que perciben los usuarios de la Internet a nivel IP. En un servicio *best-effort*, la red proporcionaba una calidad de servicio que era la *mejor posible* en cada momento; la red *hace todo lo que puede*. Cuando en 1994, el ATM Forum inició la tarea de la definición del servicio ABR sobre red ATM, una de las premisas fue que el servicio fuese *best-effort*. El concepto de servicio *best-effort* no ha sido definido con precisión en la literatura, a pesar de que se ha asumido tradicionalmente en el estudio del funcionamiento de las redes de computadores. En esta sección nos proponemos aclarar este concepto; para ello, se delimitará qué se ha entendido hasta el momento por servicio *best-effort* y se diagnosticará qué cambios conceptuales supone la provisión de este tipo de servicio sobre redes ATM.

En esta Tesis, proponemos la siguiente definición de servicio *best-effort*:

> El servicio que ofrece una red a sus usuarios lo denominamos *best-effort* cuando la calidad del servicio ofrecido por la red —y percibida por los usuarios— depende del estado de la red y éste es impredecible.

Entendemos por estado de la red el estado de ocupación de los recursos de la red en ese instante, donde por recursos se entiende el ancho de banda en los enlaces y el espacio de almacenamiento en los nodos. Afirmamos que el estado de una red con servicio *best-effort* es impredecible, pues los recursos de la red se asignan a los usuarios en el momento en que cada uno de ellos los solicita; así, en el instante en el que un paquete llega a un nodo se le asigna espacio de almacenamiento y se decide el orden en que será transmitido —por tanto, se le asigna ancho de banda—.

De acuerdo con esta definición, el servicio que proporciona el protocolo IP en una *internet* es un servicio *best-effort*, por las dos razones siguientes. En primer lugar, una *internet* IP no reserva recursos, con el objetivo de evitar niveles bajos de utilización, pues el tráfico de datos tiene un comportamiento impredecible y esporádico. Al mismo tiempo, una *internet* IP efectúa conmutación de paquetes y así consigue niveles altos de utilización al multiplexar estadísticamente los recursos de la red. En segundo lugar, la calidad de servicio en una *internet* IP —en términos de retardos de transferencia y de descarte por sobrecarga/congestión— depende no sólo de las acciones que lleva a cabo la red —y que, por tanto, puede controlar— sino también de la carga ofrecida —que la red no puede controlar—. Efectivamente, cada nodo de una *internet* intenta reenviar los paquetes tan

pronto como puede, pero la *internet* no puede comprometerse a satisfacer unos márgenes mínimos de calidad de servicio.

Podemos identificar, no obstante, algunos rasgos característicos del servicio *best-effort* IP. Si bien el servicio IP ofrece una calidad de servicio impredecible, ésta puede ser percibida. Efectivamente, el usuario del servicio IP puede conocer la calidad de servicio que recibe en términos de retardo de los segmentos y/o de pérdida de los mismos por sobrecarga/congestión de la red. Además, el servicio IP puede considerarse como servicio *best-effort* no orientado a evitar pérdidas. Efectivamente, una *internet* IP no proporciona ninguna comunicación explícita sobre el estado de la red al usuario del servicio IP, que pueda ser utilizada por éste para prever un eventual agotamiento de los recursos.

Como consecuencia de la ausencia de comunicación explícita alguna por parte de la red, el usuario del servicio IP, que es el protocolo TCP, se ve obligado a implementar un mecanismo de averiguación del estado de la red. Tal mecanismo debe basarse en el único parámetro de calidad de servicio que puede percibir, esto es, el retardo y/o pérdida que sufren los segmentos que TCP entrega al servicio de red IP. El ejemplo más conocido de mecanismo de averiguación del estado de una *internet* IP es el especificado para TCP (Postel, 1981), mejorado por Jacobson (1988). Además, la misma naturaleza del parámetro escogido para la averiguación del estado de la red, esto es, el retardo de ida y vuelta, provocan que los cambios en el estado de la red se detecten sólo después de varios tiempos de ida y vuelta y que, además, tengan únicamente un carácter de estimación estadística.

El servicio IP fue diseñado inicialmente bajo la suposición de que los usuarios del servicio cooperarían en el empeño de evitar la congestión de la red (Lefelhocz, Lyles, Shenker y Zhang, 1996). Esta suposición era factible por las siguientes razones: en primer lugar, la comunidad de usuarios de Internet era relativamente reducida y admitía una autorregulación informal en el uso del servicio. Además, existía un único algoritmo de control de congestión homogéneamente implantado en el conjunto de usuarios de la red: la versión 4.3BSD de TCP. En tercer lugar, existía un único tipo de aplicaciones usuarias en la red, las de tipo transaccional, tales como la transferencia de ficheros o el correo electrónico. Finalmente, Internet se inició como una red académica, esto es, corporativa; por tanto, el administrador de la red ejercía control sobre los usuarios.

A la hora de diseñar el soporte de un servicio *best-effort* sobre red ATM, aparece una serie de aspectos nuevos con respecto al escenario en que se proveía el servicio *best-effort* en Internet.

En primer lugar, las redes ATM son homogéneas, esto es, todos los nodos emplean la misma tecnología, en contraste con las redes IP que son *internets*, esto es, redes heterogéneas. Esta característica trae dos consecuencias importantes en el diseño del soporte de un servicio *best-effort* sobre red ATM. Por un lado, todos los elementos de la red están capacitados —o pueden capacitarse— para ejercer un control sobre el flujo de células de una conexión. En las *internets*, en cambio, los elementos de las distintas subredes no pueden actuar sobre los datagramas IP. Por otro lado, el retardo de transferencia de las células ATM a través de una conexión es menos variable que el de los datagramas IP a través de una *internet*. Estos dos aspectos permiten, en primer lugar, que se puedan diseñar meca-

nismos de realimentación explícitos mediante los cuales la red informe del estado de la red directamente al usuario, es decir, del estado de ocupación de los recursos; en segundo lugar, que la notificación del estado de la red sea puntual; por tanto, la escala temporal de respuesta a la congestión podrá ser menor que en las *internets* IP.

En segundo lugar, las redes ATM, desde el mismo instante de su concepción, han sido diseñadas como redes públicas de telecomunicación y, por tanto, con el fin de proveer servicios comerciales. Por tanto, las suposiciones iniciales de cooperación voluntaria de los usuarios de Internet ya no son factibles porque (Lefelhocz y otros, 1996), primero, no es suficiente apelar al interés público para conseguir la cooperación de los usuarios; segundo, la gran variedad de sistemas operativos y de plataformas *hardware* hace imposible mantener un algoritmo de control de congestión homogéneamente implantado en toda la red; y tercero, las aplicaciones capaces de operar sobre un servicio *best-effort* son cada vez más variadas, de modo que la calidad de servicio que consiguen del servicio IP estas nuevas aplicaciones[2] a través del algoritmo de control de congestión de TCP —que es satisfactorio para las aplicaciones de tipo transaccional—, puede no satisfacer sus requisitos.

2.3 Alternativas al soporte de servicios *best-effort* sobre redes ATM

A partir de la definición de clases de servicio abordada por el ATM Forum y por el UIT-T, que ha sido presentada en la sección 2.1, se pasa a analizar con más detalle aquellas que más se acercan a la definición de servicio *best-effort* que se ha discutido en la sección 2.2, así como los mecanismos estudiados para la provisión de cada una de ellas.

2.3.1 El servicio UBR

El servicio *best-effort* que proporciona UBR está previsto para aplicaciones transaccionales que puedan emplear el ancho de banda disponible y que no sean sensibles a las pérdidas ni a los retardos de transferencia

Las funciones CAC y UPC son opcionales en este tipo de categoria de servicio. En particular, sólo se deniega el establecimiento de conexiones cuando se especifica un valor PCR que no puede soportar la red y sólo podrá monitorizarse la conformidad de las células respecto a la tasa PCR.

La red no se compromete a implementar ningún mecanismo de control de congestión. Así, durante la congestión, la red descartará células en los nodos, pues no se asume que las fuentes deban reducir su tasa de emisión. Además, las aplicaciones usuarias deberán proveer sus propios mecanismos de recuperación frente a pérdidas y de retransmisión, tales como el control de flujo y de retransmisión por ventana deslizante de TCP.

Las ventajas de UBR como servicio *best-effort* son su simplicidad y la mínima interacción requerida entre usuario y red. Algunas desventajas de UBR son la ausencia de garantías de calidad de servicio y la posible distribución no equitativa del ancho de banda.

[2]Gran parte de estas aplicaciones se soportan sobre el protocolo RTCP, definido dentro de RTP (IETF RFC 1889), para el transporte de voz y vídeo de distribución y conversacionales sobre IP

Sin embargo, el principal problema que plantea el soporte del servicio UBR es la degradación de las prestaciones por el efecto combinado de la ausencia de mecanismo de control de congestión y de la segmentación en células. Normalmente los paquetes de las capas superiores han de segmentarse antes de ser transmitidos y conmutados por la red ATM y posteriormente han de ser reensamblados en destino, de modo que la pérdida de una célula provoca la pérdida íntegra del paquete de capa superior al que pertenece. Por un lado, la pérdida de células puede ocasionar un número elevado de paquetes de capa superior pérdidos, pues la tasa de pérdida de células y la de paquetes no guardan una relación lineal, sino que depende en gran medida de las pautas que siga la pérdida de células. Por otro, cuando una célula es descartada en un conmutador, el resto de las células pertenecientes al mismo paquete de capa superior, aun siendo inservibles en recepción, pueden continuar ocupando recursos de la red, mientras progresan hacia el receptor. Se produce entonces una degradación en el rendimiento, que es peor cuanto mayor es el tamaño del paquete de capa superior.

Existen varios mecanismos para la reducir el impacto de la segmentación en UBR (Romanow y Floyd, 1995). Tales mecanismos persiguen evitar que las células de los paquetes que han padecido el descarte de algunas de sus células progresen más alla del conmutador en donde tuvo lugar el descarte.

El primero de ellos se denomina *Partial Packet Discard* (PPD), en virtud del cual, cada vez que se descarta una célula por desbordamiento de la cola, se descartarán todas las células del mismo paquete que lleguen posteriormente al conmutador. De este modo, se evita transmitir inútilmente células que no podrán ser ensambladas en recepción. Nótese que el descarte es parcial porque, cuando se descarta una célula, algunas células del mismo paquete pueden ya haber sido aceptadas en el conmutador o incluso transmitidas. La realización de esta estrategia cuando la capa AAL utilizada es tipo 5 es inmediata, pues la indicación de última célula de paquete está contenida en la cabecera de la célula.

El segundo de ellos recibe el nombre de *Early Packet Discard* (EPD), según el cual, cuando es previsible que un paquete no podrá ser transmitido completamente en un conmutador, se descartarán todas sus células antes de que provoquen desbordamiento de la cola. La previsión de que un paquete pueda desbordar la cola se hace a partir del nivel de llenado de la misma. Con EPD se obtiene un rendimiento superior al obtenido con PPD. No obstante, el ancho de banda no se reparte equitativamente entre las conexiones, pues se discrimina desfavorablemente a aquellas conexiones con tiempo de ida y vuelta mayor, pues atraviesan un mayor número de conmutadores y, por tanto, tienen una mayor probabilidad de ser descartados antes de llegar a su destino: es el fenómeno conocido como *beat-down*. Además, el nivel de ocupación de las colas de los conmutadores es alto, por lo cual las necesidades de memoria de los conmutadores es alta, para un determinado nivel de prestaciones, y los retardos de transferencia son también altos.

Finalmente, el mecanismo denominado *Random Early Detection* (RED) tiene como finalidad acotar el nivel medio de llenado de la cola. Cuando llega la primera célula de un paquete y el nivel de la cola supera cierto umbral, el conmutador descarta con cierta probabilidad la totalidad de las células de ese paquete

2.3.2 El servicio nrt-VBR

El servicio nrt-VBR ofrece un servicio *best-effort* destinado a aplicaciones que no sean tolerantes a pérdidas, por cuanto que establece una garantía de calidad de servicio en términos de tasa CLR aceptable. En contraprestación, obliga al usuario a negociar un contrato de tráfico, cuyo descriptor de tráfico de fuente consta de:

1. una tasa PCR, que fija el intervalo mínimo de tiempo entre dos células consecutivas entregadas a la red; y

2. una tasa SCR y un tamaño máximo de ráfaga MBS, que limita a MBS el número de células que pueden ser entregadas a la red a la tasa PCR.

La red, por su parte, monitoriza la conformidad del tráfico de fuente con los parámetros descritos en el contrato de tráfico.

La red reserva recursos de red. Según Roberts, Bensaou y Canetti (1993), una posible estrategia de asignación de recursos puede ser la siguiente. El ancho de banda que se asigna a cada conexión es un valor ligeramente superior al parámetro SCR declarado, de modo que la función CAC rechazará cualquier nueva conexión si la suma de los anchos de banda asignados en cualquier punto de multiplexado a lo largo del trayecto de la nueva conexión, supera la capacidad del enlace correspondiente. El espacio de almacenamiento se puede asignar según tres alternativas:

1. Se asigna en cada nodo una cantidad de memoria igual al parámetro MBS declarado por el usuario y monitorizado por la red. Esta solución se traduce en un uso ineficiente de los recursos.

2. Se asigna en cada nodo la cantidad de memoria necesaria para garantizar una probabilidad máxima de desbordamiento de las colas bajo la suposición de comportamiento en el peor caso por parte del usuario en presencia de función UPC, esto es, emisión de ráfagas de MBS células a la tasa PCR y espaciadas por intervalos de silencio de MBS/SCR de duración. Esta solución no mejora demasiado la eficiencia respecto al caso anterior.

3. Se asigna en cada nodo la cantidad de memoria necesaria para garantizar una probabilidad máxima de desbordamiento de las colas bajo la suposición de comportamiento "natural" tal que el tráfico de entrada no es modificado por la función UPC; esta solución mejora los requisitos de memoria en un factor de 10 respecto a la asignación individual.

El retardo que experimentarían los usuarios con la estrategia de asignación descrita se puede describir como *best-effort*. Según Roberts y otros (1993), en circunstancias normales la transferencia de las ráfagas de datos a través de la red se realizaría a la tasa PCR, debido al efecto de multiplexado estadístico. En situación de sobrecarga/congestión, la reserva de ancho de banda impediría un retardo indeterminadamente elevado.

2.3.3 El servicio ABR

Desde el punto de vista del usuario, el servicio ABR se caracteriza por los siguientes aspectos (Bonomi y Fendick, 1995) (Chen y otros, 1996). ABR es apropiado *sólo* para aplicaciones que puedan adaptar sus tasas de emisión al ancho de banda disponible y tolerar retardos de transferencia imprevisibles. No obstante, durante el establecimiento de la conexión, el usuario puede especificar dos valores mínimo y máximo para el ancho de banda asignable durante el tiempo de vida de la conexión; tales cotas se denominan *Minimum Cell Rate* (MCR) y *Peak Cell Rate* (PCR), respectivamente. Así, el ancho de banda disponible para cada usuario es variable y puede disminuir hasta el mínimo especificado, en función de la disponibilidad de los recursos de la red. En ABR, la calidad de servicio se formula en términos de tasa de pérdidas CLR aceptable, condicionada a que el usuario adapte su tasa de emisión al ancho de banda disponible. Por último, la distribución de ancho de banda disponible entre los usuarios del servicio ABR debe ser equitativa: ningún usuario o conjunto de usuarios debe ser arbitrariamente discriminado ni favorecido, aunque los recursos deben ser asignados según una política definida.

Desde el punto de vista de la provisión del servicio, ABR debe proporcionar acceso rápido al ancho de banda disponible en la red en cada momento, de forma que sea ocupado de forma eficiente y asignado equitativamente entre las conexiones activas. Además, ABR debe especificar un comportamiento de referencia para el sistema fuente, para el sistema destino y para los conmutadores intermedios de cada conexión, de modo que las garantías de calidad de servicio sean aplicables a aquellos sistemas que cumplan el comportamiento de referencia

A continuación, analizaremos con más detalle el concepto de equidad y su relación con la provisión del servicio ABR.

2.3.3.1 El criterio de equidad en el reparto de ancho de banda

En la definición del servicio ABR en *ATM Forum Traffic Management 4.0*, al respecto del procedimiento de asignación de ancho de banda, se establece únicamente que debe resultar en una distribución que cumpla algún criterio de equidad. A título meramente informativo, en el mismo documento se citan algunos de los posibles criterios de equidad.

El criterio de equidad más comúnmente empleado es el criterio *max-min* . Este criterio elige como noción de equidad el que cualquier usuario tiene tanto derecho a obtener ancho de banda de la red como cualquier otro. Además, el criterio *max-min* intenta maximizar la cantidad de ancho de banda que se le asigna a aquel usuario que resulta con la menor cantidad de recurso. Una vez que se computa la mayor asignación posible a este usuario menos favorecido, se pasa a decidir la asignación que le corresponde al resto de los usuarios; es entonces razonable tratar de maximizar la asignación a aquellos usuarios menos favorecidos de entre los que han quedado tras el primer paso; y así sucesivamente hasta que a todos los usuarios tienen su asignación.

Un modo alternativo, y equivalente, de expresar la idea anterior es el siguiente: el criterio *max-min* persigue maximizar la asignación de aquel usuario *i* sujeto a la restricción de que aumentar la asignación de *i* no debe causar una disminución a otro usuario cuya

asignación sea menor o igual a la de i.

A continuación, presentamos una definición formal de equidad en el sentido *max-min* (Bertsekas y Gallager, 1992). En lo que sigue, asimilamos conexión a usuario. Sea una red descrita por un grafo $G = (\mathcal{N}, \mathcal{A})$, en donde $\mathcal{N}$ denota el conjunto de nodos de la red y $\mathcal{A}$, el conjunto de pares de nodos, esto es, de enlaces. Cada conexión p tiene asignado un trayecto fijo dentro de la red. Por razones de simplicidad, suponemos que las conexiones demandan una cantidad infinita de ancho de banda. Nótese que en el caso de que la demanda fuese finita, podríamos reducir éste al caso anterior sin más que añadir un enlace de capacidad igual a la demanda de la conexión a la entrada en la red de cada conexión.

Si denotamos como r_p a la tasa asignada por la red a la conexión p, la cantidad de tasa asignada en el enlace a de la red será

$$F_a = \sum_{\forall\, p \; atraviesa \; a} r_p$$

Si C_a es la capacidad del enlace a, podemos establecer las siguientes restricciones sobre el vector de tasas asignadas $r = \{r_p | p \in P\}$:

$$r_p \geq 0, \forall\, p \in P$$

$$F_a \leq C_a, \forall\, a \in \mathcal{A}$$

Un vector r que satisfaga las restricciones anteriores se dice que es *factible*.

Un vector de tasas r decimos que es *equitativo max-min* si es factible y si para cada $p \in P$, r_p no puede ser aumentada manteniendo r factible sin disminuir a su vez la tasa $r_{p'}$ de otra conexión p' para la cual $r_{p'} \leq r_p$. O de una forma más formal, r es equitativo *max-min* si es factible y si $\forall\, p \in P, \forall \overline{r} \mid r_p \leq \overline{r_p}$, existe siempre algún p' que cumpla $r_p \geq r_{p'}$ y $r_{p'} > \overline{r_{p'}}$.

Antes de presentar un algoritmo de obtención de un vector de tasas r que sea equitativo *max-min*, definiremos el concepto de enlace de cuello de botella: dado un vector de tasas r factible, decimos que a es un enlace de *cuello de botella* con respecto a r para la conexión p que atraviesa el enlace a, si $F_a = C_a$ y $r_p \geq r_{p'}$ para toda conexión p' que atraviese el enlace a. Se puede demostrar (Bertsekas y Gallager (1992), 527) que

> Un vector de tasas factible r es equitativo *max-min* si y sólo si cada conexión resulta con un enlace de cuello de botella con respecto a r

Presentamos a continuación un algoritmo de obtención de un vector de tasas equitativo *max-min*. La idea del algoritmo es empezar con un vector de tasas idénticamente cero y aumentar simultáneamente la tasas asignadas en todos los enlaces hasta que, en uno o más enlaces a, se cumpla $F_a = C_a$. En este punto, a todas las conexiones que atraviesa un enlace saturado dado, es decir, un enlace a tal que $F_a = C_a$, se le asigna la misma tasa. Este enlace saturado será el enlace de cuello de botella de todas las conexiones que lo atraviesen. En el paso siguiente del algoritmo, se aumentan simultáneamente la tasa asignada al resto de

conexiones, hasta que uno o más enlaces se saturen. Nótese que, si bien estos enlaces saturados pueden ser atravesados por conexiones cuyo cuello de botella haya sido encontrado en el paso anterior —aunque a una tasa inferior—, sólo constituyen enlaces de cuello de botella para las conexiones que lo atraviesan pero que no atraviesan ningún enlace saturado en el paso anterior. El algoritmo continúa paso a paso aumentando uniformemente la tasa de aquellas conexiones que no atraviesan ningún enlace saturado. Cuando todas las conexiones atraviesan al menos un enlace saturado, el algoritmo termina.

El algoritmo se especifica con mayor precisión a continuación. A^k denota el conjunto de enlaces no saturados al comienzo de la iteración k-ésima, y P^k denota el conjunto de conexiones que no atraviesan ningún enlace saturado al comienzo de la iteración k-ésima. Asimismo, n_a^k denota el número de conexiones que atraviesan el enlace a y que pertenecen a P^k; nótese que n_a^k es también el número de conexiones que compartirán la capacidad de a que aún no ha sido asignada. Finalmente, $\tilde{r}^k$ es el incremento aplicado a las conexiones de P^k en la iteración k-ésima

Condiciones iniciales: $k = 1, F_a^0 = 0, r_p^0 = 0, P^1 = \mathrm{P}$, y $A^1 = \mathcal{A}$

1. $n_a^k =$ número de conexiones $p \in P_k$ que atraviesan a
2. $\tilde{r}^k = min_{a \in A^k} \left\{ (C_a - F_a^{k-1})/n_a^k \right\}$
3. $r_p^k = \begin{cases} r_p^{k-1} + \tilde{r}^k & \text{si } p \in P^k \\ r_p^{k-1} & \text{en otro caso} \end{cases}$
4. $F_a^k = \sum_{p \text{ que atraviesa } a} r_p^k$
5. $A^{k+1} = \{a | C_a - F_a > 0\}$
6. $P^{k+1} = \{p |\ p \text{ no atraviesa ningún enlace } a \in A^{k+1}\}$
7. $k = k + 1$
8. Si P^k está vacío, finaliza; si no, ves a 1

En cada iteración k, se añade una misma cantidad a todas las conexiones que aún no atraviesan ningún enlace saturado, por lo que, en cada iteración k, todas las conexiones de P^k tienen la misma tasa. Es más, todas las conexiones de P^k que atraviesan un enlace que se satura en la iteración k-ésima tendrán asignada una tasa tan grande como cualquier otra conexión y, por tanto, este enlace será su cuello de botella. Así, cuando finalice el algoritmo, cada conexión tendrá su enlace de cuello de botella y, según se ha enunciado anteriormente, el vector de tasas resultante será equitativo *max-min*.

El criterio de equidad *max-min* fue interpretado por Charny (1994) del siguiente modo. Las conexiones que compiten por obtener ancho de banda en un nodo determinado pueden clasificarse en:

1. conexiones *limitadas* (*constrained*), que son aquellas que no pueden aprovechar su parte equitativa de ancho de banda en el nodo debido a que la tasa alcanzable viene limitada por la misma fuente o por el ancho de banda disponible en otros nodos del trayecto de la conexión; y

2. conexiones *no limitadas* (*unconstrained*), para las que el ancho de banda que pueden alcanzar está limitado por el disponible en el nodo considerado, que se denomina nodo de cuello de botella para esa conexión.

Sobre esta clasificación, el criterio de equidad *max-min* establecería que:

1. Toda conexión debe encontrar al menos un nodo que sea su cuello de botella a lo largo de su trayecto.

2. La tasa asignada a las conexiones no limitadas debe ser la misma y venir dada por la cantidad Λ:

$$\Lambda = \frac{\mu - \sum_{i \in Limitadas} \lambda_i}{N - M} \tag{2.1}$$

   donde μ es la capacidad del enlace, λ_i es la tasa asignada por la conexión limitada i ($\lambda_i < \Lambda$ para todo i), N es el número total de conexiones, y M es el número de conexiones limitadas.

A partir de esta interpretación, Charny (1994) diseñó un algoritmo distribuido que ajustase dinámicamente las tasas de las conexiones para mantener la equidad *max-min* según cambian las conexiones.

2.3.3.2 Soporte de ancho de banda mínimo garantizado en ABR

Si bien la garantía de un ancho de banda mínimo por conexión no fue uno de los objetivos iniciales especificados para el servicio ABR, su introducción, en la forma de una tasa MCR, abrió el camino para ampliar el campo de aplicación del servicio ABR (Bonomi y Fendick, 1995). Veamos algunas nuevas aplicaciones. Por un lado, las líneas alquiladas constituyen recursos dedicados por la red a un cliente corporativo, los cuales suelen estar dimensionados para satisfacer la demanda de la hora punta, si bien a costa de un uso ineficiente durante el resto del día. Su sustitución por conexiones ABR podría perfectamente cursar la demanda durante la mayor parte del tiempo, pero no ofecería ninguna garantía respecto a la demanda durante la hora punta del día. La introducción de un ancho de banda mínimo garantizado reduciría el riesgo de la migración hacia ABR, aunque la garantía no fuese suficiente para cubrir las necesidades de hora punta. Además, esta transición supondría una reducción del coste del servicio. Por otro lado, la garantía de MCR en ABR permitiría el soporte de aplicaciones de tiempo *casi* real, entendiendo por tales aquellas que han sido desarrolladas recientemente para su soporte sobre Internet y que ofrecen prestaciones similares a las aplicaciones de tiempo real pero con tolerancias mayores frente al retardo. Tales aplicaciones necesitan típicamente cierto ancho de banda mínimo garantizado, por ejemplo, correspondiente a una tasa mínima de codificación de un flujo de vídeo, pero pueden aprovechar el ancho de banda no ocupado.

Si tenemos en cuenta el ancho de banda mínimo garantizado MCR, el criterio de equidad empleado para juzgar la distribución de ancho de banda en ABR ha de ser modificado. Efectivamente, el criterio *max-min* no es aplicable directamente en una situación de mínimos garantizados. Veamos a continuación cuatro mecanismos de asignación equitativos que tienen en cuenta la existencia de mínimos garantizados (Hughes, 1994). Para su formulación haremos uso de la expresión 2.1 y denominaremos MCR_j al ancho de banda mínimo garantizado para la conexión j.

Criterio aditivo Una primera aproximación es tomar como ancho de banda disponible a repartir, el ancho total disponible para ABR menos el ancho de banda empleado por los mínimos MCR de cada conexión. A partir de aquí establecemos que toda conexión tiene derecho a una fracción equitativa del ancho de banda disponible, la cual se sumará a su mínimo MCR. En otras palabras, aplicamos el criterio de equidad *max-min* al ancho de banda disponible —excluyendo el ancho de banda empleado por los mínimos MCR— y a aquella parte de ancho de banda de cada conexión por encima del mínimo MCR correspondiente.

$$\Lambda_j = \frac{\mu - \sum_{i \in \textit{ Limitada}} \lambda_i}{N - M} + MCR_j \tag{2.2}$$

Nótese que el valor mínimo de $\sum_{i \in \textit{ Limitada}} \lambda_i$ es la suma de los mínimos garantizados de las conexiones limitadas.

Criterio de mínimos Otro modo de definir la equidad y tener en cuenta el mínimo garantizado MCR es determinar que cada conexión obtenga una fracción equitativa *max-min* del ancho de banda disponible total, pero que, cuando el valor equitativo *max-min* de una conexión sea menor que su mínimo garantizado MCR, se le asigne este valor MCR.

$$\Lambda_j = max\left(MCR_j, \frac{\mu' - \sum_{i \in \textit{ Limitada}} \lambda_i}{N - M}\right) \tag{2.3}$$

en donde μ' es inicialmente μ y, en iteraciones sucesivas, toma el valor

$$\mu' = \sum_{j \in \textit{ Limitada y } \Lambda_j \neq MCR_j} \Lambda_j$$

hasta que

$$\sum_{j \in \textit{ Limitada}} \Lambda_j = \mu$$

Criterio proporcional Podemos asignar a cada conexión una fracción proporcional a su mínimo garantizado MCR

$$\Lambda_j = \left(\mu - \sum_{i \in \textit{ Limitada}} \lambda_i\right) \frac{MCR_j}{\sum_{i \notin \textit{ Limitada}} MCR_i} =$$

$$= MCR_j + \left(\mu - \sum_{i \in \text{ Limitada}} \lambda_i - \sum_{i \notin \text{ Limitada}} MCR_i \right) \frac{MCR_i}{\sum_{i \notin \text{ Limitada}} MCR_i} \quad (2.4)$$

Criterio lineal Finalmente, podemos asignar a cada conexión su mínimo garantizado MCR más una fracción del ancho de banda disponible restante en función de un peso F_j:

$$\Lambda_j = MCR_j + F_j \left(\mu - \sum_{i \in \text{ Limitada}} \lambda_i - \sum_{i \notin \text{ Limitada}} MCR_i \right) \quad (2.5)$$

Si tomamos:

$$F_j = b\frac{1}{N-M} + (1-b)\frac{MCR_j}{\sum_{i \notin \text{ Limitada}} MCR_i}$$

estamos, de hecho, repartiendo equitativamente una parte b del ancho de banda disponible, mientras que el resto $1 - b$ se divide proporcionalemente al valor MCR de cada conexión. Este último criterio de equidad incluye como casos particulares el criterio aditivo —cuando $b = 1$— y el criterio de mínimos —cuando $b = 0$—.

Los criterios de equidad descritos arriba son simplemente una muestra de los muchos posibles. En realidad, no existe ninguna razón técnica que haga necesario la normalización de la relación entre MCR y la asignación de ancho de banda, sino que únicamente es necesario especificar el significado de MCR. De otro modo se estaría, por un lado, limitando de forma severa la capacidad de los operadores para diferenciar sus servicios mediante distintos esquemas de tarificación y, por otro, la de los suministradores para diferenciar sus productos mediante la incorporación de distintos algoritmos de soporte.

El esquema de tarificación que escoge el operador es un factor importante en la determinación del criterio de asignación de ancho de banda. Por ejemplo, el esquema proporcional es adecuado cuando la mayor parte del coste en la provisión del servicio viene ocasionado por la garantía del MCR, en cuyo caso es razonable dividir el ancho de banda disponible en proporción a este valor. Además, constituye una realización de la visión de que aquellos usuarios que necesitan más ancho de banda deben pagar más, o a la inversa, que quien pague más debe obtener más ancho de banda. No obstante, el criterio proporcional no tiene sentido cuando los usuarios no solicitan ningun mínimo garantizado. En tal caso, el esquema lineal es un compromiso entre ambos grupos de usuarios.

En el caso de redes corporativas, la cuestión del coste aparece más ambigua: nadie —o todos— pagan y, además, se asume que todos los usuarios son iguales. En tal situación, el esquema aditivo o el esquema de mínimos podrían ser adecuados, en cuanto que reflejan esta noción de igualdad, si bien es cierto que esta pretendida igualdad deja de ser relevante una vez que se permite a los usuarios solicitar distintos mínimos garantizados.

2.3.4 El servicio ABT

Como se ha introducido ya en la sección 2.1.3, durante la vida de una conexión del tipo ABT, la tasa BCR de los bloques ATM sucesivos se renegocia dinámicamente con la

red. Un bloque ATM se delimita mediante dos células de gestión de recursos (RM), lo cual permite que la renegociación de la tasa BCR se lleve a cabo mediante una petición a la red a través de una célula RM.

ABT aprovecha una característica única de ATM: que el procedimiento de establecimiento de conexión y el de asignación de ancho de banda pueden ejecutarse separadamente. En otras palabras, el ancho de banda puede reservarse y asignarse a escala de bloque, no de conexión

La renegociación de ancho de banda por bloque puede efectuarse según dos procedimientos, que corresponden a las dos clases de servicio ABT: ABT con transmisión retardada (*ABT with Delayed Transmission*, ABT/DT) y ABT con transmisión inmediata (*ABT with Immediate Transmission*, ABT/IT).

El servicio ABT garantiza parámetros de calidad de servicio, tales como tasa de pérdidas aceptable, CTD máximo y CDV de pico, a aquellas conexiones que repeten el patrón de tráfico que definen durante el establecimiento a través de los valores PCR, SCR y MBS. ABT sería por tanto un servicio de calidad garantizada, al igual que rt-VBR. En las implementaciones prácticas de ABR, tales como el *Fast Reservation Protocol* propuesto por Boyer y Tranchier (1992), se asume que SCR es cero. Por tanto la red no asume ningún tipo de compromiso de calidad de servicio. Es en esta situación en la que ABT puede ser denominado servicio *best-effort*.

2.3.4.1 ABT con transmisión retardada

En ABT/DT, el usuario transmitirá el bloque ATM una vez recibida aceptación por parte de la red a su petición de renegociación de tasa BCR. Si el tráfico fuese conforme con el descriptor de tráfico de fuente (PCR, SCR, MBS), entonces toda renegociación de tasa BCR deberá ser aceptada por parte de la red en un plazo finito de tiempo.

El proceso de renegociación de tasa BCR se lleva a cabo como sigue:

1. cuando una fuente desea aumentar su tasa BCR—p.ej., para transmitir un bloque—, enviará una célula RM de petición a la red;

2. está petición llega al primer nodo de la red, quien comprueba si puede asignar el aumento de ancho de banda que se solicita o no;

3. si puede soportar tal asignación, la petición se hace progresar al siguiente nodo, y así sucesivamente hasta el nodo de salida de la red, el cual devolverá un reconocimiento hacia la fuente, con lo que la fuente podrá transmitir su ráfaga;

4. si algún nodo no puede soportar la asignación, descartará la petición, y los recursos asignados hasta el momento en otros nodos se liberarán tras un *time-out*;

5. opcionalmente, la petición de aumento de tasa BCR puede ser modificada a la baja por la red;

6. alternativamente, el usuario puede sistemáticamente solicitar aumentos máximos de ancho de banda.

En ABT, se pueden adoptar distintas estrategias de asignación de ancho de banda. Por ejemplo, puede asignarse a cada ráfaga la totalidad del ancho de banda disponible en cada momento; de este modo, sólo una ráfaga puede ocupar el enlace. O bien, puede asignarse sólo una fracción del ancho de banda disponible en cada momento; en este caso, más de una ráfaga puede ocupar el enlace.

En ABT/DT, el usuario, mientras dura el proceso de renegociación de tasa BCR, puede transmitir a la tasa previa. Por ello, un tipo de tráfico muy apropiado para soportarse sobre ABT/DT es el resultantes de una superposición de fuentes on/off y de fuentes CBR. Mientras dura la renegociación de BCR, algunos flujos resultarán bloqueados, pero otros no verán alterada la respuesta de la red.

2.3.4.2 ABT con transmisión inmediata

En este caso, el usuario transmite el bloque ATM sin esperar a recibir reconocimiento por parte de la red. En ABT/IT, por tanto, el retardo que experimenta la aplicación es menor que el que experimentaría si utilizase ABT/DT. Si el tráfico fuese conforme con el descriptor de tráfico de fuente (PCR, SCR, MBS), entonces la probabilidad de que la transferencia del bloque ATM no se lleve a cabo con éxito, esto es, que el bloque ATM sea descartado, debe ser menor que un umbral dado

A diferencia de ABT/DT, el proceso de renegociación de BCR en ABT/IT se lleva a cabo según sigue:

1. cuando una fuente desea enviar un bloque, le antepone una célula RM de petición de ancho de banda y la entrega a la red;

2. cada nodo de la red comprueba si dispone de recursos para aceptar la ráfaga bajo las condiciones que indica la célula RM de cabeza; en caso afirmativo, la hace progresar hacia su destino; si no, la descarta;

3. la ráfaga va seguida de una célula RM de liberación de ancho de banda.

ABT/IT es apropiada para aquellas aplicaciones que no pueden esperar a obtener la conformidad de la red para modificar su comportamiento. Ello puede deberse a que la duración de la ráfaga es pequeña comparada con el tiempo de ida y vuelta de la conexión —p.ej., una trama 802.3—, con lo que una conexión ABT/DT sería ineficiente; o bien a que la aplicación no puede asumir los retardos de acceso a la red —p.ej., servicios sin conexión, *codecs* de tiempo real, etc.—. Estas aplicaciones suelen generar ráfagas de datos —posiblemente separadas por silencios—, cuyas transferencias a través de la red se negocian independientemente —p.ej., ficheros de datos, imágenes fijas—

2.3.5 El servicio GFR

El servicio UBR+ fue propuesto en el seno del ATM Forum para proporcionar un servicio con un determinado nivel de calidad garantizada y de equidad (Guerin y Heinanen, 1996). Tal servicio debería requerir una interacción mínima entre el usuario y la red;

además, debía ser menos complejo que ABR. Para desvincular la definición de este nuevo servicio de la de otros servicios preexistentes, el ATM Forum decidió utilizar el término más genérico de *Guaranteed Frame Rate* (GFR).

En GFR, el descriptor de tráfico de fuente contiene una tasa máxima PCR —y su correspondiente CDVT—; una tasa mínima de servicio, que se corresponde con un valor MCR a nivel de célula ATM; y un tamaño máximo de trama, que se relaciona con un valor MBS a nivel de célula ATM.

Por su parte, la calidad de servicio se ofrece en los términos siguientes. Se garantiza una tasa aceptable de pérdidas CLR, para aquellas tramas emitidas por debajo de la tasa mínima de servicio. La entrega de las tramas generadas por encima de la tasa mínima de servicio —tráfico excedente— se supedita, no obstante, a la disponibilidad de recursos en la red, esto es, se aplica un principio *best-effort*. Finalmente, la asignación de ancho de banda al tráfico excedente deberá ser equitativo.

En cuanto a los mecanismos de soporte, se contempla la utilización del marcado de las células, con dos funciones. Por un lado, determinar qué células cumplen el test de conformidad respecto a PCR: cuando un célula no es conforme, la red puede marcar esa misma célula y las restantes células de la misma trama. Por otro lado, en algunas implementaciones, identificar las células a las que es aplicable la calidad de servicio enunciada en el párrafo anterior. En cualquier caso, el marcado debe ser consistente, esto es, todas las células de una trama deben ser marcadas de forma idéntica. En cuanto al segundo caso, si el marcado sólo se basa en criterios de conformidad con los parámetros de tráfico, independientemente de las condiciones de congestión, se puede afectar negativamente a la equidad en la asignación de ancho de banda al tráfico en exceso. Veamos por qué. El tráfico en exceso es, por definición, tráfico no conforme. Sin embargo, en GFR se espera que el ancho de banda disponible sea distribuido entre las conexiones con tráfico en exceso de una forma equitativa. Si, como aplicación de la segunda función de marcado, se marcan todas las células no conformes como no susceptibles de aplicárseles la calidad de servicio, se está discriminando negativamente a las conexiones que atraviesen mayor número de conmutadores.

También se puede soportar GFR mediante algoritmos de planificación por conexión. En particular, puede asegurarse que cada conexión es servida a una tasa mayor o igual a MCR mediante disciplinas equitativas, como las descritas en la sección 3.3.2. Alternativamente, se puede soportar GFR mediante disciplinas FIFO en los nodos, y confiar en la función del marcado para identificar las células a las que es aplicable la garantía de servicio.

2.3.6 Discusión

Se pueden encontrar diferencias entre las clases de servicio identificadas a priori como alternativas de soporte de servicio *best-effort* sobre redes ATM, en términos de la naturaleza del servicio y de los mecanismos de soporte.

En primer lugar, vamos a comparar las clases de servicio según la naturaleza del servicio. La clase nrt-VBR asume un compromiso al respecto de la tasa de pérdidas para aque-

llas conexiones que cumplen el contrato de tráfico acordado durante el establecimiento de la conexión. No se garantiza la tasa de pérdidas para aquellas células que son emitidas incumpliendo el contrato de tráfico. Asimismo, se asume un cierto grado de protección entre usuarios, esto es, aquellas conexiones que exceden su contrato de tráfico no deberán afectar negativamente a la tasa de pérdidas negociada por aquellas conexiones que sí se mantienen bajo su contrato de tráfico.

La clase UBR no ofrece ningún compromiso vinculado al comportamiento del tráfico del usuario del servicio. Ni la tasa de pérdidas experimentada por la conexión, ni el retardo de transferencia experimentado por las células de la misma, quedan garantizadas bajo UBR. Por último, la equidad entre las conexiones no puede presumirse, aunque puedan disponerse mecanismos en los nodos encaminados a mantener un cierto grado de equidad.

La clase ABT, cuando se toma SCR=0, asume un modelo de servicio igual al de la clase UBR. Cuando se toma SCR$\neq$0 pero no se solicitan requisitos temporales, el modelo de servicio es igual al de la clase nrt-VBR.

La clase ABR se compromete a ofrecer una baja tasa de pérdidas a aquellas conexiones que se adhieran al comportamiento de referencia especificado. No existe ningún compromiso, no obstante, en cuanto al retardo de transferencia. Asimismo, si los extremos de la conexión no cumplen el comportamiento de referencia para ABR, la tasa de pérdidas no se garantiza. La equidad entre las conexiones, teniendo en cuenta el valor MCR, se asume en el servicio. Asimismo, se asume un cierto grado de protección entre usuarios, de modo que aquellas conexiones que no cumplan el comportamiento de referencia no provoquen una degradación en la calidad observada por aquellas conexiones que sí lo cumplen.

Finalmente, la clase GFR ofrece las mismas garantías que ABR, pero se vinculan únicamente al cumplimiento de un contrato de tráfico por parte del usuario, mucho más sencillo que el acordado para la clase nrt-VBR.

En segundo lugar, analicemos las diferencias entre los mecanismos de provisión de cada clase de servicio. La clase nrt-VBR incorpora un contrato de tráfico y su soporte es inherentemente preventivo o de bucle abierto. Efectivamente, tal contrato de tráfico puede ser cumplido por el mismo usuario mediante conformación de tráfico y obligado por la red mediante funciones UPC. Además, dado que se asume que los nodos reservan recursos con el fin de poder garantizar la calidad acordada en el contrato de tráfico de cada conexión, la red debe realizar funciones CAC.

La clase ABT/DT, cuando SCR=0, es conceptualmente equivalente a nrt-VBR en cuanto a soporte, aunque la escala temporal de actuación de los mecanismos respectivos es distinta. La clase ABT/IT, cuando SCR=0, es conceptualmente equivalente a UBR en cuanto a soporte.

El soporte de UBR, por su parte, opera en bucle abierto. Cada nodo o terminal puede implementar mecanismos de descarte selectivo de células o de planificación, aunque no se esté sujeto a ningún contrato de tráfico. Análogamente, el soporte de GFR debe confiar en estos mecanismos locales de planificación, aunque en este caso sí se está sujeto a un contrato de tráfico.

Por contra, el soporte de ABR opera en bucle cerrado. La fuente de cada conexión efectúa un conformado dinámico del tráfico que entrega a la red en función de la reali-

mentación que recibe de ésta. Por su parte, la red puede obligar este conformado mediante funciones UPC. Cuando se negocia un valor MCR distinto de cero, la red deberá reservar recursos para evitar que la tasa de emisión permitida por la red baje del valor MCR negociado y para que se mantenga una tasa de pérdidas baja. En consecuencia, la red deberá proporcionar CAC. En este caso, la equidad y la protección se consiguen mediante mecanismos locales en los nodos.

2.4 Conclusiones

En la sección 2.2, se ha elaborado una definición precisa de servicio *best-effort* y se han apuntado aquellos aspectos que aparecen cuando se plantea la provisión de un servicio *best-effort* sobre red ATM. A continuación, en la sección 2.3, se han descrito las clases de servicio que el ATM Forum y el UIT-T han definido para proporcionar este tipo de servicio. En esta sección, se procede a analizar cuán adecuada es cada una de estas clases de servicio a la definición aportada en la Tesis.

Según se estableció en la sección 2.2, un servicio *best-effort* es aquél en el que el usuario percibe una calidad de servicio dependiente del impredecible estado de ocupación de los recursos de la red. Esta definición conlleva necesariamente que la red no reserve, en tiempo de establecimiento de conexión, recursos para la provisión del servicio. Ello nos lleva a excluir a la clase de servicio nrt-VBR de la denominación de servicios *best-effort*, por cuanto que reserva recursos con el objetivo de ofrecer garantías respecto de pérdidas. Es cierto, sin embargo, que el retardo de transferencia es un parámetro de calidad de servicio que en nrt-VBR se ofrece sin garantías y según el paradigma tradicional de servicio *best-effort*.

También en la sección 2.2 se estableció que las redes ATM modificaban significativamente el escenario tradicional de provisión de los servicios *best-effort*. Esta modificación se fundamentaba, por un lado, en la homogeneidad tecnológica de las redes ATM, y por otro, en que los servicios que se proveen sobre redes ATM tienen un carácter comercial. En este escenario, Lefelhocz y otros (1996) proponen que la provisión de servicios de tipo *best-effort* se reformule y se ajuste al siguiente paradigma:

1. La calidad de servicio percibida por un usuario en concreto no deberá depender del comportamiento particular de otros usuarios, dado que ya no es factible asumir la cooperación de los usuarios; por tanto, los recursos de la red han de asignarse de forma controlada e imponiendo límites en su ocupación por parte de algunos usuarios.

2. La red deberá proporcionar a los usuarios información suficientemente detallada sobre el estado de la red, de modo que tales usuarios puedan utilizar de forma eficiente los recursos disponibles en cada momento en la red.

A partir de esta reformulación, Lefelhocz y otros (1996) plantean que la provisión de un servicio *best-effort* sobre red ATM se concrete en el siguiente contrato informal:

- Por un lado, los requisitos de un usuario de servicio *best-effort* serían los siguientes:

al respecto del retardo de transferencia, *as soon as possible*; al respecto del ancho de banda, *as much as possible*.

- Por otro lado, la red estaría en condiciones de establecer dos tipos de compromiso con el usuario, relativos y procedimentales:
  - Las *garantías relativas* serían aquellas en virtud de las cuales la red se compromete a que el ancho de banda recibido por aquellas conexiones que compartan el mismo trayecto o el mismo cuello de botella, quede repartido equitativamente. Nótese que, en un servicio *best-effort*, el usuario no paga por un servicio con una calidad cuantificable, sino por un modo de proporcionar servicio: es lógico pues exigir que este modo sea equitativo.
  - Las *garantías procedimentales* harían saber a los usuarios qué pueden esperar si responden adecuadamente a la realimentación de la red. Efectivamentet, aunque no exista un control de admisión, el usuario podría conseguir una baja tasa de pérdidas si la red proporcionase información respecto al ancho de banda disponible en cada momento. En este caso la garantía procedimental consistiría en garantizar una baja tasa de pérdidas siempre que el usuario se comportase de acuerdo a la información proporcionada por la red.

El servicio UBR es un servicio *best-effort* que no incorpora garantías relativas ni procedimentales.

El servicio ABT (con SCR$\neq$0) sí establece garantías relativas, pero a costa de reservar recursos en tiempo de establecimiento de conexión, por lo que no es un servicio *best-effort*. En cuanto a las garantías procedimentales, éstas se formulan de la siguiente forma: sólo si el perfil del tráfico de usuario se ajusta al descriptor de tráfico, la red garantiza los parámetros de calidad de servicio. Por su parte, ABT (con SCR=0) sólo reserva los recursos a nivel de bloque, pero no establece garantías relativas ni procedimentales.

El servicio ABR establece garantías relativas y procedimentales. Veamos cómo lo consigue.

Según se afirmó en la sección 2.2, la homogeneidad tecnológica en las redes ATM modifica el entorno de provisión de servicio *best-effort* con respecto al que existía tradicionalmente, ofreciendo nuevas posibilidades a los diseñadores de protocolos. ABR aprovecha esta ventaja y define un mecanismo de realimentación que es explícito y cuya escala temporal de respuesta es mucho menor que la del mecanismo de realimentación implícita en el servicio IP, o del que podría ser igualmente en el servicio UBR. Ello permite que, en ABR, la calidad del servicio sea percibida por el usuario en términos de tasa de células permitida y/o cursada por la red, por cuanto que esta tasa es el valor realimentado por el bucle de control que se define en la operación de ABR. La existencia de esta comunicación explícita del estado de la red permite a ABR establecer garantías procedimentales de servicio: si el usuario emite a una tasa menor o igual a la tasa de emisión permitida por la red, que le comunica mediante realimentación, la red garantiza una baja tasa de pérdidas de células.

Además, ABR incluye en su definición garantías relativas. Sin embargo, tales garantías no son intrínsecas al control de flujo por realimentación. Estas garantías sólo se pueden

ofrecer mediante mecanismos de asignación de recursos implantados en los nodos, tales como algoritmos de planificación y de asignación de *buffers*. Nótese, no obstante, que estos mecanismos no se definen en *ATM Forum Traffic Management 4.0*, por lo que no están implícitos en el control de flujo que se define para ABR.

El servicio GFR, por su parte, también establece garantías relativas en su definición. Sin embargo, sólo ofrece garantías procedimentales en lo relativo al tráfico por debajo de la tasa mínima de servicio. No ocurre así en lo relativo al tráfico excedente; este se sirve según el paradigma tradicional de servicio *best-effort*, pero el usuario sólo puede conocer la calidad de servicio que está recibiendo su tráfico excedente mediante realimentación implícita.

Concluimos por tanto que, primero, los servicios UBR, ABT (con SCR=0), ABR y GFR son ejemplo de servicios *best-effort* según la definición de la sección 2.2. Segundo, los servicios UBR y ABT (con SCR=0) no ofrecen garantías relativas ni procedimentales de servicio. Tercero, los servicios ABR y GFR sí establecen garantías relativas en su definición, aunque tal garantía depende la incorporación de mecanismos locales de planificación de recursos en cada uno de los nodos de la red. Y cuarto, las garantías procedimentales de servicio, cuando no hay especificación de tasa mínima de servicio, sólo pueden formularse cuando se incorpora un mecanismo de control de flujo por realimentación.

Así pues, de entre los servicios *best-effort* propuestos por el ATM Forum y el UIT-T, el servicio ABR definido por el ATM Forum es el servicio *best-effort* que se ajusta a la definición dada en la sección 2.2 y que, además, puede proporcionar garantías relativas y procedimentales. No obstante, la provisión efectiva de las garantías relativas está condicionada a la incorporación de mecanismos locales de planificación de recursos en cada uno de los nodos de la red.

En el capítulo siguiente, se estudian los mecanismos de planificación más adecuados para soportar una formulación del servicio ABR con garantías relativas, así como el papel que desempeña cada uno de ellos en la provisión del mismo. En concreto, se estudiarán los algoritmos de planificación equitativa para transmisión de células y los mecanismos de gestión de *buffers* por conexión.

En el capítulo 4, se estudiará el control de flujo normalizado en ABR así como los aspectos no normalizados, en concreto los algoritmos de conmutador, que generan la señal de realimentación. Una vez estudiado cada uno de los elementos que sirven para proveer el servicio ABR, en el capítulo 5 se presenta un nuevo algoritmo de conmutador para ABR.

Capítulo 3

La asignación equitativa de recursos para la provisión de servicios *best-effort*

En una red de computadores, existen diversos tipos de recursos compartidos a los que los usuarios acceden: impresoras, sistemas de ficheros, enlaces de transmisión de larga distancia, servidores WWW, etc. La compartición de recursos introduce ineludiblemente el problema de la contención entre peticiones de utilización de los recursos, para lo cual se necesita un algoritmo de planificación (*scheduling*) que decida qué petición, de entre las recibidas y en espera de servicio, servir a continuación.

La planificación realmente tiene dos aspectos ortogonales: por un lado, decidir el orden en que las peticiones son atendidas; por otro, gestionar la cola de peticiones que esperan recibir servicio. De este modo, un algoritmo de planificación asigna distintas calidades de servicio a los distintos usuarios que envían peticiones de utilización a un conjunto de recursos:

1. en la elección del orden de servicio de las peticiones recibidas, asigna diferentes retardos a cada usuario;

2. en la elección de qué peticiones de servicio descartar cuando no pueden mantenerse todas en espera dentro del sistema, asigna diferentes tasas de pérdida a cada usuario.

Aunque una red debe planificar el acceso a todos y cada uno de los recursos que se comparten, a nivel de capa ATM, dos son los tipos de recursos para los que se debe resolver su asignación:

1. el ancho de banda en un enlace; y

2. el espacio de almacenamiento (*buffers*) en un nodo.

La planificación es importante únicamente cuando se prevé que la secuencia de llegada de peticiones al sistema exhiba fluctuaciones aleatorias, que resultan en la aparición de

colas en los puntos de multiplexado de recursos. Así ocurre en las redes de conmutación de paquetes; sin embargo, no ocurre en las redes de conmutación de circuitos, donde el tráfico que generan las fuentes es regular y sin fluctuaciones significativas

Los requisitos que un algoritmo de planificación debe satisfacer depende del tipo de usuarios. Por un lado, las aplicaciones de servicio garantizado (véase la página 8) precisan garantías de servicio, por lo que los algoritmos de planificación deben ser capaces de garantizar unos márgenes máximo de retardo, mínimo de ancho de banda y máximo de pérdidas a las diferentes conexiones. Ello es posible por cuanto que la transmisión de los paquetes en los puertos de salida de cada conmutador que atraviesan las conexiones está gobernado por el correspondiente algoritmo de planificación, de modo que a cada conexión se le puede asignar:

1. diferentes retardos medios al elegir el orden de servicio;
2. diferentes anchos de banda al servir al menos un determinado número de paquetes de cada conexión dentro de un intervalo de tiempo dado; y
3. diferentes tasas de pérdida al otorgar más o menos *buffers*.

Por otro lado, las aplicaciones *best-effort* (véase la página 9) no requieren garantías de calidad de servicio, sino que precisan que la red sea equitativa en la asignación de los recursos entre las conexiones. Ello es posible por cuanto que la asignación del ancho de banda y de los *buffers* en cada uno de los conmutadores que atraviesan las conexiones está gobernado por el correspondiente algoritmo de planificación.

En esta Tesis, en cuanto que se estudia la provisión de servicios *best-effort*, el énfasis se ha puesto en cómo los algoritmos de planificación permiten asignar los recursos de una forma equitativa entre los usuarios. Esta función de los algoritmos de planificación es imprescindible para proveer un servicio ABR con garantías relativas, según se afirmó en la sección 2.4.

En la sección 3.1 se presenta una clasificación de los algoritmos de planificación para la asignación del ancho de banda, en adelante *algoritmos de planificación*, así como una enumeración de los parámetros de mérito de un algoritmo de planificación. En la sección 3.2 se analiza el algoritmo de planificación más común, la disciplina FCFS, mientras que en la sección 3.3 se presenta el conjunto de algoritmos denominados de planificación equitativa, que van a desempeñar un papel fundamental en la contribución de esta Tesis. En la sección 3.4 analizaremos los distintos algoritmos de planificación de espacio de almacenamiento, en adelante *mecanismos de gestión de* buffers .

3.1 La asignación de ancho de banda

En esta sección se estudian los algoritmos de planificación para la transmisión de paquetes en los puertos de salida de los conmutadores, que son los responsables de la asignación del recurso ancho de banda.

Podemos clasificar los algoritmos de planificación según dos criterios: en función de la conservatividad del algoritmo y de su arquitectura interna.

Un servidor es conservativo si sólo permanece inactivo cuando no hay ningún paquete en espera de ser transmitido. Ejemplos de algoritmos conservativos son GPS (véase la sección 3.3.1), WFQ (véase la sección 3.3.2), *Virtual Clock* (Zhang, 1991), *Delay*-EDD (Ferrari y Verma, 1990), WRR (véase la sección 3.3.4) y *Deficit Round-Robin* (DRR) (Shreedhar y Varghese, 1995). Un servidor no conservativo, en cambio, puede permanecer inactivo incluso si hay paquetes por transmitir. Por ejemplo, un servidor puede retrasar la transmisión de un paquete cuando espera la llegada en breve de un paquete con mayor prioridad, aunque en ese momento se encuentre inactivo; sin embargo, cuando el tiempo de transmisión de un paquete es corto, como es el caso en ATM, tal estrategia no es generalmente justificable. Además, con algoritmos de planificación no conservativos se puede conseguir que el tráfico que llega a los conmutadores siguientes sea más predecible, lo cual permite reducir tanto el tamaño de los *buffers* necesarios en las colas de salida como el *jitter* que sufre el retardo en la conexión. No obstante, los servidores no conservativos siempre tienen un retardo medio mayor que los conservativos. Ejemplos de servidores no conservativos son *Hierachical Round-Robin* (HRR) (Kalmanek y otros, 1990), *Stop-and-Go* (Golestani, 1991) y *Jitter*-EDD (Verma y otros, 1991).

Según su arquitectura interna, los algoritmos de planificación pueden clasificarse en dos grupos: los algoritmos de prioridad ordenada (*sorted-priority*) y los algoritmos enmarcados (*frame-based*). En los algoritmos de prioridad ordenada, existe una variable global asociada a cada enlace de salida del conmutador, denominada tiempo virtual. Cada vez que un paquete llega o es transmitido, el tiempo virtual se actualiza. A cada paquete que llega al sistema se le asocia una marca temporal que es función del tiempo virtual. Entonces los paquetes se ordenan según su marca temporal y se transmiten en el orden resultante. Ejemplos de algoritmos de prioridad ordenada son *Virtual Clock*, WFQ y *Delay*-EDD.

Dos son los factores que determinan la complejidad de implementación de los algoritmos de prioridad ordenada. En primer lugar, el cálculo de la marca temporal de cada paquete que llega al sistema, cuya complejidad depende del algoritmo. En WFQ, la actualización del tiempo virtual precisa procesar un máximo de N sucesos durante la transmisión de un solo paquete, donde N es el número de conexiones que comparten el enlace de salida, mientras que en *Virtual Clock*, las marcas temporales se computan en un tiempo $O(1)$. En segundo lugar, la actualización de la lista de prioridades y la elección del paquete con máxima prioridad. Ello tiene una complejidad $O(log_2 N)$.

En los algoritmos enmarcados, el tiempo se divide en intervalos —denominados periodos de trama— de una duración, bien fija, bien variable, durante cada uno de los cuales cada conexión transmite un número determinado de paquetes, en función del tráfico máximo que se le permite transmitir a la conexión durante un periodo de trama. Si la duración de la trama es fija, el servidor puede permanecer inactivo si las conexiones transmiten menos que la reserva efectuada para el periodo de trama actual. HRR y *Stop-and-Go* pertenecen a este tipo. En cambio, si la duración de la trama es variable —hasta un valor máximo—, cuando el tráfico de una sesión es menor que la reserva efectuada para el periodo de trama actual, el intervalo siguiente puede empezar con antelación. WRR y DRR son servidores de este tipo.

Desde el punto de vista de los usuarios y del proveedor de la red, un algoritmo de planificación debe satisfacer los siguientes requisitos:

Equidad en el reparto de los recursos Aceptamos que la asignación de recursos en un conmutador es equitativa si tal asignación satisface el criterio *max-min*. Este criterio fue ya definido, a partir de Bertsekas y Gallager (1992), en 21. En tal ocasión, la asignación de ancho de banda se realizaba a nivel global y de manera centralizada; además, las demandas de los usuarios eran infinitas. En cambio, un algoritmo de planificación toma únicamente decisiones de asignación de recursos a escala local. Redefiniremos, a continuación, el criterio de equidad *max-min* a nivel local. No obstante, la equidad en la distribución de recursos entre un conjunto de conexiones es un objetivo global. Bertsekas y Gallager (1992) demuestran que, si cada conexión limita su consumo de recursos a la mínima asignación equitativa a escala local que le corresponde de entre los conmutadores que atraviesa, se obtiene una asignación de recursos equitativa a escala global.

Intuitivamente, una asignación *max-min* asigna a aquel usuario con unas necesidades modestas aquello que solicita y reparte uniformemente los recursos sobrantes entre los usuarios con peticiones de importancia. Desde un punto de vista formal, una asignación *max-min* es aquella que (Keshav, 1997)

1. asigna los recursos en orden creciente de necesidades;
2. ningún usuario obtiene más recursos de los que solicita;
3. aquellos usuarios que no satisfacen sus necesidades, es decir, que no pueden obtener todos los recursos que solicitan, obtienen una cantidad igual de recursos.

Alternativamente, suponiendo una cantidad total μ_{total} de recursos y unas solicitudes ρ_i, una asignación que otorga una cantidad μ_i a cada usuario es equitativa en el sentido *max-min* (Demers y otros, 1989) si:

1. ningún usuario recibe más que lo que solicita, es decir, $\mu_i \leq \rho_i$;
2. ninguna asignación alternativa que satisfaga la condición 1 resulta en una asignación mínima mayor;
3. la condición 2 es recursivamente cierta si eliminamos el usuario con la asignación mínima y reducimos la cantidad total de recursos de forma acorde, $\mu_{total} \leftarrow \mu_{total} - \mu_{min}$.

Esta condición se puede formular como $\mu_i = \min(\mu_{fair}, \rho_i)$, donde μ_{fair}, que es la parte equitativa *max-min*, se toma tal que $\mu_{total} = \sum_{i=1}^{N} \mu_i$.

Desde un punto de vista operativo, una asignación equitativa en el sentido *max-min* se consigue de la siguiente manera. Supongamos un conjunto de usuarios $1, 2, \ldots, n$ con necesidades de recursos dadas por $x_1, x_2, \ldots, x_n$. Sin pérdida de generalidad, podemos suponer que las necesidades están ordenadas $x_1 \leq x_2 \ldots \leq x_n$. Además, la cantidad de recursos disponibles es igual a C. Inicialmente asignamos una cantidad C/n de recursos

al usuario con menores necesidades, esto es, el usuario 1; tal asignación puede ser mayor que lo que solicita el usuario 1, en cuyo caso $C/n - x_1$ queda como excedente. Tal exceso se distribuye uniformemente entre los $n - 1$ usuarios restantes, de modo que asignamos, en principio, al usuario 2 una cantidad $C/n + (C/n - x_1)/(n-1)$ de recursos; tal asignación puede ser quizás mayor que x_2, por lo que continuamos el proceso. El proceso termina cuando ningún usuario ha obtenido más recursos que los que solicita y cuando un usuario cuyas necesidades no han sido satisfechas, no haya obtenido menos recursos que los obtenidos por cualquier otro usuario con mayores necesidades.

Hasta este punto, se ha supuesto que todos los usuarios tienen el mismo derecho a la obtención de los recursos que se comparten. Se dan ocasiones en las que es deseable que algunos usuarios obtengan una parte mayor que otros (véase la sección 2.3.3.2). Más concretamente, sería deseable que la distribución de los recursos se ponderase según unos pesos $w_1, w_2 \ldots, w_n$ asociados con cada uno de los usuarios. Podemos, entonces definir una asignación equitativa ponderada en el sentido *max-min* como aquella que:

1. asigna los recursos en orden creciente de necesidades normalizadas al peso asociado a cada usuario;

2. ningún usuario obtiene más recursos de los que solicita.

3. aquellos usuarios que no satisfacen sus necesidades, es decir, que no pueden obtener todos los recursos que solicitan, obtienen una cantidad de recursos proporcional a su peso w_i.

Protección entre usuarios Un algoritmo de planificación debe impedir que el comportamiento indeseable por parte de alguna conexión —que envíe paquetes a una tasa mayor que su tasa equitativa *max-min*— afecte a la asignación de recursos decidida para otras conexiones. En otras palabras, un algoritmo de planificación debe ser capaz de garantizar la calidad de servicio a una conexión incluso en presencia de otras conexiones que exhiban un comportamiento no deseable.

La relación entre equidad y protección es la siguiente: un algoritmo de planificación equitativo automáticamente proporciona protección, pues limita la cantidad de recursos que puede obtener una conexión con comportamiento no deseable a su parte equitativa. La proposición inversa no es cierta: un algoritmo de planificación que proporciona protección no necesariamente es equitativo, pues la fracción que protege no necesariamente ha de ser la que le corresponda a cada conexión según un criterio de equidad.

La protección en la asignación de recursos en los conmutadores es necesaria incluso si se disponen mecanismos de UPC en los puntos de acceso a la red. Aunque los flujos de células estén conformes en el acceso a la red, estos flujos pueden agolparse y hacerse más esporádicos a medida que progresan por la red (Cruz, 1991a)(1991b).

Márgenes de calidad de servicio Para dar soporte a aplicaciones de servicio garantizado, un algoritmo de planificación debe ser capaz de garantizar margenes de calidad

especificables por el usuario. Tres son los tipos de parámetros que un algoritmo de planificación debe ser capaz de regular para cada conexión: el ancho de banda, el retardo de transferencia y el *jitter* asociado, y la tasa de pérdidas.

Sencillez y eficiencia en el control de admisión También en relación con el soporte de aplicaciones de servicio garantizado, un algoritmo de planificación debe facilitar el control de admisión. Además, no debe provocar la infrautilización de los recursos. Por ejemplo, con el algoritmo FCFS, podemos garantizar una tasa de pérdidas si limitamos el número de conexiones en la red y el tamaño de las ráfagas de cada conexión; no obstante, tal control de admisión provoca una baja utilización de los recursos.

Simplicidad en la implementación En una red ATM, el tiempo del que se dispone para efectuar una decisión de planificación es muy reducido. A una velocidad STM-1/OC-3 el tiempo de transmisión de una célula es de 3 μs. Entendemos por decisión de planificación la que se toma para determinar el orden de transmisión de un paquete llegado al sistema, o bien para elegir el paquete que transmitir. Un algoritmo de planificación para redes de alta velocidad requerirá, por tanto, la ejecución de un número pequeño de operaciones sencillas. Además, tal algoritmo deberá, preferentemente, ser realizable sin coste excesivo en *hardware*.

En particular, el número de operaciones necesarias para llevar a cabo una decisión de planificación debe ser lo menos dependiente posible del número de conexiones con paquetes planificados. La complejidad de implementación de un algoritmo de planificación depende de la arquitectura interna. Si el algoritmo es de prioridad ordenada, tres son los pasos que sigue el procesado de cada paquete:

1. Cálculo de la marca temporal asociada al paquete.

2. Inserción del paquete en la estructura de datos: cuando la célula llega a su cola y ésta está vacía, la inserción en una estructura de datos en árbol o en *heap* tiene una complejidad $O(log_2 N)$, donde N es el número de conexiones con paquetes planificados.

3. Selección del paquete con menor marca temporal para su transmisión: la extracción de un elemento requiere $O(log_2 N)$ operaciones elementales.

En cambio, los algoritmos enmarcados, tales como WRR y DRR, tienen una complejidad de implementación $O(1)$, y no requiren de cómputo de marcas temporales

Cuando la realización del algoritmo es sobre tecnología VLSI, es casi tan sencillo realizar la lógica de un algoritmo complicado como la de uno sencillo. La fuente de complejidad es la cantidad de memoria que se ha de disponer para almacenar el estado de planificación —consistente en punteros a colas de paquetes o variables que registran el servicio recibido por cada conexión—. En particular, el tiempo necesario para acceder al estado de planificación limita la máxima complejidad admisible en el algoritmo.

3.2 La disciplina de servicio FCFS

La disciplina de servicio FCFS (*First-Come First-Served*) es la disciplina más ampliamente implantada. FCFS sirve las peticiones de servicio en el orden en el que llegan al sistema. FCFS es la disciplina de servicio del tipo de los algoritmos de prioridad ordenada más sencilla: la marca temporal asignada a cada paquete es simplemente el instante de tiempo su llegada al sistema; los paquetes, por tanto, se sirven según el orden de llegada al sistema.

FCFS no ofrece protección entre conexiones: si una conexión es muy esporádica, la llegada de ráfagas de paquetes ocasionarán un aumento del retardo de transferencia observado por el resto de las conexiones. Por tanto, la asignación de ancho de banda entre conexiones no es equitativo. Además, FCFS puede ocasionar agrupamiento (*clumping*) en los flujos de paquetes: tal fenómeno se produciría cuando un grupo de paquetes de un flujo determinado llegase a un conmutador correctamente espaciados en el tiempo pero, debido a que fuesen colocados en la cola y no lleguasen paquetes pertenecientes a otros flujos, los paquetes abandonasen el conmutador a la tasa de salida del enlace. El agrupamiento podría ocasionar congestión en los conmutadores siguientes.

Cuando hay presente un control de flujo en la red para el soporte de aplicaciones *best-effort*, si se implanta FCFS en los conmutadores de la red, todas las conexiones experimentarán el mismo retardo medio de transferencia, aunque sólo un número reducido de ellas tenga un comportamiento acorde con la realimentación enviada por la red. Es más, FCFS puede ocasionar inequidad incluso cuando todas las conexiones se comportan correctamente; este fenómeno, conocido como *flow segregation* lo han descrito Floyd y Jacobson (1992).

Durante periodos de sobrecarga en la red, cuando los recursos son escasos, FCFS recompensa los abusos de utilización de ancho de banda, a costa de las conexiones que cooperan con el control de flujo —tal como el mecanismo de ventana adaptativa en TCP— . De este modo, la disciplina FCFS favorecería la agresividad de fuentes persistentes, dado que la persistencia se vería recompensada en términos de ancho de banda asignado.

En cuanto al soporte de aplicaciones con servicio garantizado, FCFS es incapaz de distinguir unas conexiones de otras. De este modo, no puede asignar a algunas conexiones retardos menores que a otras. Además, FCFS no puede ofrecer garantías de calidad de servicio a una conexión con independencia del comportamiento de las conexiones con las que se comparten los recursos.

La implementación de la disciplina FCFS es sencilla, dado que los paquetes son colocados al final de la cola a medida que llegan al sistema y son extraídos de la cabeza de la cola cuando ha de transmitirse el paquete siguiente.

3.3 Los algoritmos de planificación equitativa

A continuación se estudia una serie de disciplinas de servicio que se engloban bajo la denominación genérica de algoritmos de planificación equitativa (*fair queueing*). A diferencia de la disciplina FCFS, las disciplinas de planificación equitativa ordenan las peticiones

de servicio de un modo tal que la asignación de capacidad de servicio que resulta es equitativa en el sentido *max-min*.

La primera propuesta de algoritmos de planificación equitativa fue hecha por Nagle (1985), quien propuso que los *routers* IP sirviesen en un orden rotatorio los datagramas pertenecientes a cada flujo. Esta aproximación ignoraba el hecho de que el tamaño de los datagramas IP es variable, por lo que aquellos flujos cuyos datagramas fuesen mayores obtendrían una fracción mayor de ancho de banda. Este inconveniente fue resuelto por Demers, Keshav y Shenker (1989), quienes propusieron el algoritmo WFQ, que se describe en la sección 3.3.2. La disciplina GPS, que se presenta en la sección 3.3.1, es una idealización del algoritmo WFQ. Finalmente, Golestani (1994) propuso un algoritmo de planificación equitativa que simplifica la complejidad computacional del algoritmo WFQ, denominado SCFQ.

3.3.1 La disciplina *Generalised Processor Sharing*

La disciplina *Generalised Processor Sharing* (GPS) es el punto de partida de una clase de algoritmos de planificación de prioridad ordenada conocida con la denominación genérica de planificación equitativa. En GPS, como también en los algoritmos que presentaremos a continuación, las peticiones de servicio de cada usuario son distinguibles, por cuanto que, aunque el orden de servicio de las tareas de un mismo usuario respetan la secuencia de llegada al sistema, no ocurre así al respecto del orden de servicio de las tareas de distintos usuarios. Por tanto, en un sistema de planificación equitativa a cada usuario le corresponde una cola de tareas, ordenada internamente según FCFS, de modo que la tarea de cabeza de cola es la que primero recibe servicio entre las del mismo usuario.

Sea un sistema de espera descrito por las siguientes características (Greenberg y Madras, 1992):

- la capacidad de servicio es de r bits por unidad de tiempo;
- $N_{ac}(t)$ es el número de usuarios activos, esto es, aquellos usuarios, de los N existentes, cuyas colas de tareas no están vacías en el instante t; $0 \leq N_{ac}(t) \leq N$;
- sea τ_i^k el instante de tiempo en que la tarea k-ésima del usuario i llega al sistema.

se define la disciplina *head-of-line processor sharing* (PS) como aquella en la que:

> el servidor reparte uniformemente su capacidad de servicio entre aquellas tareas que ocupan la cabeza de la cola de cada uno de los usuarios activos; esto es, cada una de tales tareas recibe servicio a una tasa igual a $r/N_{ac}(t)$

Obsérvese que la disciplina PS es irrealizable, por cuanto que para realizarlo más de una tarea debería recibir servicio simultáneamente en el sistema. Además, un usuario activo será pues aquel usuario que genera peticiones de servicio a una tasa mayor o igual que la tasa a la que el servidor PS las sirve.

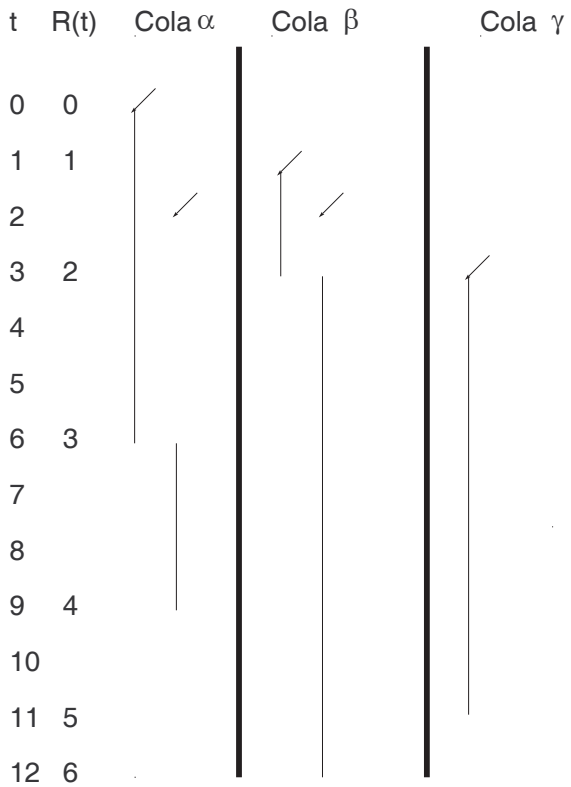

Figura 3.1. EJEMPLO DE EVOLUCIÓN DE UN SISTEMA PS CON 5 TAREAS PERTENECIENTES A 3 USUARIOS. LOS INSTANTES DE LLEGADA DE CADA TAREA SE INDICAN CON FLECHAS. CON TRAZO CONTINUO SE INDICA EL INICIO Y EL FINAL DEL SERVICIO.

Podemos asociar con un sistema PS un reloj de *tiempo virtual*, que progresa a la tasa del servicio que recibe una tarea de cabeza de cola. La relación entre tiempo virtual $R(t)$ y el tiempo real t es la siguiente:

$$\dot{R}(t) = \frac{d}{dt}R(t) = \frac{r}{\max\{1, N_{ac}(t)\}} \tag{3.1}$$

Dado que R es estrictamente creciente, existe su inversa, R^{-1}, que hace corresponder tiempo virtual a tiempo real.

En la figura 3.1 se muestra un ejemplo de evolución de un sistema PS, que describimos a continuación. En el ejemplo, cuando una tarea inicia servicio en el instante t_1 y finaliza en el instante t_2, se dice que ocupa el intervalo semi-abierto $[t_1, t_2)$. Denotamos como P_i^k la cantidad de servicio que la tarea k-ésima del usuario i requiere del sistema. El sistema del ejemplo sirve a $N = 3$ usuarios, denominados α, β and γ. Dos tareas del usuario α llegan al sistema en los instantes $\tau_\alpha^1 = 0$ y $\tau_\alpha^2 = 2$, dos tareas del usuario β en los

instantes $\tau_\beta^1 = 1$ y $\tau_\beta^2 = 2$, y una tarea de γ en el instante $\tau_\gamma^1 = 3$. Las cantidades de servicio requerido por cada tarea son, respectivamente, $P_\alpha^1 = 3$, $P_\alpha^2 = 1$, $P_\beta^1 = 1$, $P_\beta^2 = 4$, y $P_\gamma^1 = 3$. En este escenario durante el intervalo $[0, 1)$, la cola de α es la única cola activa, y por tanto recibirá una unidad de servicio, de modo que el tiempo virtual al final del intervalo será $R(1) = 1$, con lo cual la primera tarea de α necesita exactamente dos unidades más de servicio. Durante el intervalo $[1, 3)$, las colas de α y de β están activas, de modo que cada uno recibe servicio a una tasa $\dot{R} = 1/2$ y acumula una unidad de servicio, con lo cual al final del intervalo $R(3) = R(1) + 1/2 \cdot (3 - 1) = 2$. La primera tarea de β queda pues completamente servida, mientras que la primera tarea del usuario α aún necesita una unidad más de servicio. Mientras tanto, la segunda tarea de α ha llegado en $t = 2$, quedando en espera de servicio en la cola de α. Por su parte, la segunda tarea de β, que también ha llegado en $t = 2$, pasa a recibir servicio en $t = 3$, que es cuando la primera tarea ha finalizado su servicio. Durante el intervalo $[3, 9)$, las colas de los tres usuarios están activas, de modo que cada usuario recibe servicio a una tasa $\dot{R} = 1/3$ y acumula dos unidades de servicio, con lo cual al final del intervalo $R(9) = R(3) + 1/3 \cdot (9 - 3) = 4$. La segunda tarea del usuario α completa entonces sus necesidades de servicio, la segunda tarea de β necesita dos unidades de servicio más, mientras que la primera tarera de γ aún necesita una. Durante el intervalo $[9, 11)$, las colas de β y de γ están activas, de modo que cada uno acumula una unidad de servicio, con lo cual $R(11) = 5$. Queda entonces solamente la segunda tarea de β que necesita una unidad más de servicio, que le es proporcionada en el intervalo$[11, 12)$.

Podemos generalizar la definición de la disciplina PS, si asociamos con cada uno de los usuarios del sistema un número real positivo $\phi_1, \phi_2, \ldots, \phi_N$, de modo que la distribución de la capacidad de servicio sea proporcional al peso relativo del número real positivo asociado. Tal generalización recibe el nombre de *Generalised Processor Sharing*. Una definición formal de la disciplina GPS es la siguiente (Parekh, 1992):

> Sea un servidor conservativo con capacidad de servicio r. Sea $S_i(\tau, t)$ la cantidad de servicio que recibe el usuario i durante el intervalo $(\tau, t]$. Tal servidor se define como GPS si cumple
>
> $$\frac{S_i(\tau, t)}{S_j(\tau, t)} \geq \frac{\phi_i}{\phi_j}, j = 1, 2, \ldots, N \tag{3.2}$$
>
> para cualquier usuario i que esté activo durante todo el intervalo $(\tau, t]$

Obsérvese que un servidor GPS asegura que aquellos usuarios activos, que son los que reciben servicio a una tasa inferior a la que necesitan, comparten la capacidad de servicio excedente en proporción a sus pesos. Por definición, entonces, GPS consigue una asignación de la capacidad de servicio equitativa en el sentido *max-min* (véase la página 38).

Si sumamos 3.2 para todo usuario j, obtenemos

$$S_i(\tau, t) \sum_{j=1}^{N} \phi_j \geq (t - \tau) r \phi_i$$

de modo que a cada usuario i, GPS le garantiza una tasa de servicio g_i

$$g_i \triangleq \frac{S_i(\tau, t)}{t - \tau} = \frac{\phi_i}{\sum_j \phi_j} r$$

Por tanto, GPS consigue protección entre usuarios, en cuanto que permite garantizar individualmente a cada usuario una fracción de la capacidad de servicio.

Por último, en GPS la relación entre tiempo virtual $R(t)$ y el tiempo real t es la siguiente:

$$\dot{R}(t) = \frac{d}{dt} R(t) = \frac{r}{\max\left\{1, \sum_{i\text{activo}} \phi_j\right\}} \tag{3.3}$$

3.3.2 La disciplina *Weighted Fair Queueing*

Demers y otros (1989) introdujeron un nuevo algoritmo de planificación para la transmisión de paquetes, que denominó *Fair Queueing*. Tal algoritmo emulaba la disciplina PS, manejaba paquetes de tamaño variable y nunca desalojaba un paquete que estuviese siendo transmitido —como hace la disciplina PS—. Posteriormente, el algoritmo de Demers fue generalizado por Parekh (1992) (1993) (1994), quién lo denominó *Packet-by-packet GPS*, con el fin de emular la disciplina GPS. En esta memoria, emplearemos la denominación *Weighted Fair Queueing* (WFQ) para referirnos al algoritmo de Parekh.

Con el fin de describir cómo WFQ emula a GPS, introducimos las siguientes relaciones de recurrencia, que describen la evolución de un sistema GPS en tiempo virtual. Para cada $k = 1, 2, \ldots$ y $i = 1, 2, \ldots, N$, definimos S_i^k y F_i^k como los instantes de tiempo virtual en los cuales el k-ésimo paquete de la conexión i empieza a recibir servicio y acaba de recibir todo el servicio que requería, respectivamente. Se cumple que:

$$S_i^k = \max\left\{F_i^{k-1}, R(\tau_i^k)\right\} \tag{3.4}$$

$$F_i^k = S_i^k + P_i^k/\phi_i \tag{3.5}$$

donde $S_i^0 = F_i^0 = 0$.

La ecuación 3.4 establece que el paquete k-ésimo de la cola i inicia servicio, bien cuando llega al sistema —esto es, en el instante virtual $R(\tau_i^k)$—, si la cola i no contiene paquetes en ese instante, bien cuando el paquete precedente haya finalizado —esto es, en el instante virtual F_i^{k-1}—, en caso contrario. Por otro lado, la ecuación 3.5 refleja el hecho de que debe transcurrir ϕ_i unidades de tiempo virtual para que se sirva un bit de la conexión i. Nótese que, mediante 3.4 y 3.5, es posible calcular el instante virtual de finalización de servicio para cualquier paquete, si se conoce P_i^k en el instante en que llega al sistema. No obstante, no podemos calcular el instante de finalización de servicio, $R^{-1}(F_i^k)$, en el instante de llegada, pues el instante *real* de finalización depende de cuántos y qué paquetes lleguen con posterioridad.

Tomando como base las relaciones 3.4 y 3.5, podemos definir el algoritmo WFQ como sigue. Supongamos que un paquete acaba de completar su servicio en el instante t, momento en el cual existen n conexiones activas —esto es, n colas con paquetes en espera de servicio—, siendo $0 \leq n \leq N$. Entonces,

- si n $= 0$, entonces el primer paquete que llegue al sistema pasa a recibir servicio inmediatamente en el instante de llegada;
- si n $\neq 0$, de entre los paquetes en espera de servicio, se selecciona aquella que habría *finalizado* antes si el sistema se hubiese comportado idealmente como GPS; esto es, aquel paquete en espera de servicio con menor instante virtual de finalización F_i^k.

Para realizar una planificación utilizando el algoritmo WFQ, es necesario por tanto calcular los instantes virtuales de llegada al sistema y de finalización de servicio para cada paquete como si se ejecutase el algoritmo GPS. Para ello se debe mantener el valor actualizado de tiempo virtual del sistema, para lo cual es necesario determinar qué conexiones están activas en cada momento, es decir, qué colas se vaciarían y cuáles dejarían de estar vacías en cada instante, en un sistema GPS que recibiese la misma secuencia de llegada de paquetes que la que está recibiendo el sistema WFQ.

En la figura 3.2 se ha trazado la evolución de un sistema WFQ que lleve asociado un sistema GPS tal como el del ejemplo de la figura 3.1. Las líneas de trazo continuo indican los instantes de inicio y de finalización de las tareas si son servidas por el sistema GPS, mientras que las de trazo discontinuo, si son servidas por el sistema WFQ. En el instante $t = 0$, la única tarea presente en el sistema es la primera de α, por lo que pasa a recibir servicio inmediatamente y recibe en exclusividad las tres unidades de servicio durante $[0, 3)$. De entre las tres tareas en espera de recibir servicio en el instante $t = 3$ (segunda tarea de α, las dos tareas de β y la primera de γ), la primera tarea de β posee la marca temporal más pequeña de entre las tres, $F_\beta^1 = 2$, de modo que es la elegida para recibir servicio a continuación y recibe el servicio que requiere durante $[3, 4)$. Aplicando del mismo modo el criterio de elección anterior, la segunda tarea de α es servida hasta completar sus necesidades durante $[4, 5)$, la primera tarea de γ durante $[5, 8)$ y finalmente la segunda tarea de β durante $[8, 12)$.

GPS es una disciplina de servicio ideal, en cuanto que es irrealizable, pero también en cuanto que garantiza una asignación equitativa de recursos en el sentido *max-min*. En consecuencia, podemos cuantificar la equidad de cualquier algoritmo de planificación determinando su grado de aproximación a GPS. Para el caso de WFQ, podemos afirmar lo siguiente. Sea P^{max} el tamaño máximo de cualquier paquete en la red. Sean $G_i(\tau, t)$ y $S_i(\tau, t)$ las cantidades de servicio que recibe la conexión i durante el intervalo $(\tau, t]$, cuando se sirve según GPS y según WFQ, respectivamente. Por último, sea g_i la tasa de servicio garantizada para la conexión i. Puede demostrarse que WFQ se retrasa —en la provisión de servicio a las conexiones— respecto a GPS como mucho en P^{max}/g_i(Parekh, 1993), es decir,

$$S_i(\tau, t)/g_i \geq G_i(\tau, t)/g_i - P^{max}/g_i$$

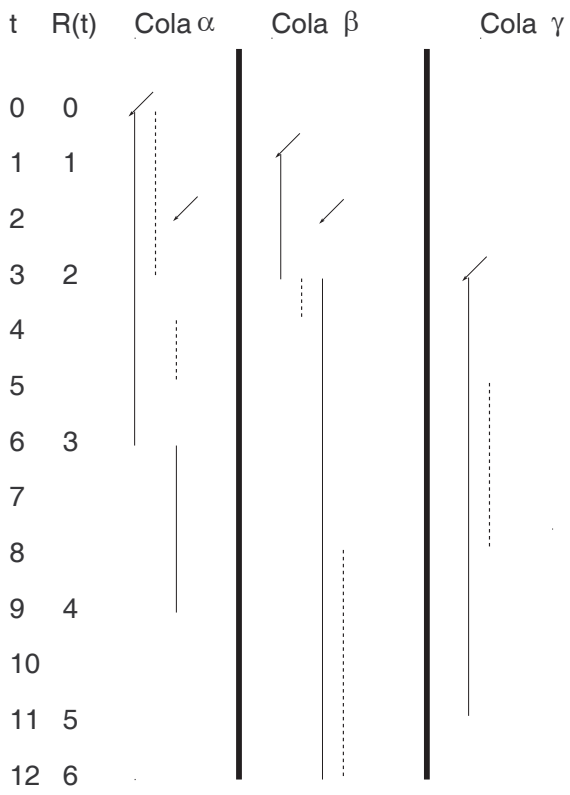

Figura 3.2. EJEMPLO DE EVOLUCIÓN DE UN SISTEMA WFQ CON 5 TAREAS PERTENECIENTES A 3 USUARIOS. EL SISTEMA GPS SIMULADO ES EL DE LA FIGURA 3.1. CON TRAZO CONTINUO SE INDICA EL INICIO Y EL FINAL DEL SERVICIO EN EL SISTEMA GPS, MIENTRAS QUE CON TRAZO DISCONTINUO, EN EL SISTEMA WFQ.

No obstante, bajo WFQ, una conexión puede recibir mucho *más* servicio que bajo GPS (Bennett y Zhang, 1996b), lo cual puede provocar inequidad a escalas temporales pequeñas. Este problema puede resolverse modificando la regla de elección en WFQ:

> En lugar de elegir para transmitir aquel paquete con menor instante virtual de finalización de entre todos los paquetes en espera de transmisión, se elige de entre aquellos paquetes en espera de transmisión que hubiesen empezado a recibir servicio —y posiblemente finalizado— en el sistema GPS simulado en tal instante

Esta variación ha sido propuesta por Bennett y Zhang (1996a) bajo la denominación *Worst-case-Fair WFQ* (WF^{2}Q).

En WFQ, la principal complicación a la hora de calcular el instante virtual de finalización para un paquete es determinar, en el instante de llegada, el tiempo virtual del sistema. Tal complicación se debe al problema de la eliminación iterativa (*iterated deletion*). Brevemente, cuando una conexión pasa a inactiva y, por tanto, es eliminada de la lista de conexiones activas, aumenta la probabilidad de que otras conexiones pasen a inactivas en breve. En otras palabras, en un sistema PS, para calcular el tiempo virtual del sistema, se debe llevar un registro del número de conexiones activas $N_{ac}(t)$, dado que, por definición, el tiempo virtual de un sistema PS aumenta a una tasa inversamente proporcional a $N_{ac}(t)$. Sin embargo, tal registro se complica por el hecho de que la decisión de si una conexión está activa o no depende a su vez del tiempo virtual del sistema PS.

Ilustraremos el problema de la eliminación iterativa mediante el siguiente ejemplo. Supongamos que la capacidad de transmisión es 1. En el instante 0 un paquete de 100 bits perteneciente a la conexión A llega al sistema. Durante $[0, 50)$, dado que $N_{ac} = 1$ y que $\partial R(t)/\partial t = 1/N_{ac}$, se tiene que $R(50) = 50$. En el instante 50, un paquete de 100 bits perteneciente a la conexión B llega al sistema. Su instante virtual de finalización será 150 $(= 50 + 100)$. Entonces, en el instante 100, P_A^0 ha completado su servicio. No obstante, durante $[50, 100)$, $N_{ac} = 2$ y, por tanto, $R(100) = 75$; dado que $F_A^0 = 100$, A está todavía activa y aún $N_{ac} = 2$. En el instante 200, P_B^0 ha completado su servicio; ¿cuál debería ser $R(200)$? Para obtener este valor, se debe notar que el número de conexiones activas debió haberse reducido cuando $R(t) = 100$. Esto ocurrió en $t = 150$, puesto que $R(100) = 75$ y, desde entonces, $\partial R(t)/\partial t = 1/2$. Por tanto, $R(200) = 100 + 1 \cdot (200 - 150) = 150$. Nótese cómo el paso de una conexión a inactiva acelera el progreso de $R(t)$, lo cual puede hacer más probable que otras conexiones hayan pasado a inactiva. Es, por ello, que se necesita realizar una eliminación iterativa de conexiones activas cada vez que se calcula $R(t)$.

El procedimiento de eliminación iterativa para el cálculo de $R(t)$ en el instante de llegada de un paquete, se muestra en la figura 3.3 (Keshav, 1991c)

Obsérvese que:

- el sistema mantiene dos variables estáticas: t_{chk} y $R_{chk} = R(t_{chk})$;
- en el instante t, el valor de $R(t)$ no puede ser menor que $R_{chk} + r/N_{ac}(t_{chk}) \cdot (t - t_{chk})$,

F, $\Delta y N$ son variables locales
$N =$ N_{ac} (t_{chk})
do:
$F =$ $\min(F_\alpha | \alpha$ está activa$)$;
$\Delta =$ $t - t_{chk}$
if $(F \leq R_{chk} + \Delta \cdot r/N)\{$
registra la conexión $F_\alpha = F$ como inactiva
$t_{chk} = t_{chk} + (F - R_{chk}) \cdot N/r$;
$R_{chk} = F$;
$N = N - 1$;
$\}$
else $\{$
$R(t) = R_{chk} + \Delta \cdot r/N$;
$R_{chk} = R(t)$;
$t_{chk} = t$;
$N_{ac}(t) = N$;
exit;
$\}$
od

Figura 3.3. ALGORITMO DE ELIMINACIÓN ITERATIVA EN WFQ

dado que N_{ac} es decreciente en $[t_{chk}, t]$, pues sólo se habrán producido salidas del sistema durante este intervalo;

- si F_α es menor que esa expresión, entonces podemos afirmar que la conexión α ha pasado a inactiva en algún instante anterior a t;
- determinamos el instante en que tuvo lugar el suceso, lo registramos como nuevo instante t_{chk} y actualizamos R_{chk};
- repetimos el procedimiento anterior hasta que no encontramos ninguna conexión inactiva más en el instante t, en cuyo caso podemos calcular $R(t)$.

Como vemos, durante la transmisión de un paquete en WFQ se pueden disparar hasta N sucesos en la simulación de la evolución del sistema GPS. Por tanto, la complejidad

de una decisión de planificación es $O(N)$, lo cual hace prohibitivo en muchos casos la implementación del algoritmo WFQ.

3.3.3 La disciplina *Self-Clocked Fair Queueing*

Según Golestani (1994), el origen de la complejidad del algoritmo WFQ radica en que decide el orden de servicio en referencia a los sucesos de un sistema ideal que simula, esto es, un sistema GPS. Esta aproximación, junto con el hecho de que en un sistema GPS más de un paquete reciben servicio simultáneamente, hacen complejo el algoritmo WFQ desde el punto de vista computacional. Como alternativa Golestani (1994) propuso el algoritmo *Self-Clocked Fair Queueing*(SCFQ).

El algoritmo SCFQ, al igual que WFQ, se basa en la noción de tiempo virtual del sistema como indicador del progreso de trabajo en el sistema. Sin embargo, a diferencia de WFQ, en el que el tiempo virtual del sistema hace referencia a un sistema simulado, en SCFQ, hace referencia a un sistema real. Además, en lugar de recurrir a una definición analítica y abstracta para el tiempo virtual, como ocurre en WFQ, en SCFQ se toma como tiempo virtual una cantidad intrínsecamente asociada con el progreso del algoritmo. Obsérvese que, según 3.5, en WFQ la marca temporal de un paquete es igual al instante de tiempo virtual cuando completa su servicio —en el sistema GPS simulado—. A partir de esta afirmación, Golestani (1994) intuyó que se puede estimar el tiempo virtual del sistema a partir de la marca temporal del paquete que está recibiendo servicio en el sistema real en cada instante t.

Se define el algoritmo SCFQ como sigue:

1. La marca temporal del paquete k-ésimo de la conexión i, que llega al sistema en el instante τ_i^k se computa a partir de las siguientes ecuaciones

$$\hat{S}_i^k = \max\left\{\hat{F}_i^{k-1}, \hat{R}(\tau_i^k)\right\} \tag{3.6}$$

$$\hat{F}_i^k = \hat{S}_i^k + P_i^k/\phi_i \tag{3.7}$$

   donde $\hat{S}_i^0 = \hat{F}_i^0 = 0$

2. El tiempo virtual del sistema SCFQ, $\hat{R}(t)$, en el instante τ_i^k se toma como $\hat{F}_j^l$, cuando el paquete l-ésimo de la conexión j es el que está recibiendo servicio en ese mismo instante.

3. Dado que no se simula ningún sistema, se dice que una conexión está activa o no en un sistema SCFQ en referencia exclusivamente al sistema real, esto es, está activa si existe algún paquete de la conexión en espera de servicio o siendo transmitido.

4. Cuando el sistema pasa a inactivo, lo cual ocurre cuando no hay ningún paquete en el sistema, el algoritmo se reinicializa, esto es, se pone a cero el valor del tiempo virtual del sistema y se pone a cero el contador k de paquetes de cada conexión i.

Presentamos a continuación una explicación intuitiva de cómo el algoritmo SCFQ descrito consigue equidad en el servicio. Un algoritmo más sencillo que SCFQ podía ser aquel en que las marcas temporales se calculasen como

$$\hat{F}_i^k = \hat{F}_i^{k-1} + P_i^k/\phi_i \tag{3.8}$$

donde $\hat{F}_i^0 = 0$;. En este algoritmo simplificado, cuando un paquete completa su transmisión, su marca temporal $\hat{F}_i^k$ es igual a la cantidad de servicio normalizado que ha recibido la conexión i hasta ese instante de tiempo. El algoritmo simplificado, dado que elige para transmitir aquel paquete con menor marca temporal de entre los que esperan servicio, intenta equilibrar la cantidad de servicio normalizado obtenida por cada conexión; no obtante, esta igualación se hace con independencia de si una conexión ha estado activa o inactiva con anterioridad. Por ejemplo, supongamos que en el instante t el paquete que recibe servicio tiene una marca temporal igual a F y que una conexión k pasa a estar activa por primera vez con la llegada de una ráfaga de paquetes. Según 3.8, las marcas temporales de los paquetes de la conexión k se computarán inicialmente a partir de 0. De este modo, mientras que estas marcas temporales no superen el valor F, los paquetes de la conexión k adelantarán a todos los paquetes de otras conexiones que hayan llegado al sistema con anterioridad.

Para resolver el problema planteado por el algoritmo simplificado de la ecuación 3.8, debemos añadir, en el cómputo de la marca temporal del primer paquete perteneciente a una conexión hasta entonces inactiva, aquella oportunidad de servicio normalizado que ha perdido mientras permanecía inactiva; la sustitución de $\hat{F}_i^{k-1}$ en 3.8 por $\max\left\{\hat{F}_i^{k-1}, \hat{R}(\tau_i^k)\right\}$ en 3.6, cumple este cometido, pues sustituye el valor $\hat{F}_i^{k-1}$ por la marca temporal del paquete en servicio, si éste es mayor.

La inequidad que presenta 3.8 fue puesta de manifiesto por Zhang (1991), quien sugirió entonces la sustitición de $\hat{F}_i^{k-1}$ en 3.8 por $\max\left\{\hat{F}_i^{k-1}, \tau_i^k\right\}$, para resolver el problema de la acumulación de oportunidades de servicio por parte de fuentes esporádicas. Esta propuesto no es adecuada, puesto que, al contrario que el tiempo virtual, el tiempo real τ_i^k no es una medida representativa del progreso del servicio en un sistema hasta el instante de llegada de un paquete.

A pesar de que la regla de actualización del tiempo virtual en el algoritmo SCFQ es sencilla desde el punto de vista computacional, el algoritmo SCFQ muestra inequidad en la distribución del ancho de banda a escalas temporales pequeñas. Veamos un ejemplo. Supongamos que se ha asignado un peso relativo de 50 a una conexión, mientras que otras 50 conexiones tienen un peso relativo de 1; además, por simplicidad, todos los paquetes son del mismo tamaño e igual a 1 bit y la capacidad del servidor es de 1 bit por unidad de tiempo. En el instante 0, llega un paquete de cada una de las 50 conexiones de peso 1; todos ellos obtendrán una marca temporal igual a 1, según 3.6 y 3.7. Justo después del instante 0, llega un paquete de la conexión de peso 50; dado que existe ya un paquete en servicio y que éste tiene una marca temporal de valor 1, el nuevo paquete obtendrá una

marca temporal igual a 1+1/50=1.02. Por tanto, el paquete será transmitido después de los 50 paquetes presentes en el sistema, es decir, en el instante 50, con lo que el retardo por espera que sufrirá será de 50 unidades de tiempo. Si el servidor hubiese sido WFQ, el paquete habría recibido una marca temporal igual a 0.02, con lo que habría sufrido un retardo por espera de 1 unidad de tiempo.

SCFQ provoca latencias en el peor caso mayores que WFQ, lo cual se traduce en una mayor inequidad a escalas temporales pequeñas. Más concretamente, si el tamaño máximo de paquete en la red es P_{max}, el número máximo de conexiones en el sistema es N y la capacidad de servicio es r, entonces la latencia en el pero caso en SCFQ es $P_{max}/g_i + (N-1)P_{max}/r$, mientras que en WFQ era $P_{max}/g_i + P_{max}/r$ (Keshav, 1997).

Golestani (1994) demostró que, dado un intervalo $[\tau, t)$ durante el cual dos conexiones i y j están activas, la diferencia entre la cantidad de servicio normalizado que recibe cada una de ellas durante tal intervalo es

$$\left| \frac{S_i(\tau,t)}{\phi_i} - \frac{S_j(\tau,t)}{\phi_j} \right| < \frac{P_i^{max}}{\phi_i} + \frac{P_j^{max}}{\phi_j} \tag{3.9}$$

siendo P_i^{max} y P_j^{max} los tamaños máximos de paquete para las conexiones i y j respectivamente.

La anterior expresión demuestra que la cantidad de servicio normalizado que recibe cada conexión a lo largo de cualquier intervalo de tiempo difiere entre conexiones un valor fijo máximo, suponiendo que las conexiones se mantengan activas durante el intervalo. A medida que la duración del intervalo $[\tau, t)$ aumenta, el impacto de esta diferencia sobre la tasa media de servicio desaparece. Consideremos una red ATM y comparemos dos conexiones con pesos relativos iguales a 1, se obtiene de 3.9 que

$$\left| S_i(\tau,t) - S_j(\tau,t) \right| < 2P$$

donde P es el tamaño de una célula ATM.

A la hora de implementar la disciplina SCFQ, Roberts (1994a) hizo notar que, a diferencia de lo que ocurre en WFQ, en SCFQ el cómputo de $\hat{F}_i^k$ puede demorarse hasta que el paquete alcance la cabeza de la cola de su conexión. Efectivamente, cuando el paquete k-ésimo de la conexión i llega a una cola llena, $\hat{F}_i^k = \hat{F}_i^{(k-1)} + P_i^k/\phi_i$. Sin embargo, cuando el paquete $(k-1)$-ésimo abandona la cabeza de la cola (sea τ tal instante), $\hat{R}(\tau) = \hat{F}_i^{(k-1)}$. De este modo, tanto si un paquete llega a la cabeza de la cola de su conexión tras esperar en la cola, como si lo hace inmediatamente a su llegada al sistema, podemos computar $\hat{F}_i^k$ como

$$\hat{F}_i^k = \hat{R}(\tilde{\tau_i^k}) + P_i^k/\phi_i$$

en donde $\tilde{\tau_i^k}$ es el instante de llegada de la célula k-ésima de la conexión i a la cabeza de la cola.

Si bien SCFQ simplifica el cálculo de la marca temporal, una implementación eficiente del algoritmo precisa, además, de un mecanismo ágil que mantenga la lista ordenada de las marcas temporales $\hat{F}_i^k$ de las células de cabeza de cola. Rexford y otros (1996) propusieron una arquitectura eficiente para ello, la cual empleaba un grupo de colas FIFO, en lugar de una lista ordenada. A continuación se describe esta propuesta.

En ATM, el valor P_i^k/ϕ_i es un valor constante para cada conexión, dado que P_i^k es igual para todas las células de todas las conexiones; entonces, $1/r_i = P_i^k/\phi_i$. Si denotamos F_{serv} como la marca temporal de la célula que está siendo transmitida, y F_i como la marca temporal de la célula de cabeza de la conexión i, se cumple que $F_i \leq F_{serv} + 1/r_i$. Por otro lado, dado que toda célula de cabeza de cola, por definición, no ha sido todavía escogida para ser transmitida, se cumple que $F_i \geq F_{serv}$, con lo cual

$$F_i \in [F_{serv}, F_{serv} + 1/r_{min}], \quad \forall i \in B(t)$$

donde $r_{min} = \min_\omega(r_\omega)$ y $B(t)$ es el conjunto de conexiones con células en sus colas en el instante t. El hecho de que el valor de F de las células de cabeza esté acotado nos va a permitir calcular las marcas virtuales empleando aritmética módulo cualquier base mayor que $1/r_{min}$.

Si, además, los valores $1/r_i$ son números enteros, también lo serán las marcas temporales F. No ocurre así en WFQ, donde $R(t)$ es una función real. Limitar los valores $1/r_i$ a que sean números enteros no supone ninguna restricción, dado que SCFQ asigna fracciones de ancho de banda a cada conexión según el cociente de los pesos ϕ_i, o lo que es igual, de los valores r_i.

Así pues, con las dos condiciones anteriores, en SCFQ la elección de qué célula transmitir se realiza de entre las células de cabeza de cola, cuyas marcas temporales toman uno de los $1 + 1/r_{min}$ valores diferentes. En otras palabras, las marcas temporales de las células de cabeza de cola en SCFQ pertenecen a un conjunto finito de valores enteros.

Con tales suposiciones, podemos implementar SCFQ empleando una cola FIFO por conexión más un conjunto reducido de colas FIFO ordenantes (*sorting bins*). Cada cola ordenante representa un valor distinto de F; en particular, la cola ordenante correspondiente a F_{serv} contiene las células en espera de transmisión inmediata. El servidor SCFQ visita cíclicamente las colas ordenantes y transmite todas las células de una cola ordenante antes de pasar a la siguiente cola ordenante no vacía. Por su parte, cada conexión sólo puede tener una célula en las colas ordenantes. Cuando la célula de cabeza de cola de la conexión i ha de ingresar en las colas ordenantes, se sitúa en la cola $1/r_i$ más allá de la cola ordenante que está recibiendo servicio. De este modo, se efectúa una planificación equitativa de forma implícita mediante la ubicación de las células de cabeza de cola en las colas ordenantes, sin necesidad de computar explícitamente la marca temporal.

Formalmente, tomamos n colas ordenantes, donde $n \geq 1 + 1/r_{min}$. Si la cola ordenante que está recibiendo servicio es la cola j, cuando una célula de cabeza de cola de la conexión i ha de ingresar en las colas ordenantes, se ubica en la cola con índice $l = j + 1/r_i$, donde la suma se realiza módulo n.

Vemos que, a diferencia de WFQ y del algoritmo SCFQ original, el algoritmo SCFQ simplificado propuesto por Rexford requiere sólo una operación de adición módulo $1 +$

$1/r_{min}$. Por otro lado, el rango y la granularidad en la asignación ponderada de ancho de banda depende del número de colas ordenantes; en concreto, si se dispone de b colas ordenantes, se puede servir conexiones con pesos $r_i \in \{\frac{1}{1}1, \frac{1}{2}, \ldots, \frac{1}{b-1}\}$.

3.3.4 La disciplina *Weighted Round-Robin*

El algoritmo *Weighted Round-Robin* (WRR), a diferencia de WFQ y de SCFQ, es un algoritmo de tipo enmarcado. En general, el algoritmo de planificación *round-robin* divide el eje temporal en periodos de trama de duración máxima y escoge un número determinado de paquetes de cada conexión según un turno rotatorio, de modo que el número total de paquetes es menor o igual al tamaño máximo de trama. A diferencia de la disciplina GPS, que sirve una cantidad infinitesimal de las tareas de cada usuario activo, el algoritmo *round-robin* escoge una tarea entera de cada usuario activo. Cuando se asigna distinto peso a cada conexión, el algoritmo *weighted round-robin* sirve un número de paquetes de cada conexión activa en proporción a su peso relativo.

El algoritmo WRR emula razonablemente bien a GPS cuando todas las conexiones tienen el mismo peso relativo y todos los paquetes tienen el mismo tamaño. Cuando los paquetes de las distintas conexiones tienen distino tamaño, el algoritmo WRR divide el tamaño medio de paquete de cada conexión entre el peso relativo de la conexión para así obtener un conjunto normalizado de pesos relativos. Así, para emular correctamente GPS en este caso, un sistema WRR debería conocer con antelación el tamaño medio de paquete de cada conexión, lo cual no es factible: en tal caso, el algoritmo WRR no puede asignar el ancho de banda equitativamente.

El algoritmo WRR sólo puede conseguir equidad a lo largo de periodos de tiempo mayores que el periodo de trama. A una escala temporal menor que el periodo de trama, algunas conexiones pueden obtener más servicio que otras. Concretamente, si una conexión tiene un peso relativo muy pequeño o bien el número de conexiones activas es grande, se pueden producir periodos prolongados de inequidad.

En redes ATM, las células tienen un tamaño fijo y pequeño, por lo que los periodos de trama son reducidos. Bajo tales condiciones, la emulación GPS que se consigue con el algoritmo WRR puede considerarse aceptable.

3.3.5 Realización de algoritmos de planificación equitativa

Algunas de las disciplinas de servicio anteriores han sido efectivamente implementadas en conmutadores ATM. Veremos a continuación algunos de los prototipos experimentales que han sido presentados en la bibliografía. No hemos encontrado ninguna referencia en la que se presente una implementación de la disciplina WFQ en prototipos que operen a las velocidades STM-1. La razón reside en la complejidad asociada a la simulación de un sistema GPS, como apuntabamos en la página 48.

Antes de pasar a describir los prototipos presentados en la bibliografía, veamos qué implementaciones existen en el mercado. Fore Systems, uno de los pioneros en la fabricación de conmutadores ATM ha sido también uno de los fabricantes que, desde el inicio de

la actividad del ATM Forum, ha apostado por la incorporación de algoritmos de planificación equitativa en los conmutadores. En particular, en el modelo *Forerunner* ASX-200BX, se proporciona la posibilidad de planificar la transmisión de las células mediante un algoritmo WRR.

Ya Katevenis y otros (1991) apuntaban la necesidad de que todo el procesado llevado a cabo en un conmutador ATM se realizase en *hardware*. Presentaba una arquitectura para un *chip* conmutador que realizaba la gestión de los *buffers*, la conmutación y el multiplexado mediante *hardware*. El *layout* se realizó mediante tecnología CMOS *full-custom* de 1 μm. En particular, la planificación de las células en cada puerto de salida se llevaba a cabo mediante un algoritmo WRR implementado en *hardware*. Para ello, mantenía la información de estado y el peso relativo de cada conexión en una memoria con organización similar a una memoria CAM. El estado de cada conexión contenía:

- un bit "ready", que se activaba cuando en el *buffer* había al menos una célula de la conexión correspondiente;
- un bit "still unvisited", que se activaba al inicio de cada ciclo de servicio y que durante el ciclo, después de que la conexión recibiera servicio, era desactivado;
- y durante qué ciclos de servicio la conexión tiene derecho a recibir servicio.

En cada ciclo de servicio, una célula como máximo puede ser transmitida de cada conexión. Katevenis expone una solución de implementación eficiente al problema de determinar, a partir del peso relativo de la conexión, cuáles son los ciclos de servicio en los que tiene derecho a recibir servicio.

Por su parte, Garrett (1996) construyó un prototipo experimental a partir del *output port controller* del *Sunshine Switch*, que fue un proyecto desarrollado en Bellcore (Giacopelli y otros, 1991). El OPC es capaz de implementar algoritmos FQ en tiempo real a 150 Mbit/s; dispone de dos entradas y una salida con interfaces STM-1; e incorpora un procesador i960 y un *buffer* de alta capacidad. Se codificó el algoritmo SCFQ en código C y se optimizó el código objeto resultante de la compilación.

Las funciones que debían ser ejecutadas en cada ciclo de recepción/transmisión son:

1. Calcular la marca temporal F_i^k para el paquete recibido.
2. Almacenar el paquete en la cola FIFO de la conexión i.
3. Detectar la llegada de un paquete a la cabeza de la cola de cada conexión e insertarlo en una lista ordenada por marcas temporales.
4. Transmitir el paquete con menor marca temporal en la lista ordenada.
5. Almacenar el valor mínimo de marca temporal, a emplear en 1.

La máxima complejidad temporal en el algoritmo descrito reside en el mantenimiento de la lista ordenada de las marcas temporales de los paquetes de cabeza de cola de cada conexión. Dado que su realización en *software* sería demasiado compleja en banda ancha

(p.ej., el tiempo de una célula ATM es de 3 μs a 150 Mbit/s), se escogió el chip VLSI diseñado por Chao (1991) (1992), que era capaz de ordenar hasta 256 elementos a una velocidad un orden de magnitud mayor de lo requerido. El prototipo realizado ofrecía un ancho de banda de 147 Mbit/s.

El proyecto Xunet 2 (Kalmanek y otros, 1997) incluía diez conmutadores ATM experimentales conectados con líneas de transmisión de 45 Mbit/s. Cada uno de estos conmutadores se ubicó en un *user site*, donde una estación de altas prestaciones funcionaba como *router* IP entre la red de área local FDDI y la red ATM . El conmutador Xunet se diseñó con arquitectura basada en bus y con colas por conexión en los puertos de salida. El gestor de colas de cada puerto constaba de:

1. Un *array* de DRAMs multipuerto con un ancho de banda de 1.3 Gbit/s. Este *array* contenía hasta 64K colas virtuales, implementadas mediante listas enlazadas de células.

2. Un planificador que soportaba los algoritmos WRR y FRR para cada nivel de prioridad. El algoritmo FRR es una versión simplificada de *Hierarchical Round- Robin* (Kalmanek y otros, 1990). Se distinguían 16 niveles de prioridad, en cada uno de los cuales el planificador mantenía una cola de control. Cada cola de control contenía una lista ordenada de los identificadores de las conexiones que disponen células que transmitir. En modo de funcionamiento WRR no ponderado, el planificador extrae un identificador de la cabeza de la cola de control, decide la transmisión de la célula de cabeza de la cola virtual identificada y, si esta cola virtual aún dispone de células que transmitir, coloca el identificador al final de la cola de control.

El *chipset* ATLANTA^{TM}, diseñado por Lucent Technologies (Chiussi y otros, 1997), comprende cuatro *chips* con los que pueden pueden conmutadores ATM con capacidad desde 622 Mbit/s hasta 25 Gbit/s. El dispositivo de conmutación básico, denominado *ATM Switch Module* (ASX), es un módulo conmutador a 5 Gbit/s que puede funcionar bien como conmutador 8×8 autónomo, bien como primera o tercera etapa en configuraciones multietapa. Dispone las colas en los puertos de salida, en cada uno de los cuales se provee una cola de servicio FIFO por cada uno de los cuatro niveles de prioridad contemplados. Todas las colas de un puerto comparten un único *pool* de capacidad igual a 512 células situado en el mismo *chip*. La planificación de las células de cada cola es de tipo WRR y se controla mediante una plantilla de 16 *slots*, lo cual proporciona una granularidad del 6%. Las oportunidades de transmisión no aprovechadas por un nivel de prioridad se ofrecen a las otras colas en orden estricto de prioridad. La estructura de planificación del dispositivo ASX se encuentra replicada en otro dispositivo, denominado *ATM Buffer Manager* (ABM), pero sólo mantiene en el mismo *chip* las células de cabeza de cada cola, mientras que el resto de células y de punteros se almacenan en una SRAM con capacidad de hasta 64K células.

3.4 La asignación del espacio de almacenamiento en los conmutadores

La gestión de los *buffers* es un aspecto importante del control de congestión para el soporte de servicio *best-effort*. Efectivamente, se necesita disponer de *buffers* para almacenar temporalmente los paquetes que llegan a un conmutador más rápidamente que los que son transmitidos por los puertos de salida. Si la sobrecarga se mantiene durante un periodo suficientemente largo, la capacidad de almacenamiento del conmutador se saturará, y habrá que descartar paquetes. El papel de la gestión de los *buffers* es decidir qué paquetes se descarta en una situación como la descrita.

En relación con la gestión de los *buffers*, el modelo de servicio *best-effort* sobre red ATM propuesto en la sección 2.4 establecía que se debe garantizar pérdidas escasas a aquellos usuarios que responden adecuadamente a la realimentación por congestión que genera la red. En particular, ello supone que cuando el ancho de banda disponible para un usuario disminuye, la red debe disponer de suficientes *buffers* para que los usuarios tengan su oportunidad de reducir la carga que ofrecen sin perder paquetes.

Los dos mecanismos de gestión de los *buffers* más utilizados son el mecanismo de compartición total y el de asignación por conexión. En el mecanismo de compartición total (*shared pool*), se utiliza un único *pool* de *buffers* para satisfacer las necesidades de almacenamiento de todas las conexiones. Los *buffers* se asignan a medida que se solicitan, es decir, de un modo *first-come first-use* (FCFU). La decisión de qué paquetes descartar se toma en función del grado de ocupación del *pool* único: cuando un paquete llega al conmutador y no quedan *buffers* disponibles, el paquete queda descartado.

Debido a su simplicidad, el mecanismo de compartición total es el más comúnmente implementado en la actualidad. Claramente, este esquema de gestión no ofrece protección entre conexiones: una única conexión podría ocupar todos los *buffers*, de modo que ninguna otra conexión podría recibir servicio.

El mecanismo de asignación por conexión (*per-flow allocation*) asigna los *buffers* individualizadamente. Se monitoriza la utilización que cada conexión está haciendo de los *buffers* y se descartan paquetes según el grado de ocupación relativo de *buffers* por parte de cada conexión. Este esquema de gestión protege a aquellas conexiones que se comportan correctamente en términos de utilización de *buffers*. La asignación de los *buffers* a las conexiones no sólo tiene en cuenta el número de conexiones activas —esto es, de conexiones que compiten por ocupar espacio de almacenamiento—, sino que también puede considerarse el ancho de banda asignado a cada conexión así como el número de *buffers* ya ocupados por cada una.

Cuando se emplea un algoritmo de planificación equitativa, puede aprovecharse la ordenación que genera el algoritmo para llevar a cabo una gestión de *buffers* del tipo asignación por conexión. En el caso de los algoritmos de prioridad ordenada, tales como WFQ y SCFQ, cuando el espacio de almacenamiento se ha saturado, se decide el descarte de aquel paquete con mayor marca temporal, esto es, aquel paquete que se serviría en último lugar si no llegase ningún otro paquete al sistema. Nótese que este algoritmo de descarte efectúa una distribución del espacio de almacenamiento análoga al que el algoritmo de

planificación hace del ancho de banda. Es por tanto una distribución equitativa *max-min*. En cuanto a los algoritmos enmarcados, tales como WRR, análogamente se decide el descarte de aquel paquete que se serviría en último lugar si no llegase ningún otro paquete al sistema. Este paquete es aquél paquete que ocupa el último lugar de aquella cola cuyo tamaño normalizado al peso de la conexión, es mayor.

La gestión de *buffers* es un elemento necesario en el control de congestión para el soporte de servicios *best-effort* y que no puede ser suplantado por un algoritmo de planificación; por ejemplo, un sistema WFQ puede garantizar una fracción equitativa del ancho de banda del enlace de salida, pero no de espacio de almacenamiento. Veamos una situación en la que se muestra la necesidad de la gestión de los *buffers* aun en presencia de un algoritmos de planificación equitativa. Sea un conmutador que sirve dos conexiones, A y B, cada una de las cuales llega por un enlace de entrada distinto; supongamos que ambas conexiones tienen el mismo peso relativo, por lo que su parte equitativa de ancho de banda es del 50 % . Aunque A transmita exactamente a su parte equitativa, B puede perfectamente estar emitiendo al doble de su parte equitativa, esto es, al total de la capacidad, por lo que, si no existe ningún mecanismo de gestión de *buffers*, B conseguiría copar todo el espacio de almacenamiento del conmutador. Además, si suponemos que una nueva conexión C pasa a activa en la situación descrita, la parte equitativa de ancho de banda de A se reducirá pero, dado que B ocupa todos los *buffers*, A no podrá detectar el cambio sin perder paquetes. Se incumpliría de este modo las garantías relativas y procedimentales que hemos enunciado como básicas en un servicio *best-effort*.

Si bien es un elemento necesario, la gestión de los *buffers* no es suficiente en el control de congestión y no puede suplantar al algoritmo de planificación. Efectivamente hace falta un algoritmo de planificación más potente que FCFS para conseguir una asignación de ancho de banda equitativa, pues con FCFS la fracción de ancho de banda asignada a cada conexión es proporcional al número de paquetes de la conexión en espera de servicio. Sólo podríamos conseguir una asignación equitativa de ancho de banda con FCFS y un mecanismo de gestión de *buffers* efectivo bajo las dos condiciones siguientes:

- que se asigne a cada usuario la misma fracción de espacio de almacenamiento que de ancho de banda; y
- que cada usuario mantenga el espacio de almacenamiento asignado completamente ocupado todo el tiempo.

Lamentablemente, mantener todos los *buffers* ocupados conduce a una tasa excesiva de paquetes perdidos.

3.5 Conclusiones

Los algoritmos de planificación y los mecanismos de gestión de *buffers* son mecanismos necesarios en la provisión de un servicio *best-effort*, en cuanto que se encargan de asignar los recursos a los usuarios del servicio *best-effort*. Se les exige a estos algoritmos que distribuyan los recursos de un modo equitativo, que protejan las asignaciones de cada usuario y que tengan una realización sencilla.

El algoritmo FCFS es sencillo pero no proporciona protección ni, por tanto, equidad en el reparto de los recursos. El algoritmo GPS es equitativo, pues distribuye el ancho de banda según el criterio *max-min* ponderado y, por tanto, proporciona protección; pero es un algoritmo no realizable. El algoritmo WFQ realiza una asignación equitativa muy próxima a la realizada por GPS, aunque presenta dificultades de implementación por su complejidad computacional. El algoritmo SCFQ es una simplificación heurística de WFQ, que tiene unas prestaciones ligeramente inferiores pero que es realizable. Finalmente, el algoritmo WRR es el más apliamente implantado en prototipos y en equipos ATM, pero sólo muestra unas prestaciones aceptables si el ciclo de servicio es corto.

La gestión de los *buffers* no puede sustituir a un algoritmo de planificación, ni puede ser sustituido por éste. Cada uno de ellos planifica la asignación de un recurso, pero no de los dos simultáneamente.

Capítulo 4

El control de flujo para la provisión de la clase de servicio ABR

En la clase de servicio ABR, se ha definido como marco de referencia para su provisión un esquema de control de flujo por realimentación de tasa. En este capítulo se estudia el papel del control de flujo en general, y en ABR en particular, en la provisión de un servicio *best-effort*.

En la sección 4.1, se analizan los conceptos de congestión, de control de congestión y de control de flujo. A continuación, se examina cada uno de los elementos que debe incorporar un mecanismo de control de flujo para la provisión de ABR: la generación de la señal de la realimentación (sección 4.4), el envío de la señal de realimentación (sección 4.2) y el ajuste de tasa en el terminal (sección 4.3). Los dos últimos elementos han sido normalizados por el ATM Forum en *ATM Forum Traffic Management 4.0*, mientras que el primero queda sujeto a la diferenciación de los fabricantes. Será en este último aspecto en donde se centra la contribución de la Tesis.

4.1 El control de flujo

Para la provisión de un servicio *best-effort*, como para la provisión de otros tipos de servicio sobre una red de paquetes, es necesaria la implementación de mecanismos de control de congestión. Así ocurre en las redes ATM. Vamos a abordar a continuación los conceptos de congestión y de control de congestión, para centrar el estudio en el control de flujo.

Sea una red que soporta un servicio *best-effort*. La red que da soporte al servicio asigna, por tanto, los recursos en el momento en que se solicitan. Supongamos que la tasa de llegada de los paquetes a un conmutador excede momentáneamente la capacidad de un enlace de salida del conmutador. Ello provoca que los paquetes queden almacenados en espera de transmisión, lo cual ocasiona retardos. Este retardo puede ocasionar eventualmente el vencimiento de temporizadores de retransmisión en las fuentes; las correspondientes retransmisiones aumentarán la carga en el conmutador. Esta realimentación positiva lleva

a un deterioro acelerado en el que las retransmisiones predominan en la carga ofrecida al conmutador, con lo cual el rendimiento efectivo disminuye rápidamente (Jacobson, 1988). Además, si existe control de flujo conmutador a conmutador, se dará la situación de que nuevos paquetes no puedan entrar en el conmutador, por lo que el conmutador precedente se verá obligado a retener estos paquetes. Esta retención puede llevar a una situación de bloqueo, en la cual se paralice el flujo de los paquetes en la red (Tanenbaum, 1989).

Nótese que los tres fenómenos tienen lugar sucesivamente. Primero, el retardo por espera de los paquetes en los conmutadores aumenta. Segundo, tienen lugar pérdidas de paquetes. Finalmente, en el estado de congestión, las retransmisiones predominan en el tráfico de la red, de modo que el rendimiento efectivo disminuye. Así pues, podemos definir congestión como (Keshav, 1991a): "Una red está congestionada si, a causa de la sobrecarga, se produce la condición X", donde X puede interpretarse como retardo excesivo por espera en las colas (Jain, 1986) (Ramakrishnan y Jain, 1990), pérdida de paquetes (Jacobson, 1988) o disminución del rendimiento de la red (Lemieux, 1981).

Así pues, la congestión es un fenómeno que aparece en condiciones de sobrecarga de la red. Por tanto, el éxito en el control de la congestión se basa en saber cuándo se sobrecarga la red y tomar entonces acciones de control; en otras palabras, la clave del control de congestión radica en determinar a qué escala temporal se produce la sobrecarga de la red y tomar consecuentemente medidas a esa escala temporal.

Apliquemos esta consideración a la carga media de un enlace punto a punto; nótese que la carga media sólo tiene sentido si al mismo tiempo especificamos el intervalo de tiempo a lo largo del cual se toma el promedio. Si la carga media es alta durante un intervalo pequeño de tiempo, entonces el mecanismo de control de congestión tendrá que resolver la distribución de los recursos durante un periodo igualmente pequeño de tiempo. Si, en cambio, la carga media es alta durante un intervalo mayor de tiempo, el mecanismo escogido para controlar la congestión deberá resolver la distribución de los recursos durante ese periodo más largo de tiempo, así como durante intervalos más pequeños. Veamos un ejemplo. Sea una conexión que atraviesa un enlace de capacidad unidad. La conexión es esporádica, en cuanto que generará una carga alta durante intervalos de duración de, digamos, 1 ms, aunque la carga media durante periodos de 1 hora es mucho menor que 1. En este caso, si la conexión es sensible al retardo, el mecanismo escogido de control de congestión debe tomar medidas para poder satisfacer estos requisitos a una escala temporal de 1 ms. A escalas mayores, dado que la demanda media de recursos es pequeña, no se plantea la necesidad de disponer de mecanismos de control de congestión. En cambio, si la conexión genera una carga elevada durante intervalos de duración de 1 hora así como durante intervalos de 1 ms, necesitaremos mecanismos que actúen a ambas escalas de tiempo. Por ejemplo, puede aplicarse un control de admisión que asegure la disponibilidad de recursos para las conexiones que se establecen; este mecanismo actuaría a la escala de 1 hora. Simultáneamente, se podría disponer de un algoritmo de planificación en los nodos que garantizara los requisitos temporales de la conexión; este segundo mecanismo estaría actuando a la escala de 1 ms.

Este ejemplo ilustra tres puntos. Primero, el control de la congestión debe actuar simultáneamente a diferentes escalas de tiempo. Segundo, los mecanismos que actúen a

diferentes escalas de tiempo deben cooperar. En el ejemplo anterior, los algoritmos de planificación no pueden ofrecer garantías temporales sin contar con un control de admisión; al mismo tiempo, el control de admisión, antes de decidir la admisión de una conexión en la red, debe conocer la naturaleza de la asignación de recursos que lleva a cabo el algoritmo de planificación. Tercero, la escala temporal a la que debe actuar un mecanismo de control de congestión es la misma a la que el usuario percibe variaciones en el estado de la red.

Abordamos a continuación cinco escalas temporales a las que es aplicable el control: varios meses, un día, la duración de una conexión, varios retardos de ida y vuelta y menos de un retardo de ida y vuelta. En cada una de ellas se cita los mecanismos apropiados para controlar la congestión y el fundamento teórico del diseño del mecanismo correpondiente.

Meses Algunos cambios en la red tienen lugar durante periodos de varios meses; por ejemplo, un aumento del número de sedes interconectadas o del tráfico cursado entre dos sedes remotas estrechamente vinculadas. Si tales cambios ocasionan una sobrecarga continuada en la red durante periodos largos de tiempo, entonces la única solución para mantener un servicio aceptable es aumentar la capacidad de los enlaces. Por muy potentes que sean los controles aplicados a escalas menores de tiempo, la red nunca será capaz de adaptarse al aumento habido en la carga.

A esta escala de tiempo, el control de congestión se formula como un problema de dimensionamiento de redes, el cual ha sido abordado en profundidad por el teletráfico.

Un día Las medidas del tráfico cursado por las redes muestran que la utilización de las redes está sometida a un comportamiento cíclico cuyo periodo es un día: las redes están poco cargadas durante la noche o incluso desocupadas, pero están altamente ocupadas en las horas de trabajo durante el día. Un mecanismo de control de congestión que actuase a esta escala temporal debería conseguir distribuir de forma más uniforme la carga de la red a lo largo de las 24 horas del día. Tal fin puede conseguirse mediante esquemas de tarificación en el uso de la red, semejantes a los utilizados por los operadores de telefonía básica, que estimulasen la transferencia a través de la red fuera del horario punta del día.

Conexión En las redes orientadas a conexión, el control de admisión efectúa en esencia el control de congestión a la escala temporal definida por la duración de la conexión: si la admisión de una nueva conexión puede degradar la calidad de servicio que están recibiendo otras conexiones ya establecidas en la red, entonces la nueva conexión no debe ser admitida.

Por otro lado, la elección de la ruta de una conexión en el establecimiento de la misma puede emplearse asimismo para controlar la congestión.

Varios retardos de ida y vuelta Un retardo de ida y vuelta es la constante fundamental de tiempo en el mecanismo de control de congestión conocido como control de flujo por realimentación. Se trata del tiempo mínimo que se necesita para que una estación pueda determinar el efecto del ritmo al que envía los datos a la red.

El fundamento teórico del diseño del control de flujo es la teoría de colas (Schwartz, 1994) y la teoría de control.

Menos de un retardo de ida y vuelta A una escala menor que un retardo de ida y vuelta, el control de congestión se lleva a cabo mediante la planificación de la asignación de los recursos en los nodos de la red. Ya hemos afirmado que la planificación en la asignación del ancho de banda y del espacio de almacenamiento determina el ancho de banda, el retardo y el *jitter* que recibe cada conexión. Además, para que el control de congestión sea efectivo a esta escala temporal, el algoritmo de planificación no debe modificar sus asignaciones en función de la carga en la red.

En el capítulo 3 se ha estudiado los algoritmos de planificación, que son mecanismos de control de congestión que operan a una escala temporal menor que un retardo de ida y vuelta. En esta sección nos centraremos en el mecanismo de control de flujo por realimentación, que es el mecanismo básico de control de congestión que se ha definido para el soporte del servicio ABR en redes ATM.

El control de flujo hace referencia al conjunto de técnicas que permiten a una fuente de datos adaptar su tasa de transmisión a la tasa de servicio disponible en cada instante en el receptor de los datos y en la red. El control de flujo es una variante del problema de control clásico (Keshav, 1997). Efectivamente, desde el punto de vista del control de flujo, una red puede interpretarse como un sistema que consta de estados, entradas y salidas. Los estados del sistema están constituidos por la longitud de las colas, el número de paquetes en transmisión por los enlaces y el contenido de las señales de realimentación generadas por la red. Las entradas al sistema son las tasas a las que las fuentes están enviando datos a la red. Y las salidas son las tasas de servicio del flujo de datos emitidos por cada fuente en los nodos. El propósito de un mecanismo de control de flujo es calcular las entradas apropiadas a tal sistema de modo que el estado del sistema se estabilice, a partir de la observación de las salidas.

Cada nodo sirve los paquetes que hay en sus *buffers* a una tasa de servicio. La fuente debe adaptar su tasa de envío de paquetes a la red de modo que los *buffers* nunca queden saturados ni infrautilizados. Por añadidura, la fuente sólo puede conocer la tasa de servicio del nodo en cada momento sino después de un tiempo y, además, el efecto de una eventual modificación de la tasa sobre el nodo se produce después de otro tiempo. La suma de ambos tiempos se denomina *retardo de ida y vuelta*. Como hemos afirmado anteriormente, el RTT es la constante de tiempo fundamental en los mecanismos de control de flujo por realimentación, pues es el tiempo mínimo que necesita una fuente para conocer el efecto de sus acciones de control, esto es, sobre las variaciones que ejerce sobre la tasa de envío de paquetes a la red.

La principal diferencia con el control clásico de sistemas es que la salida del sistema depende no sólo de la acción de una fuente determinada, sino también de la acción de toda otra fuente que comparte recursos de red con la primera fuente. Este acoplamiento entre el comportamiento de las fuentes hace del control de flujo un problema difícil y no tratable directamente con las técnicas de la teoría de control clásica.

Junto con los algoritmos de planificación para la transmisión de paquetes y de gestión del espacio de almacenamiento en los nodos, el control de flujo es un componente esencial para proveer un servicio *best-effort*. Si la planificación y la gestión de los *buffers* permiten a la red asignar de forma dinámica los recursos disponibles de la red, el control de flujo por realimentación es el mecanismo por el cual la red informa a los usuarios del estado de la red, esto es, del estado de ocupación de los recursos, y por el que los usuarios adaptan su comportamiento en función de él, cerrando así el bucle de control. Veamos cuál es la relevancia de cada uno de estos dos aspectos del control de flujo.

La señal de realimentación que envía la red a los usuarios les informa del estado de la red y les permite calcular las respectivas partes equitativas de los recursos. Si la red proporcionase una señal de realimentación inconsistente o no equitativa a los usuarios, aparecerían diversas patologías en la red. El envío de la señal de realimentación es, pues, un componente necesario para el soporte de un servicio *best-effort*. No obstante, la señal de realimentación no puede por sí misma hacerse cumplir: son la planificación y la gestión de *buffers* los mecanismos que deben efectuar esta función.

Por su parte, el ajuste del usuario como respuesta a la señal de realimentación cierra el bucle de control. Es necesario en el diseño de la red, pues ajustes poco eficaces pueden ocasionar tasas altas de pérdidas o utilizaciones bajas de recursos. Sin embargo, el ajuste del usuario no es eficaz si la calidad de la señal de realimentación de la red es baja. Por otro lado, del mismo modo que la red no puede confiar en que los usuarios obedezcan la señal de realimentación, tampoco puede confiar en que los usuarios ejerzan un mecanismo concreto de ajuste, de modo que la integridad de la red deberá depender de los mecanismos internos de la red, es decir, de la planificación y de la gestión de los *buffers*.

Un mecanismo de control de flujo debe cumplir las siguientes propiedades (Keshav, 1991a) (Shenker, 1990):

Eficiencia Son dos los aspectos relacionados con la eficiencia en un mecanismos de control de flujo. En primer lugar, un mecanismo es más eficiente cuanto menos recursos de red consuma en su operación. Como caso extremo de mecanismo ineficiente, en TCP/IP los mensajes ICMP de tipo "Source Quench" pueden sobrecargar la red ya congestionada. En segundo lugar, un mecanismo que solicita a los usuarios una reducción de la tasa de envío de paquetes a la red cuando no existe peligro de congestión conduce a una utilización ineficiente de los recursos de red que se asignan a los usuarios.

Heterogeneidad A medida que las redes aumentan las distancias que cubren y las velocidades de transmisión que soportan, aparecen gran variedad tecnológica de plataformas de red. Un mecanismo de control debe ser operativo en un entorno de red diverso; en particular, no debe asumir un tamaño determinado de paquete, ni un protocolo único de transporte, ni siquiera un tipo único de servicio de red.

Resistencia En las redes públicas, el operador de la red no ejerce control administrativo sobre las estaciones, por lo que los usuarios son libres para manipular sus protocolos de red con el fin de maximizar la utilidad que obtienen de la red. De este modo, cuando la

red pide a un usuario que reduzca su tasa de envío de paquetes a la red, el usuario puede decidir no obedecer y así conseguir enviar más datos. Un mecanismo de control debe evitar que este tipo de comportamientos no cooperativo afecte a la calidad de servicio que reciben aquellos usuarios que sí se comportan correctamente. Alternativamente, puede disuadir a aquellos usuarios no cooperativos o incentivar un comportamiento correcto por parte de cualquier usuario.

Estabilidad En el control de flujo, existen dos fuentes principales de inestabilidad en la red, ante las que un mecanismo de control de flujo debe ser robusto. En primer lugar, el hecho de que existe un retardo no despreciable desde que se detecta la congestión hasta que el usuario recibe la señal de realimentación. En segundo lugar, la aparición de transitorios en el comportamiento de la red produce errores en la estimación que del estado de la red hacen los usuarios a partir de la señal de realimentación.

Así pues, es deseable que un mecanismo de control de flujo sea estable. Por estabilidad, idealmente entendemos que bajo cualesquier condiciones iniciales el vector de tasas asignadas ha de converger a un valor estacionario. Esta condición es extremadamente difícil de verificar, por lo que se suele considerar la condición más débil de estabilidad *lineal*. El mecanismo de control de flujo será linealmente estable si ante cualquier desviación que se produzca en el régimen estacionario, el mecanismo hace evolucionar al sistema de nuevo al régimen estacionario.

Escalabilidad Es deseable que un mecanismo de control sea operativo independientemente de la escala de la red en dos de sus ejes: el ancho de banda y el tamaño de la red. Sea $r_{SS}(\mu)$ el vector de tasas en régimen estacionario cuando el vector de capacidades de los enlaces de la red es μ: para cualquier constante positiva c, el mecanismo de control de flujo debe conseguir que $r_{SS}(c \cdot \mu) = c \cdot r_{SS}(\mu)$. Por otro lado, el tamaño de la red se mide en términos de número de nodos conectados y de distancias geográficas cubiertas. A este respecto, el valor del vector de tasas en régimen estacionario debe ser independiente de los retardos de realimentación presentes en el sistema.

Simplicidad La simplicidad de un mecanismo hace más sencilla su implementación, garantiza su operatividad incluso a velocidades superiores que las que se emplearon para su diseño y facilita su adopción como norma.

Equidad Un mecanismo de control nunca debe discriminar de forma arbitraria a ningún usuario. Más específicamente, el vector de tasas asignadas en régimen estacionario debe cumplir un criterio de equidad, según se abordó en la sección 2.3.3.

A la hora de diseñar un mecanismo de control de flujo, existen dos alternativas en la atribución de la responsabilidad del control de congestión: hacer responsable a las estaciones, con lo cual éstas deberán detectar la congestión en la red y evitarla; o bien, hacer responsable a los conmutadores, en cuyo caso se les exige que establezcan los mecanismos necesarios para que los usuarios cumplan las eventuales reducciones de ancho de banda

cuando detecte congestión, así como que asignen los recursos de la red con el objeto de evitar la congestión.

Creemos que la red debe hacerse responsable del control de la congestión, por dos razones. En primer lugar, evitando hacer recaer la responsabilidad sobre los usuarios de la red, quienes no están bajo el control administrativo del operador de la red, eliminamos riesgos sobre la operación de la red. En segundo lugar, de este modo la funcionalidad asociada al usuario de la red se simplifica, pues sólo se necesita incorporar la funcionalidad asociada a la detección de la congestión en los nodos, cuyo número es reducido, y no en las estaciones. Nótese que tener la responsabilidad en el control de la congestión no implica que los conmutadores deban llevar a cabo todas las acciones involucradas en el control de la congestión. Responsabilidad no es igual a funcionalidad.

En las secciones siguientes, abordamos cada uno de los mecanismos involucrados en el control de flujo:

1. La generación de la señal de realimentación.
2. El envío de la señal de realimentación.
3. El ajuste en los terminales.

De los tres mecanismos anteriores, la especificación del servicio ABR en *ATM Forum Traffic Management 4.0* (The ATM Forum Committee, 1996) normaliza los dos últimos. El primero queda, por el contrario, sujeto a la especialización del fabricante, si bien se especifican las directrices genéricas de su comportamiento en el bucle de control.

4.2 Envío de la señal de realimentación

La señal de realimentación desde la red hacia el usuario de la red proporciona a este último la información necesaria para ajustarse a la variación en la ocupación de los recursos de la red.

Las redes ATM son redes heterogéneas en cuanto al diseño de los conmutadores y a su fabricación, por lo que es muy importante desde el punto de vista de la arquitectura, separar dos aspectos de la realimentación en el control de flujo:

- El formato de la señal, que es la interfaz definida para comunicar la señal de realimentación, esto es, cómo se envía la señal de realimentación.
- El valor e instante de envío, que es la implementación de la señal de realimentación, es decir, los algoritmos específicos que calculan la señal de realimentación (este aspecto se tratará en la sección 4.4).

En función de la naturaleza de la señal de realimentación, distinguimos dos tipos de control de flujo : control de flujo por realimentación explícita y control de flujo por realimentación implícita.

En la realimentación explícita, el control de flujo emplea un campo del paquete del flujo asociado a la conexión como indicador explícito que comunica el estado de la red al

usuario. La realimentación explícita puede efectuarse en el sentido del flujo o en sentido contrario:

- En la realimentación hacia delante, el indicador se marca cuando el flujo de paquetes progresa desde el origen hacia el destino; este último es responsable de devolver la señal de realimentación al usuario origen.

- En la realimentación hacia atrás, los conmutadores envían directamente la señal de realimentación al usuario origen del flujo de paquetes, generando así un flujo de paquetes en el sentido contrario.

El mensaje ICMP "Source Quench" en IP es un ejemplo de realimentación hacia atrás (Tanenbaum, 1989). *Frame relay* contempla tanto realimentación hacia delante, denominada *Forward Explicit Congestion Notification* (FECN), como hacia atrás, *Backward Explicit Congestion Notification* (BECN) (Stallings, 1995).

Las señales explícitas de realimentación pueden clasificarse también en función de la cantidad de información que contengan. En las señales *binarias*, se notifica únicamente la presencia o ausencia de congestión en la red, con lo que el usuario debe intentar estimar cuantitativamente la gravedad de la notificación. Ante la incertidumbre asociada a la estimación, el usuario suele adoptar un comportamiento asimétrico de respuesta: cuando se notifica la aparición de la congestión, la tasa de envío a la red se reduce rápidamente a un valor estimado como seguro; cuando ya no se recibe notificación de congestión, se va aumentando lentamente la tasa de envío. El algoritmo Ramakrishnan-Jain de los protocolos DNA es un ejemplo de realimentación binaria hacia delante (Ramakrishnan y Jain, 1990). Por otro lado, cuando el control de flujo emplea señales de *valor explícito*, se notifica el grado de congestión que experimenta la red, en términos de tasa equitativa calculada, de espacio de almacenamiento disponible, o de ambos

La realimentación explícita, tanto binaria como de valor explícito, requiere que los conmutadores incorporen los mecanismos de generación de la señal de realimentación y de inyección en el flujo de paquetes de la conexión, lo cual aumenta la complejidad del mecanismo de realimentación. Por contra, la información de control que se proporciona es cuantitativa, no cualitativa, lo cual acelera el procedimiento de adaptación frente a cambios en la red.

La realimentación implícita, al contrario, precisa que el usuario monitorice las prestaciones que obtiene de la red en la transferencia de sus datos con el fin de deducir el estado de la red.

El ejemplo más conocido de realimentación implícita es el del algoritmo *slow-start* (Jacobson, 1988) utilizado en TCP, que hace uso de la pérdida de paquetes como señal implícita de congestión. Sin embargo, el descarte de un paquete no es necesariamente una señal de agotamiento de los *buffers* en los conmutadores, como así ocurre en FCFS. Algunos algoritmos de descarte, tal como *Random Early Discarding*(RED) descartan los paquetes para advertir de la inminencia de la congestión (Floyd y Jacobson, 1993). A no ser que los usuarios conozcan cuál es el algoritmo de descarte de paquetes utilizado en los conmutadores, no podrán distinguir entre la señal de inminencia de congestión de un conmutador

RED y la señal de agotamiento de *buffers* de un conmutador FCFS y, por tanto, no podrá decidir la acción más apropiada de control como respuesta.

Otro ejemplo de realimentación implícita es el del protocolo *Packet-Pair* (PP) (Keshav, 1991b), que toma como señal de realimentación implícita la tasa observada a la que los paquetes de una determinada conexión abandonan la red, lo cual asume que la tasa de servicio del conmutador de cuello de botella de esa conexión queda reflejada en la tasa del flujo en la salida de la red. Esta señal de realimentación contiene mucho ruido, por lo que requiere que se promedie durante un periodo de tiempo. La duración de los intervalos de promediado es un aspecto clave en las prestaciones de este tipo de algoritmos; en particular, si el intervalo es más largo que el periodo de refresco de la señal de realimentación, aparecerán problemas debidos a retardos en el control. Además, los algoritmos de planificación en los nodos condicionan de forma importante la duración de estos intervalos de promediado.

Por su parte, el protocolo NETBLT (Clark y otros, 1987) compara el *throughput* observado para el flujo de paquetes de la conexión con la tasa de envío a la red de los mismos y así determina cuál es el *throughput* máximo que puede conseguir.

Un último ejemplo de realimentación implícita es la medida del cambio en el retardo de extremo a extremo según se varía la tasa de envío a la red. Este mecanismo se emplea en TCP Vegas (Brakmo y Peterson, 1995) y en el control *delay-based* (Jain, 1986). Las prestaciones de estos mecanismos están influidos por el algoritmo de planificación implantado. Si los conmutadores implementan algoritmos de planificación equitativa, el retardo observado será relativamente constante hasta que se alcance la tasa equitativa asignada por el conmutador, momento en el cual se producirá almacenamiento de paquetes en espera de servicio y, por tanto, se observará un aumento brusco del retardo sin ningún aumento en el *throughput*. En conmutadores que implementen algoritmos FCFS, esta técnica no funciona de manera óptima, pues siempre es observable un aumento en el *throughput* al aumentar la tasa de envío a la red, a expensas por supuesto de un aumento en la ocupación de las colas y del retardo extremo a extremo para todas las conexiones.

La principal ventaja de la realimentación implícita es que la red sólo debe ocuparse de la asignación de los recursos y no del cálculo de una señal de realimentación exacta. Ya hemos visto que el usuario puede derivar implícitamente el estado de la red a partir del *throughput*, del retardo extremo a extremo y de la pérdida de paquetes. No obstante, es discutible si, en cualquier situación, puede derivarse información correcta y precisa a partir de las prestaciones observadas de la red. Por ejemplo, en el caso de *flow segregation*, la señal de realimentación puede ser engañosa: por tanto, para que la realimentación implícita sea efectiva, todos los nodos deben ser coherentes a la hora de asignar los recursos de red.

4.2.1 Formato de la señal de realimentación en ABR

La especificación del mecanismo de control de flujo para soportar la categoría de servicio ABR hecha por el ATM Forum ha escogido un esquema de realimentación explícita. A continuación describimos el formato de la célula ATM de gestión de recursos (*Resource*

Management, RM), que sirve de indicador explícito de realimentación para cada conexión. Veremos este formato de señal de realimentación permite tanto la realimentación hacia delante y hacia atrás, como la realimentación binaria y de valor explícito.

Cada célula RM tiene una cabecera de célula ATM con los siguientes valores de campo:

1. el campo *Payload Type Indicator* (PTI) que toma el valor 110_2 e identifica a la célula ATM como de tipo RM;
2. el campo de identificador de protocolo, de ocho bits de longitud, que toma el valor 1 e identifica a las conexiones ABR.

Además, dispone de los siguientes campos específicos:

1. el bit de *direction* (DIR) distingue entre células RM hacia delante (*forward RM*, FRM), es decir, que fluyen en el mismo sentido que las células de datos de la conexión a la que están asociadas (DIR=0), y células RM hacia atrás (*backward RM*, BRM);
2. el bit de *Backward Notification* (BN), que se pone a 1 cuando la célula RM la genera un conmutador;
3. el bit de *Congestion Indication* (CI), que se emplea para notificar congestión extrema;
4. el bit de *No Increase* (NI), que se emplea para notificar congestión moderada;
5. el campo *Explicit Rate* (ER), que indica el valor máximo de tasa de envío permitido al usuario origen;
6. los campos *request/acknowledge*, *queue length* y *sequence number*, que son contemplados únicamente por el UIT-T en I.371;
7. el campo *Current Cell Rate* (CCR), que sirve para que el usuario indique a la red cuál es la tasa actual de emisión de células; esta indicación puede ser utilizada por los conmutadores para generar la correspondiente señal de realimentación, o bien puede el conmutador tomar sus propias medidas;
8. el campo *Minimum Cell Rate* (MCR), que, al igual que otros parámetros, tales como PCR e ICR, se fijan en el momento de establecer cada conexión ABR; no obstante, su presencia permite reducir el número de consultas a tablas necesarias en los conmutadores.

Todas las tasas, por ejemplo, ER, CCR y MCR, de las células RM se representan empleando un formato de coma flotante particular de 16 bits, que permite un valor máximo de 4290772992 cell/s (1.8 Tbit/s). Durante el establecimiento de la conexión, no obstante, los parámetros de tasa se negocian utilizando un formato de coma fija de 24 bits, que limita el valor máximo a 16777215 cell/s ó 7.1 Gbit/s.

Obsérvese que:

- la realimentación hacia delante se consigue con células RM con BN=0, generadas por la fuente (DIR=0) y devueltas por el destino (DIR=1);

- la realimentación hacia atrás se consigue generando células RM en los nodos con DIR=1 y BN=1;

- los bits CI y NI constituyen la señal binaria de realimentación;

- alternativamente, el campo ER sirve de señal de realimentación de valor explícito.

Estos cuatro mecanismos de realimentación pueden operar simultáneamente en la red, lo cual permite la presencia de conmutadores de distinto grado de complejidad y favorece la existencia de plataformas multi-proveedor de red. Por otro lado, esta variedad obliga a diseñar un algoritmo de ajuste en la fuente capaz de responder adecuadamente a señales de realimentación de naturaleza diversa.

El formato de célula RM especificado por *ATM Forum Traffic Management 4.0* sirve de soporte a un control de flujo por realimentación de tasa de emisión. En otras palabras, la señal de realimentación en ABR es básicamente la tasa de emisión permitida a la fuente del flujo de células de la conexión. No se tiene en cuenta el grado de ocupación de los *buffers* en los conmutadores salvo de forma indirecta a través del bit CI.

Lyles (1994a) ha puesto de relieve la necesidad de proveer un mecanismo de notificación explícita del tamaño de las colas en los nodos, como así ha sido aceptado por la Comisión de Estudio del UIT-T. Intuitivamente, el argumento puede ejemplificarse como sigue. Supongamos que el ancho de banda disponible en la red cambia de X a $X - \delta$. Si el usuario sólo se adapta al valor de ancho de banda disponible en cada momento, tras el transitorio quedarán permanentemente ocupados un número de *buffers* igual al producto δ por el tiempo de respuesta del bucle de control, que en el mejor de los casos será igual al retardo de ida y vuelta entre la fuente y el cuello de botella. En realidad, la tasa de emisión debería reducirse por debajo de $X - \delta$ con el fin de que se vacíen los *buffers*.

Altman y otros (1993) han demostrado que para un mecanismo genérico de control de flujo por realimentación que module la tasa de emisión en función de la señal de realimentación obtenida desde la red, implícita o explícitamente, se consigue estabilidad y eficiencia sólo si las acciones de control que toma la fuente están basadas en la información que obtiene no sólo de la estimación de la tasa de servicio en el cuello de botella, sino también de la estimación del tamaño de la cola en el cuello de botella. Por inestabilidad, se entendía que para cualquier tamaño inicial de cola y para cualquier constante finita C, $P[q_n < C]$ tiende a 0 a medida que n tiende a infinito; en otras palabras, el tamaño de la cola tiende a infinito en términos probabilísticos.

Nótese que la realimentación del grado de ocupación de los *buffers* no equivale a considerar un control de flujo basado en créditos. En la especificación *ATM Forum Traffic Management 4.0*, el bit CI es el único mecanismo utilizable para la realimentación del estado de ocupación de los *buffers* en la red. Sin embargo, se trata de una realimentación binaria y, al igual que se mencionó en la realimentación binaria de tasa, impide un ajuste rápido y eficiente en la fuente. Alternativamente, el vaciado de las colas en los conmutadores puede resolverse fijando un valor objetivo de utilización de ancho de banda menor del 100%, a tener en cuenta en la generación de la señal de realimentación.

4.3 Ajuste de tasa en los terminales

El ajuste en los terminales, o sistemas finales, constituye la respuesta del controlador en el bucle de control que define todo mecanismo de control de flujo. En sistemas de control por ajuste de tasa, tal como el contemplado en la especificación *ATM Forum Traffic Management 4.0*, el bucle cumple dos funciones:

1. igualar la tasa de emisión de la fuente a la tasa a la que el cuello de botella en la red puede servir el flujo de paquetes;

2. en situaciones de sobrecarga, debe permitir que los paquetes acumulados en el cuello de botella en la red puedan ser evacuados.

La eficacia del algoritmo de ajuste depende fuertemente del tipo de realimentación recibida de la red. Si ésta no es completa o es imprecisa, puede ser incluso necesario realizar una búsqueda del punto óptimo de operación; por ejemplo, en el caso de realimentación binaria el punto óptimo de operación sólo se encuentra tras un proceso iterativo de realimentación y ajuste.

Por otro lado, la precisión del ajuste determina, a su vez, las prestaciones de la red. En particular, si se permite que los *buffers* eventualmente se vacíen, nunca podrá obtenerse el *throughput* máximo posible. Por ello, manteniendo paquetes en espera de servicio en las colas de los conmutadores, se conseguirá un *throughput* alto puesto que la línea nunca quedará libre, ni siquiera ante aumentos de ancho de banda disponible y en presencia de retardos de realimentación. Por otro lado, si las colas se llenan demasiado, los paquetes serán descartados por los algoritmos de gestión de *buffers*.

Finalmente, como se abordó en la sección 2.2, la red no debe confiar en que todos los usuarios implementen de forma uniforme un algoritmo de ajuste. Además, la heterogeneidad puede surgir a consecuencia de que algunos usuarios particulares hayan preferido reaccionar de modo diferente ante la misma señal de realimentación, porque las aplicaciones que soportan simplemente tengan distintas tolerancias al retardo y a las pérdidas. Por ejemplo, si una aplicación está diseñada para tolerar grandes pérdidas, tal como el protocolo de transferencia de imágenes de Turner y Peterson (1992), es improbable que reduzcan sus tasa de forma tan rápida o agresiva como si se tratase de una aplicación sensible a pérdidas. Así pues, no existe una única respuesta esperable o aceptable por parte de los sistemas finales, por lo que la red no debe esperar un comportamiento determinado por parte del usuario. Al contrario, es la red quien debe presentar un comportamiento ante el cual el usuario pueda responder confiadamente pues, de otro modo, la respuesta de la red ante el ajuste del sistema final no se podrá predecir y el usuario no podrá contar con el ajuste para controlar la calidad de servicio que le ofrece la red.

Aunque existen diversas soluciones al tipo de procesado que realizan los sistemas finales fuente y destino sobre la señal de realimentación, el comportamiento más común es el de realimentación hacia adelante, en el cual el destino procesa la señal de realimentación procedente de la red y genera un mensaje de realimentación que envía a la fuente. Por ejemplo, el destino podría monitorizar el bit EFCI de los paquetes que recibe y calcular la señal de realimentación hacia fuente en función del porcentaje de paquetes recibidos con

bit EFCI a 1. Presentamos a continuación una clasificación de esquemas de realimentación atendiendo al tipo de ajuste realizado por la fuente cuando recibe —o deja de recibir— señal de realimentación desde el destino. Distinguiremos entre realimentación negativa, realimentación positiva y realimentación bipolar.

En la realimentación negativa, la fuente como comportamiento por defecto aumenta su tasa, esto es, en ausencia de realimentación, y reduce su tasa únicamente cuando el destino le notifica la aparición de congestión en la red. La realimentación negativa está considerada en ocasiones una estrategia optimista pues las fuentes intentan continuamente conseguir un ancho de banda mayor mientras no reciban realimentación. Es más, si la señal de realimentación se pierde debido a la congestión, esta estrategia agravará de hecho la situación. Por esta razón, no es deseable una tasa de aumento de tasa alta; por otro lado, la tasa de crecimiento de la tasa debe ser suficientemente alta para permitir a una conexión recién establecida obtener en un tiempo breve su fracción correspondiente de ancho de banda.

Por su parte, en la realimentación positiva la fuente reduce su tasa por defecto y la aumenta únicamente cuando el destino le notifica la ausencia de congestión en la red. La realimentación positiva es una estrategia pesimista dado que las fuentes liberan recursos en ausencia de realimentación. Es por esta razón una estrategia apropiada cuando la señal de realimentación se pierde debido a congestión en la red. No obstante, esta estrategia puede provocar situaciones en las que algunas fuentes dejen de recibir realimentación de una forma indiscriminada. Por ejemplo, una conexión que atraviese muchos conmutadores podría encontrar una tasa alta de pérdida de mensajes de realimentación, lo cual provocaría una asignación más reducida de ancho de banda que otras conexiones con un trayecto más corto.

Por último, en la realimentación bipolar la fuente mantiene su tasa por defecto mientras el destino no le solicite el cambio a otro valor determinado de tasa. Tanto en la realimentación negativa como en la positiva, la fuente suele converger gradualmente hacia el valor correcto de tasa a través de sucesivos aumentos y disminuciones en la tasa de envío. Esta estrategia tiene como consecuencia un comportamiento estable, aunque oscilatorio, de la red dado que la evolución del sistema es en todo momento gradual. Por otro lado, una conexión recién establecida precisa de un periodo de tiempo para ocupar el ancho de banda que le corresponde y, al mismo tiempo, la red sólo podrá recuperarse lentamente de situaciones de congestión. Con la realimentación bipolar, la fuente puede ajustarse de forma inmediata a la tasa correcta en lugar de converger gradualmente a ella. De este modo, la adquisición de ancho de banda es más rápida, como también lo es la desaparición de congestión en la red. Sin embargo, estas ventajas se consiguen a costa de incorporar a la red la función de determinar la tasa correcta de envío para cada conexión.

4.3.1 Evolución de la definición del algoritmo de ajuste en ABR

La idea de controlar mediante realimentación explícita la tasa de las fuentes fue incorporada a protocolos normalizados en la norma *Frame Relay* de ANSI. Muy sucintamente, se contemplaba la posibilidad de que la cabecera de la trama contuviese dos bits que se

utilizarían opcionalmente como indicadores de congestión en la red. No se obligaba a que la fuente ajustase de una forma determinada a esta señal de realimentación, y presentaba a nivel informativo únicamente un algoritmo específico de ajuste de tasa. El algoritmo que se recomendaba era una traducción del algoritmo de Ramakrishnan-Jain, que fue utilizado en el protocolo DECnet para el ajuste del tamaño de la ventana de emisión, para permitir la modulación de la tasa de emisión de tramas. En el algoritmo de Ramakrishnan-Jain, la fuente periódicamente decidía entre aumentar el tamaño de su ventana en una cantidad fija, o bien reducirla proporcionalmente al tamaño actual de la misma. Este ajuste conducía a un aumento lineal o una disminución exponencial del tamaño de la ventana en función del tiempo. La realimentación utilizada era hacia adelante y explícita binaria; la señal de realimentación era recibida por el destino, quien se encargaba de comunicarla a la fuente a través de los paquetes de reconocimiento.

La primera propuesta en el ATM Forum para el soporte de ABR a través de control por ajuste de tasa la hizo Newman (1994). En el esquema propuesto, aquel nodo que eventualmente sufriera congestión generaría periódicamente realimentación negativa hacia atrás mediante células RM. Si, durante un intervalo de la misma duración, la fuente recibía una señal de realimentación, ésta reduciría su tasa de forma proporcional al valor actual. En caso contrario, aumentaría su tasa también de forma proporcional. Este esquema conseguía sencillez y economía en la implementación, pues los cambios de tasa se podían computar mediante desplazamientos lógicos hacia la derecha o hacia la izquierda y, además, las células RM sólo se generaban en caso de congestión. Se concibió para su uso en redes LAN.

La segunda propuesta aprarecida en el ATM Forum se atribuye a Yin y Hluchyj (1994). Proponían emplear como señal de realimentación hacia adelante y explícita binaria un bit EFCI en la cabecera de todas las células ATM. De forma periódica, el destino comprobaría si la célula recibida más recientemente contenía el bit EFCI a 1 y, en tal caso, enviaría una célula RM a la fuente otorgándole permiso para aumentar su tasa en una cantidad fija. Si, a lo largo de un periodo de la misma duración, la fuente no recibía permiso para aumentar su tasa, la disminuiría en una cantidad proporcional a la tasa permitida en ese momento. Tal tasa permitida podría ajustarse entre un valor mínimo y un valor máximo.

El esquema Hluchyj-Yin era análogo al algoritmo Ramakrishnan-Jain en cuanto a que:

- El ajuste de la tasa en función del tiempo muestra aumentos lineales y disminuciones exponenciales.
- La realimentación es positiva y el ajuste es periódico.
- Fue concebido para su uso tanto en redes LAN como WAN.

Sin embargo, difería del algoritmo Ramakrishnan-Jain en dos aspectos:

- El destino interpreta la activación de un bit EFCI como razón suficiente para no enviar la señal de realimentación positiva, mientras que el algoritmo Ramakrishnan-Jain basa su decisión de aumentar o disminuir su ventana en el porcentaje de indicadores de congestión activados.

- La realimentación es positiva, lo cual hace que el esquema sea robusto frente a pérdidas y retrasos en la señal de realimentación, aunque conlleva que las células RM consuman ancho de banda cuando la red no está de hecho congestionada.

El esquema Hluchyj-Yin fue apoyado por numerosos miembros del ATM Forum, que a su vez contribuyeron a refinar la propuesta, dando lugar a un esquema altamente flexible que Barnhart (1994a) propuso bajo la denominación de *Proportional Rate Control Algorithm* (PRCA). El esquema PRCA se caracterizaba por:

- Emplear realimentación positiva, al igual que el esquema Hluchyj-Yin. Más específicamente, la fuente disminuía proporcional y continuadamente su tasa permitida de emisión hasta que recibía una señal de realimentación, que le permitía compensar la disminución experimentada durante el último periodo además de aumentar la tasa permitida en una cantidad fija.

- Limitar el ancho de banda consumido por la realimentación —esto es, por las células RM— a un porcentaje fijo del ancho de banda disponible para ABR, a diferencia del esquema Hluchyj-Yin, en donde este porcentaje aumentaba al aumentar el número de conexiones y también al disminuir el ancho de banda disponible para ABR.

- Conseguir que la velocidad de crecimiento de la tasa permitida a cada conexión fuese proporcional a esta tasa, para así conseguir que la velocidad de crecimiento de la suma de las tasas permitidas fuese proporcional a esta suma, la cual es independiente del número de conexiones activas. De este modo, las fluctuaciones del tamaño de las colas en los conmutadores, que dependen principalmente de las fluctuaciones de la tasa agregada, serían independientes del número de conexiones activas.

- Incorporar un mecanismo *use-it-or-lose-it*, en virtud del cual la tasa permitida de una fuente se vería reducida a un valor predeterminado si la fuente no estuviese haciendo uso efectivo de la tasa permitida. Este mecanismo permitiría asegurar el mecanismo frente a retenciones intencionadas de tasa permitida por parte de algunos usuarios.

Las características segunda y tercera se consiguieron haciendo que el destino generase una célula BRM por cada N células recibidas desde la fuente, si la célula N-ésima no tenía el bit EFCI activado. De esta manera, cuando la fuente recibiese señal de realimentación positiva, aumentaría en una cantidad fija el valor de su tasa permitida. El aumento así conseguido es de hecho exponencial, pues la frecuencia de los aumentos es proporcional a la tasa de emisión. Nótese que este crecimiento exponencial trae consigo que, ante un eventual aumento del ancho de banda disponible, las conexiones con menor tasa de emisión aumenten su tasa más lentamente.

Al mismo tiempo, Charny y otros (1995) proponían basar el control por ajuste de tasa en realimentación de valor explícito. Se conseguiría, en principio, de este modo un ajuste más rápido a la tasa correcta en cada momento y un comportamiento menos oscilatorio que con realimentación explícita binaria. Además, no supondría ningún aumento en la

complejidad de la red, pues si la red se decidiese a controlar el tráfico de entrada a la red, tendría entonces que calcular la tasa correcta de emisión para cada una de las conexiones.

La alternativa de realimentación de valor explícito podría implementarse fácilmente si la fuente generase un flujo continuado de células RM, cada una de las cuales contendría un campo para el valor de tasa explícita, y si el destino las devolviese hacia la fuente. Cada conmutador en el trayecto de la conexión podría entonces reducir el campo de tasa explícita bien en el camino de ida, bien en el de vuelta de la conexión. Asimismo, tal como propusieron Charny y otros (1995), las células RM podrían contener un campo que indicase la tasa permitida de emisión en el momento en que fue generada la célula en la fuente; esta medida facilitaría a los conmutadores el cómputo de la tasa correcta para cada conexión.

En una contribución importante, Fedorkow y A. Jain mostraron cómo unificar los esquemas PRCA y de realimentación por tasa explícita en un mismo algoritmo de ajuste. Tal unificación consistió en que la fuente interpretara la tasa explícita realimentada como un tope máximo a la tasa calculada según indicaba PRCA. En otras palabras, la fuente computaría la tasa permitida de emisión a partir de la tasa permitida previa y la realimentación positiva binaria que pudiese haber recibido, pero a continuación tomaría como nueva tasa permitida de emisión el mínimo entre la recién calculada y la tasa explícita realimentada más recientemente por la red. Tal como precisaba el esquema de realimentación por tasa explícita, la fuente generaría un flujo de células RM, que el destino reenviaría a la fuente. Pero, tal como PRCA establecía, la generación de células RM tendría lugar a una tasa proporcional a la tasa permitida de emisión y, además, cada célula RM contendría un campo de realimentación binaria. De este modo, los conmutadores en el trayecto de la conexión podrían realimentar a la fuente haciendo uso del campo de tasa explícita, del indicador binario, o de ambos simultáneamente.

La síntesis de los esquemas PRCA y de realimentación por tasa explícita se denominó *enhanced PRCA* (EPRCA). Finalmente, en abril de 1996, el ATM Forum aprobó la especificación *ATM Forum Traffic Management 4.0*, que, en particular, normalizaba el algoritmo de ajuste en la fuente de cada conexión ATM establecida bajo la categoría de servicio ABR. Asimismo normalizaba la participación del destino y de los conmutadores en el bucle de control de flujo por realimentación. Las dos diferencias más significativas respecto del esquema EPRCA fueron, primero, que las fuentes mantienen sus tasas permitidas de emisión entre la recepción de dos células RM consecutivas, a diferencia del esquema EPRCA en que las fuentes disminuían su tasa permitida de forma continuada; y segundo, que se incorporaba un mecanismo de *back-off* sobre la tasa permitida de emisión, el cual protegía el sistema ante la eventual pérdida de células RM por congestión.

4.3.2 Algoritmo de ajuste en ABR según *ATM Forum Traffic Management 4.0*

A continuación se describe el comportamiento de fuente tal como se especifica en *ATM Forum Traffic Management 4.0*, en donde se enumera trece reglas de comportamiento en lenguaje informal. Tras citar la regla en inglés, se discute su significado y su trascendencia (Jain y otros, 1996).

Previamente, discutiremos la diferencia entre células RM dentro de tasa (*in-rate*) y fuera de tasa (*out-of-rate*), dado que el tratamiento que de cada uno de tales tipos se hace en el control de tasa es diferente. La mayoría de las células RM que generan las fuentes de las conexiones ABR se tienen en cuenta al igual que las células de datos, en el sentido de que la suma de las tasas de emisión de las células de datos y de las células RM no debe superar la tasa permitida de la fuente. Tales células RM reciben el nombre de células RM dentro de tasa. No obstante, bajo circunstancias excepcionales, tanto el destino como los conmutadores, e incluso la fuente, pueden generar células RM extras. Tales células RM no se computan dentro de la tasa permitida de la fuente y se denominan células RM fuera de tasa. Se distinguen de las células dentro de tasa en que tienen el bit CLP puesto a 1, lo cual supone que ante situaciones de sobrecarga en los conmutadores serán las primeras células en ser descartadas. Las células RM fuera de tasa pueden generarse, en la fuente o en los conmutadores, a una tasa no superior a 10 células por segundo para cada conexión. Algunos de los ejemplos de uso de las células RM fuera de tasa son los siguientes:

- como señal de realimentación hacia atrás generada por los conmutadores;
- para obtener información sobre el estado de la red por parte de una fuente cuya tasa permitida ha disminuido hasta cero;
- cuando la tasa permitida en el camino de vuelta en una conexión ABR es insuficiente, o incluso nula, para que el destino pueda devolver todas las células FRM que recibe.

Nótese que la distinción entre células dentro de tasa y fuera de tasa sólo se aplica a las células RM. Todas las células de datos en ABR deben tener su bit CLP puesto a 0 y deben ser emitidas dentro de la tasa permitida por la red.

Regla de comportamiento de fuente nº 1

> The value of ACR shall never exceed PCR, nor shall it ever be less than MCR. The source shall never send in-rate cells at a rate exceeding ACR. The source may always send in-rate cells at a rate less than or equal to ACR.

Las fuentes deben siempre emitir a una tasa igual o por debajo de la tasa permitida de emisión, en adelante tasa ACR. La tasa ACR no puede exceder el valor PCR y no necesita disminuir por debajo del valor MCR.

Regla de comportamiento de fuente nº 2

> Before a source sends the first cell after connection setup, it shall set ACR to at most ICR. The first in-rate cell sent shall be a forward RM-cell.

Al inicio de la transferencia en una conexión, la tasa permitida es igual al valor ICR. Además, la primera célula emitida es siempre una célula FRM dentro de tasa, lo cual favorece que la realimentación se reciba de la red lo antes posible.

Regla de comportamiento de fuente nº 3

> After the first in-rate forward RM-cell, in-rate cells shall be sent in the following order:
>
> 1. The next in-rate cell shall be a forward RM-cell if and only if, since the last in-rate forward RM-cell was sent, either:
>   (a) at least M_{rm} in-rate cells have been sent and at least T_{rm} time has elapsed, or
>   (b) $N_{rm} - 1$ in-rate cells have been sent.
> 2. The next in-rate cell shall be a backward RM-cell if condition (a) above is not met, if a backward RM-cell is waiting for transmission, and if either:
>   (a) no in-rate backward RM-cell has been sent since the last in-rate forward RM-cell, or
>   (b) no data cell is waiting for transmission.
> 3. The next in-rate cell sent shall be a data cell if neither condition (a) nor condition (b) above is met, and if a data cell is waiting for transmission.

En todo momento, una fuente tiene tres tipos de células que emitir: células de datos, células FRM y células BRM correspondientes al flujo de datos en sentido inverso. La prioridad de cada uno de estos tres tipos depende del estado de la transmisión.

En primer lugar, se requiere que la fuente envíe una célula FRM cada N_{rm} células. Sin embargo, si la tasa de emisión es baja, el tiempo transcurrido entre células FRM será grande y por tanto se retrasará la llegada de la realimentación. Por tanto, se especifica que se genere una célula FRM si ha transcurrido más de T_{rm} segundos desde que se envió la última célula FRM. Esta medida puede repercutir negativamente sobre las fuentes de baja actividad, pues si cada oportunidad de transmisión aparece con más de T_{rm} de intervalo, se enviará una célula FRM y, por tanto, nunca se enviará una célula de datos. Para resolver este inconveniente, se ha añadido la condición de que debe haber al menos otras M_{rm} células entre dos células FRM.

La frecuencia de las células FRM viene determinada por los parámetros N_{rm}, T_{rm} y M_{rm}, cuyos valores por defecto son 32, 100 ms y 2, respectivamente. En la práctica, cuanto menor es el valor de N_{rm} que se elija, mayor será la sensibilidad del control y mayor será la carga de procesado que se impone sobre los conmutadores y las estaciones. Por ejemplo, para una conexión a 155 Mbit/s el tiempo entre células RM es de 86.4 μs, mientras que es de 8.6 ms para la misma conexión a 1.55 Mbit/s. De todos modos, el factor que desaconseja valores altos para la frecuencia de las células FRM es la ineficiencia que supone dejar de emplear $1/N_{rm}$ del ancho de banda disponible.

En segundo lugar, una célula BRM en espera de transmisión tiene prioridad sobre una célula de datos, siempre que ninguna célula BRM haya sido enviada desde la emisión de la última célula FRM. Por supuesto, si no hay ninguna célula de datos en espera de transmisión, la célula BRM en espera se transmitirá.

Finalmente, las células de datos tienen prioridad sobre los *slots* restantes.

Las segunda y tercera partes de la regla de comportamiento garantizan que las células BRM no son retrasadas innecesariamente y que no todo el ancho de banda disponible es utilizado por células RM.

Regla de comportamiento de fuente nº 4

> Cells sent in accordance with source behaviors #1, #2, and #3 shall have CLP=0.

Todas las células RM que se envíen según las reglas 1, 2 y 3 son células RM dentro de tasa, por lo que se envían con CLP=0. Células RM adicionales podrán enviarse fuera de tasa, para lo cual deberán tener CLP=1.

Regla de comportamiento de fuente nº 5

> Before sending a forward in-rate RM-cell, if ACR > ICR and the time T that has elapsed since the last in-rate forward RM-cell was sent is greater than ADTF, then ACR shall be reduced to ICR.

La tasa permitida sólo se considera válida durante un intervalo de duración igual al *ACR Decrease Time Factor* (ADTF), cuyo valor por defecto es 500 ms. Si una fuente no emite ninguna célula RM durante ese intervalo, se considera que ya no utiliza la tasa ACR que se le había asignado. En tal caso, la fuente debe obtener de nuevo realimentación desde la red, para lo cual emite una célula FRM, al tiempo que disminuye su tasa ACR al valor ICR negociado durante el establecimiento de la conexión. Si la tasa ACR ya fuese menor que el valor ICR, deberá la fuente permanecer a aquel valor de tasa permitida.

Esta regla tan sencilla fue la causa de un debate intenso dentro del proceso de definición de la especificación. El objetivo de esta regla es resolver el problema conocido como retención de ACR. Si una fuente envía una célula RM cuando la red no está cargada, la fuente puede obtener una tasa ACR muy alta. Esta fuente podría retener este valor de tasa ACR y emplearlo cuando la red sí estuviese cargada. Es más, una fuente podría incluso establecer varias conexiones y así obtener una ventaja injusta. Con el objeto de resolver este problema, se propuso varias soluciones denominadas *Use-It-or-Lose-It* (UILI). Algunas de estas soluciones se basaban en acciones tomadas por la fuente; otras relegaban la funcionalidad a los conmutadores. En el primer caso, se requería de las fuentes que comprasen sus tasas de emisión y que redujesen su tasa ACR gradualmente si ésta fuese mucho mayor que aquella. Las soluciones UILI tienen un impacto muy significativo en las prestaciones de las fuentes con un perfil esporádico de tráfico, que son las que constituyen el grueso del tráfico de datos. Finalmente, el ATM Forum optó por normalizar un algoritmo UILI muy sencillo, que estaba basado en un temporizador. Los fabricantes de equipo pueden añadir cualquier algoritmo más complicado y propietario en la fuente o en el conmutador. Kalyanaraman (1997) expone muchos de los aspectos que se discutieron en el diseño de soluciones UILI.

Regla de comportamiento de fuente nº 6

> Before sending an in-rate forward RM-cell, and after following behavior #5 above, if at least CRM in-rate forward RM-cells have been sent since the last backward RM-cell with BN=0 was received, then ACR shall be reduced by at least ACR·CDF, unless that reduction would result in a rate below MCR, in which case ACR shall be set to MCR.

Dado que las células RM pueden quedar bloqueadas en las colas de los conmutadores, a causa de caídas de los enlaces o congestión severa en los nodos, las fuentes pueden no recibir realimentación. Se especifica que la fuente reduzca su tasa ACR si la realimentación de la red no se recibe oportunamente. De este modo, la red queda protegida ante flujos incontrolados de entrada en tales eventualidades.

Cuando la red funciona de forma estable, toda fuente debería recibir una célula FRM por cada célula FRM que envíe. Sin embargo, en caso de congestión, las células BRM pueden retrasarse. Cuando una fuente ha enviado CRM (*missing RM cell count*) células FRM sin haber recibido ninguna célula BRM, debe suponer que se ha producido congestión en la red y deberá reducir su tasa ACR por un factor CDF (*Cutoff Decrease Factor*). Los valores CRM y CDF se negocian durante el establecimiento de la conexión. Téngase en cuenta que las células generadas por los conmutadores, que se identifican con BN=1, no se cuentan como BRM. Además, una vez que se dispara la condición nº 6, también se satisface para cada una de las siguientes células FRM que se envíen hasta que se reciba una célula BRM. Por tanto, se produce una rápida disminución exponencial en la tasa ACR.

Un efecto importante de esta regla es que la condición puede dispararse innecesariamente cuando el retardo de ida y vuelta es grande, a no ser que se tome un valor CRM sufientemente alto. Es por ello que CRM se calcula a partir de otro valor denominado *Transient Buffer Exposure* (TBE), que también se negocia durante el establecimiento de la conexión. TBE determina el máximo número de células que se permite que entren en la red durante el trayecto de ida y vuelta de la primera célula FRM, periodo en el cual la fuente no ha recibido ninguna realimentación desde la red. Además, durante este intervalo, la fuente habrá enviado $\mathrm{TBE}/\mathrm{N_{rm}}$ células RM cells. Por tanto, el parámetro CRM deberá ser igual a

$$\mathrm{CRM} = \left\lceil \frac{\mathrm{TBE}}{\mathrm{N_{rm}}} \right\rceil$$

Además, durante el establecimiento de la conexión se calcula la parte invariable del retardo de ida y vuelta (*Fixed part of the Round-Trip Time*, FRTT), esto es, el mínimo retardo en el trayecto de ida y vuelta, sin incluir los retardos por espera en las colas de los conmutadores. Durante un intervalo de duración FRTT, una fuente recién activada puede enviar ICR·FRTT células como máximo. A pesar de que el valor FRTT se negocia separadamente, debe mantenerse las siguientes relaciones entre ICR y TBE

$$\mathrm{ICR} \cdot \mathrm{FRTT} \leq \mathrm{TBE} \qquad \text{o} \qquad \mathrm{ICR} \leq \mathrm{TBE}/\mathrm{FRTT}$$

Por añadidura, las fuentes deben reducir el valor ICR negociado con la red para que cum-

pla las condiciones anteriores, es decir,

$$\text{ICR fuente} = \min\{\text{ICR negociado}, \text{TBE/FRTT}\}$$

Cuando la red negocia el valor TBE, los conmutadores deben tener en cuenta el espacio de almacenamiento disponible. Como su nombre indica, un conmutador puede quedar expuesto en cualquier instante durante el primer trayecto de ida y vuelta a la llegada de TBE células. Así, si los conmutadores no disponen de muchos *buffers*, el valor TBE negociado deberá ser bajo; no obstante, un valor TBE pequeño provocará disparos espurios de la condición nº 6.

Regla de comportamiento de fuente nº 7

> After following behaviors #5 and #6 above, the ACR value shall be placed in the CCR field of the outgoing forward RM-cell, but only in-rate cells sent after the outgoing forward RM-cell need to follow the new rate.

Cuando se envíe una célula FRM, la fuente deberá insertar su tasa ACR actual en el campo CCR de la célula.

Regla de comportamiento de fuente nº 8

> When a backward RM-cell (in-rate or out-of-rate) is received with CI=1, then ACR shall be reduced by at least ACR·RDF, unless that reduction would result in a rate below MCR, in which case ACR shall be set to MCR. If the backward RM-cell has both CI=0 and NI=0, then the ACR may be increased by no more than RIF×PCR, to a rate not greater than PCR. If the backward RM-cell has NI=1, the ACR shall not be increased.

Regla de comportamiento de fuente nº 9

> When a backward RM-cell (in-rate or out-of-rate) is received, and after ACR is adjusted according to source behavior #8, ACR is set to at most the minimum of ACR as computed in source behavior #8, and the ER field, but no lower than MCR.

Las reglas 8 y 9 describen cómo debe reaccionar la fuente ante la recepción de la señal de realimentación. Esta consiste en el campo ER y en los bits CI y NI. Una solución inmediata hubiera podido ser que la fuente cambie su tasa ACR al valor ER recibido; pero esto habría ocasionado diversos problemas:

- Si el nuevo valor ER es muy grande en comparación con el valor ACR actual, pasar directamente a ER causaría cambios bruscos en el nivel de llenado de las colas. Es por ello que se limita el crecimiento de la tasa ACR por medio de un *Rate Increase Factor* (RIF) que determina el máximo aumento posible en cada paso. Así, la fuente no puede aumentar su tasa ACR en más de RIF·PCR.

- Si existe algún conmutador capaz únicamente de ejercer control a través del bit EFCI de las células, el campo ER no se verá modificado. En tal caso, se especifica, como veremos más adelante, que el destino monitorice el bit EFCI y devuelva el valor del último bit EFCI observado en el bit CI de una célula BRM. Un bit CI igual a 1 significa que la red está congestionada y obliga a la fuente a reducir su tasa ACR; esta reducción es multiplicativa y viene determinada por un factor *Rate Decrease Factor* (RDF) de modo que

$$\text{ACR} \leq \text{ACR}(1 - \text{RDF})$$

Se ha demostrado que un aumento aditivo combinado con una disminución multiplicativa es suficiente para conseguir equidad (Chiu y Jain, 1989). Otras combinaciones como aumento y disminución aditivos, aumento y disminución multiplicativos y aumento multiplicativo con disminución aditiva no son equitativos en el sentido *max-min*.

El bit NI se ha introducido en la especificación para considerar situaciones de congestión inminente. En estos casos, el conmutador puede especificar un valor de tasa explícita ER, y activar el bit NI para indicar a la fuente que, aun en el caso de que la tasa ACR esté por debajo de ER, no debe aumentar su tasa ACR.

En general, las acciones correspondientes a cada combinación de valores CI y NI se muestran en la tabla 4.1:

NI	*CI*	*Acción*
0	0	ACR ← min(ER, ACR+RIF·PCR, PCR)
0	1	ACR ← min(ER, ACR−ACR·RDF)
1	0	ACR ← min(ER, ACR)
1	1	ACR ← min(ER, ACR−ACR·RDF)

Tabla 4.1. AJUSTES POSIBLES EN EL SISTEMA FINAL SEGÚN NI Y CI

Regla de comportamiento de fuente nº 10

> When generating a forward RM-cell, the source shall assign values to the various RM-cell fields as specified for source-generated cells [...].

Las células FRM deben inicializar sus campos del siguiente modo:

1. en conexiones de trayecto virtual (VPC), el identificador VCI será invariablemente 6; en conexiones de circuito virtual, se empleará el que corresponda;
2. en cualquier caso el identificador *Protocol Type Id* (PTI) de la cabecera de la célula será 6; el identificador *protocol id* de la carga útil de la célula RM será 1;
3. el bit de dirección DIR será 0 (ida);
4. el bit BN será 0, para indicar que la célula RM ha sido generada por la fuente;

5. el campo ER se inicializa al máximo valor de tasa por debajo del valor PCR que la fuente puede soportar;

6. el campo CCR se inicializa con el valor ACR actual;

7. el campo MCR se inicializa con el valor negociado durante el establecimiento;

8. los campos QL, SN y R/A se inicializan a 0 o según UIT-T I.371.

Se contempla la posibilidad que la misma fuente asigne valores apropiados en ER y NI para indicar congestión local.

Regla de comportamiento de fuente nº 11

> Forward RM-cells may be sent out-of-rate (i.e., not conforming to the current ACR). Out-of-rate forward RM-cells shall not be sent at a rate greater than TCR.

La tasa de generación de células RM fuera de tasa viene limitada por el valor *Target Cell Rate* (TCR), que toma el valor por defecto de 10 cell/s

Regla de comportamiento de fuente nº 12

> A source shall reset EFCI on every data cell it sends.

El bit EFCI debe ponerse a 0 en cada célula de datos que se envíe.

Regla de comportamiento de fuente nº 13

> The source may implement a use-it-or-lose it policy to reduce its ACR to a value which approximates the actual cell transmission rate [...].

Como ya se ha comentado, la fuente puede implementar un mecanismo UILI adicional

Pasamos a describir el comportamiento normalizado en *ATM Forum Traffic Management 4.0* para el destino en el bucle de control de flujo.

Regla de comportamiento de destino nº 1

> When a data cell is received, its EFCI indicator is saved as the EFCI state of the connection.

El destino del flujo de una conexión ABR debe monitorizar el bit EFCI de las células que reciba y almacenar el valor observado más recientemente como estado EFCI de la conexión.

Regla de comportamiento de destino nº 2

> On receiving a forward RM-cell, the destination shall turn around the cell to return to the source. The DIR bit in the RM-cell shall be changed from "forward" to "backward", BN shall be set to zero, and CCR, MCR, ER, CI, and NI fields in the RM-cell shall be unchanged except:
>
> 1. If the saved EFCI state is set, then the destination shall set CI=1 in the RM-cell, and the saved EFCI state shall be reset. It is preferred that this step is performed as close to the transmission time as possible;
> 2. The destination (having internal congestion) may reduce ER to whatever rate it can support and/or set CI=1 or NI=1. A destination shall either set the QL and SN fields to zero, preserve these fields, or set them in accordance with ITU-T Recommendation I.371-draft. The octets defined [...] as reserved may be set to 6A (hexadecimal) or left unchanged. The bits defined as reserved [...] for octet 7 may be set to zero or left unchanged. The remaining fields shall be set in accordance with [...] (Note that this does not preclude looping fields back from the received RM-cell).

Se especifica que el destino debe devolver a la fuente las células FRM que reciba, modificando su contenido según sigue:

1. El bit de dirección DIR se cambia a 1 para indicar que se trata de una célula BRM.
2. El bit BN se pone a 0 para indicar que no se trata de una célula generada por la red.
3. Los campos CCR y MCR no deben modificarse.
4. Si el bit EFCI de la última célula de datos que se recibió era 1, el bit CI de la célula BRM se pondrá a 1 y reinicializar el estado EFCI de la conexión. Nótese que el estado EFCI de una conexión se activa y se desactiva cada vez que llega una célula de datos con el bit EFCI a 1 y a 0 respectivamente.
5. Si el destino experimenta congestión interna, puede reducir el valor ER o poner el bit CI o el bit NI a 1, análogamente a un conmutador. Esta opción es de utilidad en la configuración VS/VD, en donde el destino virtual constituya un cuello de botella a causa de la tasa permitida por el segmento siguiente. En cualquier caso, el valor ER nunca deberá aumentarse.

Regla de comportamiento de destino nº 3

> If a forward RM-cell is received by the destination while another turned-around RM-cell (on the same connection) is scheduled for in-rate transmission:
>
> 1. It is recommended that the contents of the old cell are overwritten by the contents of the new cell;

2. It is recommended that the old cell (after possibly having been over-written) shall be sent out-of-rate; alternatively the old cell may be discarded or remain scheduled for in-rate transmission;
3. It is required that the new cell be scheduled for in-rate transmission

Regla de comportamiento de destino nº 4

> Regardless of the alternatives chosen in destination behavior #3 above, the contents of an older cell shall not be transmitted after the contents of a newer cell have been transmitted.

En principio, el destino debe devolver las células RM tan pronto como sea posible; no obstante, una célula BRM puede retrasarse si la tasa ACR en el sentido de vuelta es bajo. En tales ocasiones, se contempla la posibilidad de que las células BRM con información caducada sean descartadas. Alternativamente, la célula BRM con información caducada puede enviarse fuera de tasa.

Si la tasa ACR de vuelta no es cero, la célula BRM se enviará dentro de tasa. La transmisión de células BRM fuera de tasa por su parte, permite que las células BRM sean devueltas regularmente, incluso cuando la tasa ACR es cero. Nótese que no existe un límite especificado sobre la tasa de devolución de células BRM fuera de tasa. Las implicaciones de cada una de las alternativas dadas en la regla 3 se detallan en el *Informative Appendix I* de la especificación *ATM Forum Traffic Management 4.0*; básicamente.

Regla de comportamiento de destino nº 5

> A destination may generate a backward RM-cell without having received a forward RM-cell. The rate of these backward RM-cells (including both in-rate and out-of-rate) shall be limited to 10 cells/second, per connection. When a destination generates an RM-cell it shall set either CI=1 or NI=1, shall set BN=1, and shall set the direction to backward. The destination shall assign values to the various RM-cell fields as specified for destination generated cells in Table 0-4.

Se permite que el destino genere células RM de vuelta, en caso de que el destino experimente congestión muy severa y no pueda esperar a la recepción de una célula FRM. En tal caso, la célula RM de vuelta no debe solicitar un aumento de tasa —para lo cual CI=1—; además, la tasa de generación de estas células no será superior a 10 cell/s y deben enviarse fuera de tasa, esto es, con CLP=1.

Regla de comportamiento de destino nº 6

> When a forward RM-cell with CLP=1 is turned around it may be sent in-rate (with CLP=0) or out-of-rate (with CLP=1).

Una célula FRM fuera de tasa puede ser devuelta tanto dentro de tasa como fuera de tasa.

Por último, mencionar que, si bien entre la fuente y el destino de la conexión ABR sólo se requiere la actuación de un bucle de control de flujo, la especificación *ATM Forum Traffic Management 4.0* contempla la posibilidad de que se implemente una segmentación del bucle de control mediante *Virtual Source* (VSs) y *Virtual Destinations* (VDs). Efectivamente un conmutador intermedio de la red puede decidir cerrar el bucle de control comportándose respecto de la fuente de la conexión como un destino y como una nueva fuente respecto del destino de la conexión. Las razones principales para permitir VS/VD son, por un lado, reducir la latencia del bucle de control, lo cual siempre conduce a mejoras de prestaciones; por otro, crear dominios de control separados, por razones de tipo administrativo. El acoplamiento entre segmentos de control consecutivos, esto es, entre el destino virtual y la fuente virtual adyacente dentro de una conexión ABR es un aspecto sujeto a diferenciación por parte de los fabricantes.

4.4 Generación de la señal de realimentación

En cuanto a los conmutadores, la especificación *ATM Forum Traffic Management 4.0* sólo establece que deben incorporar un mecanismo de generación de señal de realimentación y se limita a especificar las diferentes alternativas de modificación de los campos de las células RM de ida y/o de vuelta. Sin embargo, los algoritmos específicos para calcular la señal de realimentación no quedan normalizados. Tanto este algoritmo como los mecanismos de VS/VD, de UILI, de planificación de transmisión de las células y de gestión de *buffers*, así como la selección de los valores de los parámetros de operación de ABR, son dependientes de la implementación y, por tanto, es un área para que los fabricantes de equipos diferencien sus productos.

El comportamiento especificado para los conmutadores presentes en el trayecto de las conexiones ABR se especifica a través de las siguientes cinco reglas:

Regla de comportamiento de conmutador nº 1

> A switch shall implement at least one of the following methods to control congestion at queuing points:
>
> 1. EFCI marking: The switch may set the EFCI state in the data cell headers;
> 2. Relative Rate Marking: The switch may set CI=1 or NI=1 in forward and/or backward RM-cells;
> 3. Explicit Rate Marking: The switch may reduce the ER field of forward and/or backward RM-cells;
> 4. VS/VD Control: The switch may segment the ABR control loop using a virtual source and destination.

Esta regla establece que el conmutador debe soportar, a través de los algoritmos oportunos, al menos uno de los siguientes mecanismos de generación de señal de realimentación, que denomina métodos de marcado:

- Marcado EFCI

  Constituye la alternativa de realimentación binaria, en la cual los conmutadores pueden poner a 1 el bit EFCI en la cabecera de las células de datos. Como se ha mencionado anteriormente, el destino almacena el estado EFCI de cada conexión y pone el bit CI de las células BRM a 1 si tal estado está activado.

- Marcado de tasa relativa

  Esta alternativa permite que el conmutador active los bits CI y NI de las células RM. Cuando la célula BRM llega a la fuente, el bit CI=1 obliga a la fuente a reducir su tasa permitida, mientras que el bit NI=1 le impide que la aumente, con independencia de la realimentación del campo ER. Estos dos bits proporcionan más flexibilidad a los conmutadores que con el marcado EFCI.

- Marcado de tasa explícita

  Permite al conmutador especificar la tasa máxima a la que desea que la fuente emita. Los conmutadores sólo pueden reducir, nunca aumentar, el valor del campo ER de las células RM de ida o de vuelta; de este modo, se asegura que la señal de realimentación que llega a la fuente es correcta y coherente.

- Control VS/VD

  En este método, el conmutador puede segmentar el bucle de control e implementar las funcionalidades de VS/VD.

Regla de comportamiento de conmutador nº 2

> A switch may generate a backward RM-cell. The rate of these backward RM-cells (including both in-rate and out-of-rate) shall be limited to 10 cells/second, per connection. When a switch generates an RM-cell it shall set either CI=1 or NI=1, shall set BN=1, and shall set the direction to backward. The switch shall assign values to the various RM-cell fields as specified for switch-generated cells [...].

Esta regla especifica cómo el conmutador puede generar una célula RM, en caso de experimentar congestión severa y no reciba células RM para conmutar. Básicamente, se especifica que estas células RM sólo pueden solicitar una reducción de la tasa permitida y que se envíen fuera de tasa.

Regla de comportamiento de conmutador nº 3

> RM-cells may be transmitted out of sequence with respect to data cells. Sequence integrity within the RM-cell stream must be maintained.

Esta regla establece que las secuencia de células RM de cada conexión debe mantenerse, aunque no necesariamente la secuencia definida por la fuente conjuntamente para células de datos y RM. De este modo, se permite que los conmutadores puedan opcionalmente asignar mayor prioridad temporal a las células RM que a las de datos, lo cual se traduciría en una realimentación más rápida en situaciones de congestión. Sin embargo, separar las células RM del flujo de células de datos, provoca que se pierda la correlación entre las cantidades que declaran en las células RM y los valores reales del flujo de células de datos.

Regla de comportamiento de conmutador nº 4

> For RM-cells that transit a switch (i.e., are received and then forwarded), the values of the various fields before the CRC-10 shall be unchanged except:
>
> 1. CI, NI, and ER may be modified as noted in #1 above
> 2. RA, QL, and SN may be set in accordance with ITU-T Recommendation I.371-draft
> 3. MCR may be corrected to the connection's MCR if the incoming MCR value is incorrect.

Esta regla especifica la alineación con UIT-T I.371 y, además, garantiza la integridad del campo MCR.

Regla de comportamiento de conmutador nº 5

> The switch may implement a use-it-or-lose-it policy to reduce an ACR to a value which approximates the actual cell transmission rate from the source. Use-it-or-lose-it policies are discussed in Appendix I.8.

4.4.1 Alternativas genéricas de diseño de un mecanismo de generación de señal de realimentación

A la hora de examinar las distintas alternativas para el diseño de algoritmos de soporte a la generación de la señal de realimentación, nos centraremos en cuatro aspectos:

1. La naturaleza de la señal de realimentación.
2. La elección del punto de trabajo.
3. La medida de la congestión.
4. La estimación de la tasa equitativa.

Al respecto de la naturaleza de la señal de realimentación, el algoritmo de generación de la señal de realimentación puede producir una realimentación agregada o individual.

En el esquema agregado, la realimentación que se envía a una fuente es función de la ocupación de los recursos en el conmutador. En caso de que éste experimente congestión, todas las fuentes reciben la misma señal de realimentación —que seguramente pedirá una reducción de tasa permitida— desde el conmutador, independientemente de qué fuente es realmente la causante de la congestión. Esta aproximación a la generación de la señal de realimentación se utilizó en la versión original del algoritmo Ramakrishnan-Jain (Ramakrishnan y Jain, 1990); así también, en el algoritmo *slow-start* de Jacobson (1988), aunque implícitamente, pues la señal de realimentación utilizada es una propiedad del mecanismo de descarte de paquetes, el cual suele ser descartar el paquete que llega cuando la cola está llena, y que es independiente de la implementación de TCP.

En el esquema individualizado, la señal de realimentación que recibe la fuente tiene un carácter más individualizado en cuanto que refleja cuál es la carga relativa que el conmutador está soportando por los paquetes que recibe de esa fuente. Este esquema se adoptó en la última versión del algoritmo Ramakrishnan-Jain (Ramakrishnan y otros, 1987).

A la hora de escoger el punto de trabajo por parte de un mecanismo de control de congestión, en las redes por conmutación de paquetes, existe un compromiso entre utilización de los enlaces y tiempo de espera en el conmutador. Cuando la utilización es baja, las colas en el conmutador son pequeñas y el tiempo de espera es bajo. Cuando la utilización es muy alta, las colas aumentan. Finalmente, cuando las colas se llenan, se descartan paquetes. En tal situación, aunque la utilización del enlace sea alta —de hecho es máxima pues el tamaño de las colas es mayor que cero—, el *throughput* efectivo que se consigue es bajo, puesto que no todos los paquetes consiguen llegar a su destino. En general, cuando consideramos redes enteras en lugar de conmutadores podemos sustituir los términos *utilización del enlace* y *tiempo de espera en el conmutador* por *throughput* y *retardo de transferencia*, respectivamente.

En la figura 4.1, se muestra la evolución típica del *throughput* y del retardo en una red a medida que varía la carga. El punto de trabajo cuyo *throughput* está próximo al 100% y en el que el retardo es moderado se denomina codo (*knee*) de la curva retardo-*throughput*. Formalmente, el codo es el punto donde la relación entre el *throughput* en el cuello de botella de la red y el retardo en ese punto de la red es máxima según variamos la carga de la red. De hecho, en redes que operan en condiciones óptimas, se observa un comportamiento estacionario con oscilaciones controladas del *throughput* próximo a 100% y del tamaño de las colas próximo a 0. Aquellos algoritmos que intentan fijar el punto de trabajo en este punto se denominan esquemas de *congestion avoidance*.

Si la carga aumenta por encima del codo, el *throughput* aumenta, aunque también lo hace el retardo. No obstante, a partir de un cierto valor de retardo, el *throughput* empieza a disminuir y el retardo crece de forma acusada —generalmente debido a mecanismos de retransmisión por *time-out* que operan en capas superiores—. Este punto se denomina *cliff* de la curva retardo-*throughput*. Este punto es un punto de trabajo inestable y, además, supone grandes retardos por espera en las colas, lo cual no es ventajoso.

Los puntos de trabajo intermedios entre el codo y el *cliff* pueden ser deseables. Tales puntos de trabajo presentan un *throughput* alto en regimen estacionario y, además, mantienen en las colas algunos paquetes en espera. De este modo, los eventuales aumentos

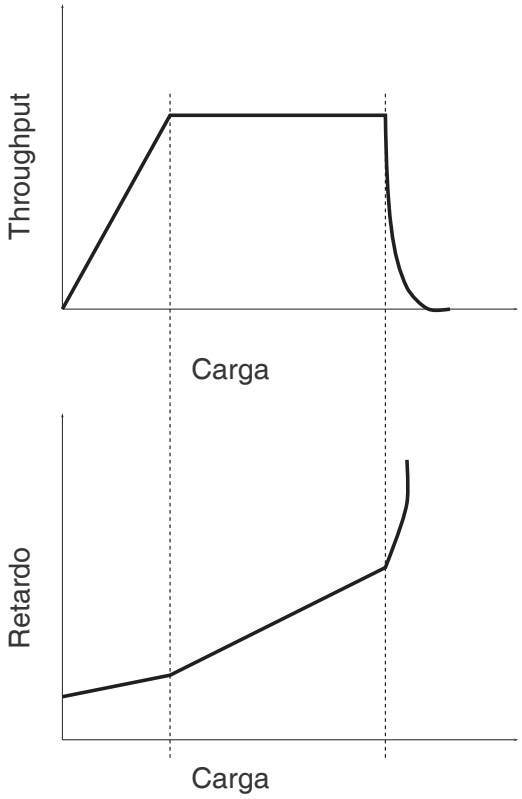

Figura 4.1. COMPORTAMIENTO REAL DE UNA RED DE COMPUTADORES A MEDIDA QUE VARÍA LA CARGA

de ancho de banda disponible en los nodos pueden ser aprovechados inmediatamente. Nótese que, sin embargo, estos puntos no son muy estables bajo control por ajuste de tasa, como se verá adelante.

Por último, la elección de la medida de la congestión que se emplee para generar la señal de realimentación tiene una repercusión importante sobre la complejidad de implementación del mecanismo de generación de realimentación y sobre la estabilidad y las prestaciones del bucle de control. Además, en tal elección se debe tener en cuenta la arquitectura del conmutador, el punto de trabajo deseado, los recursos que se desean proteger preferentemente (*buffers* o ancho de banda) y, finalmente, la escala de tiempo sobre la que se desea alcanzar el punto de trabajo. Enumeramos, a continuación, algunas de las alternativas posibles.

La opción más sencilla, que es además la más extendida, consiste en observar el tamaño instantáneo de la cola de células en espera de transmisión en los puertos de salida del conmutador. El puerto se considerará que ha entrado en congestión en el momento en que el tamaño de la cola supere un valor umbral dado. Este mecanismo es rudimentario pues usa una medida binaria para la congestión; sin embargo, puede detectar la utilización excesiva tanto de *buffers* como de ancho de banda, dado que la aparición de células en espera de transmisión refleja un exceso transitorio de demanda de ancho de banda, por encima del ancho de banda disponible en el puerto.

Una variación del mecanismo anterior es la utilización de más de un valor umbral, que permite diferenciar entre distintos grados de congestión. Téngase en cuenta, no obstante, que los umbrales constituyen puntos de discontinuidad, esto es, que la realimentación generada va a ser diferente si el sistema se encuentra a un lado u otro del punto. Estas discontinuidades en el control dan lugar a oscilaciones no despreciables, incluso sin considerar el efecto del retardo por realimentación (Rohrs y otros, 1995). Con el fin de reducir este comporamiento no deseable, puede introducirse histéresis en las discontinuidades, es decir, establecer un valor umbral aplicable cuando aumenta el tamaño de cola y otro valor umbral menor cuando el tamaño de cola disminuye.

Debe observarse que es más probable que se alcance un cierto valor umbral si el número de paquetes en la cola aumenta o disminuye rápidamente que si lo hace lentamente. Por tanto, es razonable escoger el valor de la derivada del tamaño de la cola con respecto al tiempo puede como medida de la congestión. Por un lado, una derivada positiva indica de forma más directa un exceso transitorio en la demanda de ancho de banda que simplemente el tamaño de la cola. Por otro, la derivada permite anticipar el efecto de los paquetes que lleguen al conmutador durante la realimentación; nótese que el número de paquetes que llegan al conmutador durante el tiempo que tarda en tener efecto la realimentación es mayor cuando mayor es la tasa de crecimiento de la cola. Como contrapartida, este esquema obliga a actualizar su estimación de la derivada cada determinado periodo de tiempo.

Finalmente, podemos estimar directamente la demanda de ancho de banda conjunta en los puertos de conmutador como indicador de la congestión. Concretamente, podemos utilizar la relación entre la tasa agregada de llegada de paquetes y el ancho de banda disponible en el puerto, que se denomina *factor de carga*. Si el factor de carga es menor

que la unidad, el puerto no está congestionado, incluso si el tamaño de la cola es grande, pues la cola tenderá a vaciarse. Por otro lado, si el factor de carga es mayor que la unidad, el sistema deberá iniciar acciones encaminadas a reducir la congestión, dado que la cola tenderá a crecer. Además de ser un indicador significativo del nivel de congestión en el puerto, el factor de carga es un indicador preciso del grado de sobrecarga. Por ejemplo, si la tasa de llegada de paquetes es de 20 células por segundo y el puerto sólo puede transmitir 10 células por segundo, sabemos que el factor de sobrecarga es 2 y que por tanto la tasa permitida a las fuentes deberá reducirse por la mitad. Estas conclusiones no pueden extraerse si se utiliza la derivada del tamaño de la cola con respecto del tiempo.

Otro argumento a favor de la utilización del factor de sobrecarga como medida de la congestión, que también es aplicable a la derivada del tamaño de la cola, tiene que ver con la naturaleza del control de flujo. En las redes clásicas de conmutación de paquetes el control de flujo se efectuaba mediante ventana deslizante; la mayoría de éstos mecanismos empleaban el nivel de llenado de las colas como indicador de la congestión. La combinación entre ventana deslizante y nivel de llenado de las colas funcionaba, porque el control por ventana deslizante puede controlar el tamaño de la cola: dado un tamaño de ventana, se garantiza en cualquier caso que el tamaño de la cola estará por debajo del tamaño de la ventana. No ocurre así en las redes ATM que proveen ABR, en donde el control se efectúa por ajuste de tasa: el control por ajuste de tasa sólo puede controlar la tasa de crecimiento de la cola. Efectivamente, dado un valor de tasa de llegada de paquetes al puerto, puede garantizarse que la tasa de crecimiento de la cola estará por debajo del valor de tasa de llegada, pero nada puede establecerse respecto al tamaño máximo de la cola. O, de otro modo, el tamaño de la cola no da información alguna sobre la diferencia entre la tasa real de llegada y la tasa ideal.

El cuarto aspecto a tratar en el diseño de los algoritmos de generación de la señal de realimentación es la estimación en el conmutador de la fracción equitativa de ancho de banda correspondiente a cada conexión ABR. Hemos visto que la especificación *ATM Forum Traffic Management 4.0* establece que el valor de realimentación hacia la fuente ha de ser directamente la tasa deseada de emisión o bien una indicación de congestión que permita calcular en la fuente la tasa permitida, por lo cual el conmutador ha de mantener una estimación de la asignación equitativa de ancho de banda para cada conexión. En las alternativas que se presentan a continuación se asume como criterio de equidad para la asignación de ancho de banda, el criterio *max-min*. Estas alternativas se corresponden con los algoritmos más conocidos que se emplearon durante el proceso de definición del control de flujo en ABR a partir del esquema EPRCA y que desembocó en la especificación *ATM Forum Traffic Management 4.0*; tales algoritmos son: EPRCA, CAPC y ERICA. Los agruparemos en algoritmos de cálculo aproximado de la tasa equitativa y de cálculo exacto (Arulambalam y otros, 1996).

4.4.2 Algoritmos de conmutador con cálculo aproximado de las tasas equitativas

Los algoritmos de esta categoría intentan aproximar la tasa equitativa *max-min* en el conmutador a partir del tamaño de la cola y del valor CCR que transportan las células RM. Más específicamente, cada conmutador mantiene actualizado para cada puerto de salida un promedio temporal de tasas, a partir del nivel de congestión medido y del valor CCR de las conexiones que atraviesan el puerto relevante, con el propósito de que en régimen estacionario este promedio se aproxime al valor teórico de tasa resultante de una asignación equitativa del ancho de banda del puerto de salida.

4.4.2.1 *Explicit Proportional Rate Control Algorithm* (EPRCA)

A partir del trabajo de Charny (1994) y de Siu y Tzeng (1994), se estableció que la tasa equitativa *max-min* en un conmutador podía calcularse como el cociente entre la diferencia entre la capacidad del enlace de salida y la capacidad de aquellas conexiones limitadas en el nodo y la diferencia entre el número total de conexiones activas y el número de conexiones limitadas (véase la sección 2.3.3.1). No obstante, el conocimiento de todos los valores involucrados en este cálculo no es siempre viable. (Roberts, 1994b) propuso un método alternativo más sencillo para calcular la tasa equitativa.

Partía de la intuición de que la tasa equitativa *max-min* era realmente el promedio de las tasas de las conexiones que no están limitadas en otros nodos. A continuación, propuso que se estimara la tasa equitativa *max-min* a partir del cálculo de la *Mean Allowed Cell Rate* (MACR), definida como

$$\text{MACR} = \text{MACR} + \alpha \cdot (\text{CCR} - \text{MACR})$$

donde α es un factor ponderador normalmente igual a 1/16. Durante los periodos sin congestión, este promedio tiene en cuenta únicamente aquellas conexiones tales que

$$\text{CCR} > \text{VCS} \cdot \text{MACR}$$

donde VCS se toma por defecto igual a 7/8; esta condición es una aproximación heurística para determinar aquellas conexiones no limitadas en el nodo. Cuando se detecta congestión en el nodo, el promedio MACR sólo tendrá en cuenta el valor CCR de aquellas conexiones tales que

$$\text{CCR} < \text{MACR}$$

La medida de la congestión en EPRCA se efectúa mediante la utilización de dos valores umbrales de tamaño de cola QT y DQT. Cuando el tamaño de cola supera el umbral QT, se declara congestión; el valor MACR se emplea para determinar selectivamente a qué conexiones solicitar una reducción de su tasa permitida. Existen dos alternativas:

1. Marcado de tasa relativa: se marca con CI=1 las células BRM de aquellas conexiones tales que CCR>DPF·MACR, donde DPF (*Down Pressure Factor*) se toma por defecto igual a 7/8.

2. Marcado de tasa explícita: si el valor ER de la célula BRM es mayor que ERF·MACR, donde ERF (*Explicit Reduction Factor*) se toma igual a 15/16, se sustituye aquél por éste.

Cuando el tamaño de cola supera el umbral DQT, se declara congestión severa; exiten dos alternativas:

1. Marcado de tasa relativa: se marca con CI=1 todas las células BRM.

2. Marcado de tasa explícita: si el valor ER de la célula BRM es mayor que MRF·MACR, donde MRF(*Major Reduction Factor*) se toma igual a 1/4, se sustituye aquél por éste.

Para asegurar la convergencia de los valores CCR y MACR bajo condiciones diversas, EPRCA emplea los coeficientes VCS, DPF, ERF y MRF. Se consigue básicamente que el conmutador trabaje en el punto de saturación y que los valores CCR de las conexiones se mantengan ligeramente por debajo del valor MACR.

La ventaja principal del algoritmo EPRCA es la baja complejidad de implementación, cuyos requisitos de memoria son $O(1)$, así como sus requisitos de cálculo, esto es, tanto la memoria como el tiempo de cómputo no dependen del número de conexiones activas.

Sin embargo, EPRCA necesita un ajuste conservador de los coeficientes para evitar un comportamiento oscilatorio grave en régimen estacionario, lo cual degrada las prestaciones del bucle de control en régimen transitorio. Además, no siempre se converge al valor de tasa equitativa, como, por ejemplo, cuando existen conexiones limitadas en otros conmutadores, cuyas tasas provocan una estimación a la baja de la tasa equitativa, o bien cuando aparecen transitorios, e incluso cuando el comportamiento de los usuarios no es cooperativo. Chiussi y otros (1996) proponen el algoritmo *Dynamic Max Rate Control Algorithm*(DMRCA), que incorpora algunas modificaciones que mejoran la equidad del algoritmo EPRCA.

En EPRCA, la técnica del promediado exponencial, que es similar a la media aritmética, no es apropiado para promediar ratios, tales como las tasas de emisión —que son número de células partido por tiempo—, cuando los denominadores no son iguales. En el caso concreto de EPRCA, provoca un sesgo hacia las tasas altas; por ejemplo, dadas dos fuentes con tasas de emisión de 1000 Mbit/s y 1 Mbit/s, durante un intervalo de tiempo dado, la primera fuente enviará 1000 veces más células RM que la segunda, de modo que un promediado ponderado exponencial estará sesgado hacia 1000 Mbit/s independientemente de la ponderación aplicada.

4.4.2.2 *Congestion Avoidance using Proportional Control* (CAPC)

Este esquema (Barnhart, 1994b) emplea las ideas del filtrado *phased locked loop* para converger hacia el valor de tasa equitativa. El conmutador opera como un regulador, en el cual la variable de control es la utilización del enlace y el valor de referencia se sitúa ligeramente por debajo del 100%. De forma más detallada, el conmutador calcula el factor de carga z, como el cociente entre la tasa agregada medida de llegada al puerto y la capacidad ABR disponible en el mismo, y utiliza este factor como variable de control a

partir de la cual obtiene una estimación de la tasa equitativa en el puerto. Nótese cómo el factor de sobrecarga es asimismo el indicador de congestión en CAPC, a diferencia de EPRCA, en el cual se empleaba el nivel de llenado de las colas.

Cuando $z < 1$, el puerto no está congestionado, y la estimación de tasa equitativa aumenta según:

$$\text{ERS} = \text{ERS} \cdot \min(ERU, 1 + (1 - z) \cdot R_{up})$$

donde R_{up} es la pendiente del aumento lineal del coeficiente y está en el rango 0.025 a 0.1. *ERU* es el coeficiente máximo de crecimiento para ERS y se toma 1.5.

Cuando $z > 1$, el puerto pasa a estar congestionado y la estimación de tasa equitativa se reduce:

$$\text{ERS} = \text{ERS} \cdot \max(ERF, 1 - (z - 1) \cdot R_{dn})$$

donde R_{dn} es la pendiente de la reducción lineal del coeficiente y está en el rango 0.02 a 0.8. *ERF* es la inversa del coeficiente máximo de reducción para ERS y se toma 0.5.

Este algoritmo opera únicamente en modo de marcado por tasa explícita y toma el valor ERS como máxima tasa permitida a una conexión.

Nótese que CAPC toma como punto de trabajo el codo de la curva retardo-*throughput*, siendo por tanto un mecanismo de *congestion avoidance*. Como mecanismo adicional, el algoritmo CAPC incorpora un umbral de tamaño de cola, transpasado el cual se considera que se ha producido congestión severa y se marcan con CI=1 las células RM de todas las conexiones.

Las aportaciones del algoritmo CAPC son:

- consigue en régimen estacionario un comportamiento libre de oscilaciones: las frecuencias de las oscilaciones es función de $1 - z$, por lo que en régimen estacionario, $z = 1$ y por tanto el periodo de las oscilaciones es infinito;
- al igual que EPRCA emplea un umbral único de tasa equitativa, lo cual permite que la complejidad de espacio sean también $O(1)$;
- no utiliza el valor CCR de las células RM en el control.

Por otro lado, en CAPC el tiempo de convergencia es largo dado que los parámetros R_{up} y R_{dn} se toman conservativamente. Además, nótese que, en situación de congestión severa, la realimentación es agregada, lo cual ocasiona situaciones de inequidad.

Martínez (1997) ha propuesto diversas mejoras del algoritmo CAPC. Una de ellas, denominada CAPC+, acelera la convergencia del valor ERS al valor teórico equitativo a través de dos modificaciones. En la primera, CAPC+ recuerda el máximo valor CCR observado durante periodos sin congestión para utilizarlo como estimación de partida de ERS en periodos de congestión. En la segunda, CAPC+ toma el cociente entre el ancho de banda disponible y el número de conexiones activas como valor mínimo de ERS, evitando así que el valor ERS disminuya excesivamente durante periodos de congestión (Martínez y otros, 1996). La segunda mejora, denominada CAPAC (*Control Avoidance using Proportional Adaptive Control*), consigue reducir las oscilaciones de ERS en torno al valor teórico

equitativo, al tiempo que mantiene un tiempo de respuesta rápido cuando ERS se encuentra lejos de este valor. Para ello, adapta R_{up} y R_{dn} en función de $z - 1$ (Martínez y otros, 1998).

4.4.3 Algoritmos de conmutador con cálculo exacto de las tasas equitativas

Los algoritmos incluidos en esta categoría, como su nombre implica, intentan calcular directamente el valor de la tasa equitativa *max-min* deducido por Charny (1994) de forma distribuida, a partir del conocimiento del ancho de banda disponible y de información de estado por conexión. Para llevar a cabo este cómputo, el conmutador debe crear una tabla indexada por conexión donde poder almacenar información sobre el estado de la conexión, lo cual permite ordenar aumentos agresivos de tasa y aun así alcanzar el régimen estacionario sin la aparición de oscilaciones. Por otra parte, la complejidad de implementación aumenta, aunque depende de la memoria de almacenamiento y del número de divisiones en coma flotante que se necesite en cada algoritmo en particular.

4.4.3.1 *Explicit Rate Indication for Congestion Avoidance* (ERICA)

Al igual que en CAPC, el algoritmo ERICA (Jain y otros, 1995) toma como variable de control el factor de carga z. El valor objetivo de utilización de la capacidad disponible para ABR, denominado *Target Utilization* (TU) toma unos valores típicos de 0.9 ó 0.95. Se toma como punto de trabajo aquél en el que $z = 1$, por lo que ERICA constituye otro ejemplo de mecanismo de *congestion avoidance*.

Sin embargo, a diferencia de CAPC, el algoritmo ERICA, además de utilizar el factor de carga como indicador de congestión, tiene en cuenta de forma separada cuál debería ser la tasa equitativa de cada conexión, independientemente del grado de congestión que experimenta el puerto. Para ello, se estima la tasa equitativa a partir de la expresión:

$$FairShare \leftarrow \frac{Capacidad\ ABR}{Numero\ de\ conexiones\ activas}$$

donde $Capacidad\ ABR \leftarrow Target\ Utilization \cdot Capacidad\ disponible$.

El conmutador permite que toda fuente emita al valor *FairShare*, pero si la fuente no usa toda la tasa *FairShare* que se le permite, entonces el conmutador asignará la capacidad no utilizada entre las fuentes que sí podrían utilizarla. Para ello, el conmutador calcula el valor:

$$VCShare \leftarrow \frac{CCR}{z}$$

Obsérvese que, si todas las conexiones ajustasen sus tasas de emisión al valor *VCShare*, se conseguiría, al final del ciclo siguiente de realimentación, alcanzar el punto de operación, esto es, $z = 1$. En resumen, por una parte *VCShare* conduce al sistema a un punto de operación eficiente, que no necesariamente ha de ser equitativo, mientras que *FairShare* garantiza que como mínimo todas las conexiones obtienen su parte equitativa de ancho de banda, aunque posiblemente conduzcan transitoriamente al sistema a una situación de

sobrecarga. ERICA combina ambos aspectos de la siguiente manera:

$$ER\ Calculated \leftarrow \max(FairShare, VCShare)$$

El comportamiento del bucle de control resultante es el siguiente: durante el primer bucle de realimentación, a cada fuente se le permite emitir al valor *FairShare*, lo cual intenta garantizar la equidad. En los sucesivos bucles, si el valor *VCShare* de una conexión es mayor que *FairShare*, a la fuente se le permite emitir a la tasa *VCShare*, consiguiendo así, por un lado, maximizar la utilización del enlace y, por otro, que las conexiones no limitadas en el nodo evolucionen hacia sus valores de tasa equitativa *max-min*.

Por supuesto, el valor empleado como realimentación por tasa explícita nunca podrá ser mayor que *ABR Capacity*, de modo que:

$$\text{ERS} \leftarrow \min(ER\ Calculated, Capacidad\ ABR)$$

ERICA ofrece las siguientes ventajas: en primer lugar, no depende del ajuste preciso de parámetros; en segundo lugar, los valores de tasa permitida convergen con rapidez y sin presencia de oscilaciones. Por otro lado presenta como inconvenientes la aparición de situaciones de inequidad. Nótese que la distribución de ancho de banda excedente entre las conexiones que se consigue con *VCShare* es en principio equitativa en el sentido *max-min*, puesto que se reparte en proporción a la demanda de cada conexión. No obstante, existen situaciones en las que no se obtiene una distribución de tasas permitidas equitativas *max-min*. En particular, se ha detectado que ocurre cuando el sistema evoluciona a una situación con las siguientes condiciones:

- el factor z alcanza el valor 1;
- quedan algunas conexiones limitadas en otros conmutadores situados antes en el trayecto;
- el resto de conexiones están emitiendo a una tasa mayor que *FairShare*.

En tal caso, el sistema no evoluciona, porque el valor CCR/z es mayor que *FairShare* para aquellas conexiones no limitadas en ningún nodo. Esta situación resultante puede o no ser equitativa *max-min*.

4.5 Conclusiones

La clase de servicio ABR se fundamenta en la operación de un mecanismo de control de flujo por realimentación. Esta realimentación puede ser binaria o por valor explícito, pero no incluye valor de ocupación/asignación de espacio de almacenamiento, sino únicamente valor de tasa asignada por la red. El algoritmo de ajuste en los sistemas finales y el formato y contenido de la señal de realimentación han sido normalizados por ATM Forum, pero los mecanismos específicos de generación de la señal de realimentación de tasa se han dejado a discreción del implementador. Estos mecanismos, junto con la

planificación de células y la gestión de los *buffers*, estudiados en el capítulo 3, constituyen los cuatro mecanismos necesarios para el soporte de servicios *best-effort*. El ajuste en los sistemas finales se implanta en el acceso a la red, esto es, en los sistemas finales, mientras que el resto se implantan en el núcleo de la red, esto es, en los nodos.

Las redes de telecomunicación en general se construyen con el objetivo de soportar la evolución tecnológica a medio y largo plazo. En concreto, los operadores de redes necesitan que los equipos de la red troncal incorporen una tecnología particularmente estable: es impensable correr el riesgo que supone actualizar frecuentemente el núcleo de una red operativa o dejar de soportar un tipo determinado de servicios. O, dicho de otro modo, cualquier nuevo servicio que para su introducción suponga una modificación importante del núcleo de la red es firme candidato a sufrir largos retrasos; no ocurriría así si se plantea una actualización en el acceso a la red.

Si trasladamos esta tesis al soporte de servicios *best-effort* extraemos las siguientes conlusiones:

- Si bien los algoritmos de planificación adoptan diversas apariencias e implementaciones, todos aquellos algoritmos que permiten asignar discriminadamente fracciones de tasa de servicio muestran importantes similitudes (Stiliadis y Varma, 1996), de modo que la forma exacta de planificación puede considerarse menos importante que la existencia de la misma: este es pues un elemento estable de la red.

- Cambiar el formato y el contenido de la señal de realimentación supondría una modificación fundamental en la concepción de diseño de la red y, por tanto, es poco deseable; por tanto, éste es otro elemento estable.

- La gestión de los *buffers* debe tener en cuenta los retardos de realimentación en el bucle de control así como las tasas permitidas de emisión de cada conexión; por tanto, las asignaciones de *buffers* a cada conexión dependerá de la eficiencia de los mecanismos de estimación del producto retardo por ancho de banda y de las características estadísticas del tráfico. Es esperable pues que la política de gestión de los *buffers* evolucione, aunque ello no afectaría negativamente a los usuarios.

- Si se partiese de que la red es resistente y no asume ningún comportamiento determinado en los usuarios de la red, el usuario podría, en principio, modificar los parámetros del bucle de control —tales como el coeficiente de aumento de la tasa de emisión— o incluso los mismos algoritmos de ajuste en el sistema final. Por ejemplo, nuevos mecanismos de ajuste en los sistemas finales podrían mejorar la estimación del estado de la red incorporando algoritmos de predicción. En conclusión, esta resistencia permite y favorece la experimentación y la evolución de los mecanismos de ajuste.

Dado que los algoritmos de ajuste en el bucle de control son los más susceptibles de sufrir una evolución técnica, estos algoritmos deberían implementarse únicamente en el acceso a la red. Además, los conmutadores deberían proporcionar exlusivamente la señal de realimentación, pero no participar directamente en los cálculos del bucle de control. Sí

participan en el control, en cambio, los conmutadores en aquellos esquemas de control de flujo *hop-by-hop*, así como en aquellos esquemas que descansan en que los conmutadores modifiquen la señal de realimentación según consideraciones de control. Un ejemplo de cómo el conmutador queda involucrado en el ajuste necesario en un mecanismo de control de flujo es el normalizado para ABR, en el que la señal de realimentación es exclusivamente un valor de tasa, lo cual obliga a que los conmutadores tengan que modificar el valor realimentado de tasa teniendo en cuenta además la ocupación de los *buffers*, lo cual consituye una acción que atañe a la funcionalidad del ajuste, no a la de generación y envío de realimentación.

Así, el mecanismo de control de flujo en ABR incorpora parte de la funcionalidad del ajuste en los nodos, lo cual hipoteca la tecnología del núcleo de la red. Si se hubiese limitado la funcionalidad de los nodos en el control de flujo a generar una señal de realimentación de tasa y de cola explícitas, el algoritmo de ajuste en los sistemas finales hubiera podido evolucionar y mejorar manteniendo intacta la tecnología de los nodos de la red (Lefelhocz y otros, 1996).

En otro orden de cosas, en la sección 4.4 se ha comprobado cómo la totalidad de los mecanismos de generación de señal de realimentación asumen que la planificación se realiza mediante el algoritmo FCFS. Si bien esta decisión de implementación no impide que la señal de realimentación sea individualizada, sí comporta otra consecuencia. La utilización de la disciplina FCFS impide que el conmutador monitorice la ocupación de los recursos por parte de cada conexión. Algunos algoritmos, tal como ERICA, consiguen determinar la actividad/inactividad de cada conexión, pero ninguno gestiona individualizadamente los *buffers*, ni determina cuál es la fracción de ancho de banda que una conexión está utilizando. A causa de ello, la medición de la ocupación de los recursos en el nodo, esto es, espacio de almacenamiento y ancho de banda, es una medición agregada, lo cual impide que se proteja a los usuarios. El servicio ABR que se proporciona no establece por tanto garantías relativas, las cuales identificamos en la sección 2.4 como uno de los puntos irrenunciables en el soporte de servicios *best-effort* sobre red ATM. Se ha visto en el capítulo 3 que los algoritmos de planificación equitativa protegen los recursos que utiliza cada uno de los usuarios y además los asigna de forma equitativa. En el capítulo 5, se describe una propuesta de soporte de servicio ABR basada en la utilización de algoritmos de planificación equitativa en los nodos.

Capítulo 5

Propuesta de un algoritmo de conmutador basado en planificación equitativa

El algoritmo de conmutador que propone esta Tesis se basa en la utilización de algoritmos de planificación equitativa. El mecanismo de generación de la señal de realimentación realiza una estimación de la asignación de ancho de banda efectuada por el algoritmo de planificación equitativa. El control de flujo resultante ofrecerá un reparto de tasas de emisión que será equitativo, pues lo es la asignación de ancho de banda. Es más, el criterio de equidad de este reparto de tasas de emisión va a venir determinado por el criterio de equidad del algoritmo de planificación. Extenderemos, entonces, los criterios de equidad del servicio ABR a los criterios *max-min* ponderado y al criterio *max-min* ponderado con garantía de MCR, dando lugar a lo que denominamos un *servicio ABR generalizado*.

Se trata de una propuesta original que parte de una ideas apuntadas por Lyles y Lin (1994b). A diferencia de los algoritmos de conmutador propuestos en la literatura, que asumen algoritmos FCFS en los nodos, el algoritmo propuesto en esta Tesis utiliza algoritmos de planificación equitativa. Ello va a permitir, por un lado, garantizar el ancho de banda que se realimenta a las fuentes, y por otro, ofrecer un criterio de equidad más genérico que el criterio *max-min*, que es el que ofrecen los algoritmos propuestos en la literatura.

En la sección 5.1, se describe a grandes rasgos las características del mecanismo de control de flujo. En la sección 5.2, se describe con más detalle las características de los algoritmos de planificación y de gestión de *buffers* incorporados. El mecanismo de generación de la señal de realimentación, principal contribución de esta Tesis, se presenta en la sección 5.3 y se analiza en la sección 5.4. Finalmente, en la sección 5.5 se propone los mecanismos de soporte de un servicio ABR generalizado.

5.1 Descripción general del mecanismo de control de flujo

En este capítulo se describe una propuesta de soporte de servicio *best-effort* en redes ATM cuyas características son las siguientes. En primer lugar, respeta el modelo de servicio ABR definido por *ATM Forum Traffic Management 4.0*, que se ha descrito en la sección 2.3.3, e incorpora mecanismos adecuados para la provisión del paradigma de servicio *best-effort* descrito en la sección 2.4.

En segundo lugar, adopta los mecanismos de soporte del servicio ABR especificados en *ATM Forum Traffic Management 4.0*. Más concretamente, incorpora los siguientes elementos:

- el control de flujo es por control en bucle cerrado por conexión y por ajuste de tasa de emisión;
- el formato y significado de la señal de realimentación, según la célula RM especificada (véase la sección 4.2.1);
- el mecanismo de ajuste de tasa en la fuente, según se especifica (véase la sección 4.3.2);
- el mecanismo de marcado es por tasa explícita, según se especifica (véase la sección 4.4);
- los comportamiento de fuente, de destino y de conmutador según se especifica.

En tercer lugar, la propuesta adopta como algoritmo de planificación de la transmisión de las células un algoritmo de planificación equitativa, bien los algoritmos de prioridad ordenada WFQ, SCFQ o WF^{2}Q (véase las secciones 3.3.2 y 3.3.3), bien el algoritmo enmarcado WRR (véase la sección 3.3.4).

En cuarto lugar, el algoritmo de gestión de espacio de almacenamiento realiza la asignación de *buffers* según FCFU y el descarte de paquetes por ordenación del algoritmo de planificación.

Finalmente, el mecanismo de generación de la señal de realimentación es el que describe la sección 5.3.

5.2 Elección de los algoritmos de planificación de recursos

Asumimos una arquitectura de conmutador en la que la matriz de conmutación no provoca bloqueo y en la que la planificación se realiza exclusivamente en los puertos de salida. Los conmutadores adoptan por tanto una disposición de colas a la salida. A continuación se describen los condicionantes que el mecanismo de soporte presentado en la sección 5.1 impone sobre los algoritmos de planificación de recursos, a saber, el algoritmo de planificación y el algoritmo de gestión de *buffers*.

5.2.1 Algoritmo de planificación

Un mecanismo de control de flujo que soporte servicio ABR requiere de un algoritmo de planificación que cumpla los siguientes requisitos. En primer lugar, el algoritmo de planificación deberá asignar el ancho de banda de forma equitativa en el sentido *max-min* (véase la página 38), porque el algoritmo de conmutador estima *grosso modo* la asignación del ancho de banda que se lleva a cabo en la planificación de la transmisión. Así se consigue que la asignación de ancho de banda a escala local sea equitativa *max-min*. Como el mecanismo de control de flujo realimenta a cada fuente la mínima asignación local equitativa *max-min* efectuada a lo largo del trayecto de la conexión, se consigue una asignación de ancho de banda equitativa *max-min* a escala global, que es el objetivo deseable del control de flujo para el soporte ABR. Los algoritmos de planificación equitativa cumplen este requisito.

En segundo lugar, el algoritmo de planificación deberá conseguir protección entre conexiones. La asignación de ancho de banda que deciden los algoritmos de planificación equitativa son, en primera aproximación, independientes del patrón de llegada de células de cada conexión.

Consideremos la situación en que el número de conexiones que envían células a un conmutador es constante y que cada conexión siempre dispone de células que transmitir en el momento en que se decide su tranmisión. En un conmutador con planificación FCFS, si una conexión envía una ráfaga larga de células, todas las conexiones verán reducida efectivamente la tasa observada de servicio mientras la ráfaga permanece en el conmutador. Por el contrario, en un conmutador con planificación equitativa, el resto de conexiones no se verá afectado.

No obstante lo anterior, con planificación equitativa la tasa asignada de servicio puede variar por dos razones: porque el número de conexiones establecidas cambie —nótese que la tasa asignada a cada conexión por un algoritmo de planificación equitativa es inversamente proporcional al número de conexiones activas—, o bien porque una conexión emita a una tasa menor que la tasa asignada por el algoritmo de planificación —en tal caso, el planificador equitativo trata a la conexión como inactiva, aumentando durante ese intervalo la tasa asignada al resto de conexiones—. Esta circunstancia también se da cuando la fuente emite según un patrón esporádico.

Finalmente, se exige que la realización del algoritmo de planificación posibilite el acceso a la primera célula de cada conexión. Más concretamente, la estructura de datos empleada para la ordenación de las células en el puerto del conmutador debe permitir detectar el instante en que una célula pasa a ser la próxima célula de una conexión dada que se transmitirá.

Los algoritmos WFQ, SCFQ y WRR cumplen los requisitos anteriores.

5.2.2 Algoritmo de gestión de *buffers*

El mecanismo de generación de realimentación propuesto no impone ningún requisito específico sobre la gestión del espacio de almacenamiento en el nodo. No obstante, del análisis del papel que desempeña dentro del control de flujo que se ha llevado a cabo en

la sección 3.4, hemos decidido que la gestión del espacio de almacenamiento en el nodo se efectúe por conexión. A continuación, describimos los detalles del algoritmo de gestión de los *buffers*.

Por un lado, la asignación de los *buffers* no impone ninguna restricción a las conexiones. Se asignan a medida que se solicitan. Cuando se ocupa la totalidad del espacio de almacenamiento, se descarta aquella célula que, en caso de no producirse el descarte que se acomete ni la llegada de ninguna otra célula, sería transmitida en último lugar. Para los algoritmos de prioridad ordenada ello se consigue descartando la célula con menor prioridad, esto es, con mayor marca temporal. En cambio, para los algoritmos enmarcados, descartando la última célula de la cola con mayor tamaño normalizado al peso de la conexión. En el caso de que los pesos de todas las conexiones sean iguales, el algoritmo de descarte se simplifica, pues en ambos casos consistiría en eliminar el último paquete recibido de la conexión que más *buffers* está ocupando. Nótese que se consigue de este modo una asignación de *buffers* equitativa en el sentido *max-min*.

Si el algoritmo de planificación es del tipo prioridad ordenada, cuando se descarta una célula de una conexión, ésta puede computarse en la asignación de ancho de banda de la conexión a la cual pertenece, o bien no computarse. Se ha optado por no computarlo, para lo cual, cuando se descarta una célula perteneciente a la conexión i, se ajustan los valores S_i y F_i. Así, no se penaliza el *throughput* observado por la conexión que ya sufre las pérdidas.

Finalmente, nótese que el algoritmo de asignación/descarte presentado no respeta la asignación de *buffers* decidida en intervalos de realimentación anteriores, es decir, cuando se acepta una célula en el conmutador, su permanencia en el mismo dependerá de la evolución de los recursos disponibles en el conmutador hasta que el momento en el que se decide su transmisión. Es decir, el conmutador no adopta ningún compromiso por el hecho de asignar un *buffer* la célula de una conexión.

5.3 Descripción del mecanismo de generación de la señal de realimentación por tasa explícita

5.3.1 Algoritmo de cálculo de la tasa equitativa

El algoritmo de cálculo de la tasa equitativa que se realimenta por tasa explícita a la fuente se basa en el cómputo de una magnitud temporal que denominamos *tiempo "head-of-line"*, o tiempo HOL, y que se define a continuación, primero para el caso ideal de un sistema con disciplina PS y segundo, para un sistema real con planificación equitativa, que denominaremos *sistema FQ/RR*.

5.3.1.1 Estimación de la tasa equitativa bajo planificación PS

Un sistema PS puede concebirse, sin pérdida de generalidad, como un subsistema de espera con colas separadas para cada uno de los flujos de tareas que se identifiquen, más un subsistema de servicio que da servicio simultáneamente a aquellas tareas que son las

primeras de cada flujo. Asumimos por simplicidad en la exposición que la fracción de recurso que obtiene cada flujo con tareas en el subsistema de servicio es la misma.

Definimos *tiempo HOL de una tarea en un sistema PS* como el tiempo que emplea la tarea en el subsistema de servicio. En otras palabras, el tiempo que transcurre desde que llega a la cabeza de su cola hasta que abandona el sistema. Dado que sólo puede haber una tarea por flujo en el subsistema de servicio y que cuantas más tareas haya en el subsistema de servicio, menor fracción de recurso obtendrá, podemos afirmar lo siguiente:

> El tiempo HOL de una tarea en un sistema PS es un indicador del número de flujos que han estado compartiendo el subsistema de servicio durante ese intervalo de tiempo

Denominaremos flujo *activo* a aquel que dispone de una tarea en el subsistema de servicio del sistema PS. Esta definición es coherente con la denominación de usuario activo en la sección 3.3.1.

Si empleásemos un sistema PS en los puertos de salida de un conmutador ATM, las tareas serían células ATM cuya transmisión habría que planificar y los flujos de tareas serían conexiones ATM, de modo que la afirmación anterior se convierte en:

> El *tiempo HOL de una célula ATM en un sistema PS* es un indicador del número de conexiones ATM activas en el puerto

Además, dado que las células ATM son paquetes de tamaño fijo, desde el punto de vista del sistema PS, todas las células ATM tienen las mismas necesidades de servicio.

Según una aproximación más formal, podemos transformar la última afirmación en la siguiente:

> Sea un puerto ATM con planificación PS cuya capacidad de servicio es de r células por unidad de tiempo y en el que, durante un intervalo $(\tau, t]$ no varía el número de conexiones activas, que denominaremos N_{ac}; podemos afirmar que el tiempo HOL de una célula ATM que inicie y finalice su transmisión dentro del intervalo $(\tau, t]$ es igual a
>
> $$t_{HOL}^{PS} = \frac{N_{ac}}{r}$$

Evidentemente, un puerto de salida de un conmutador ATM no es realizable con un sistema PS, pues el servicio que demandan las células ATM no puede satisfacerse por compartición del subsistema de transmisión. No obstante, proponemos que el tiempo HOL computado por un sistema PS en el puerto de un conmutador ATM sea la señal de realimentación para controlar el flujo de conexiones ATM de la categoría de servicio ABR. Para ello, en primera aproximación, se estimaría la tasa equitativa de una conexión ABR como la inversa del tiempo HOL de sus células.

Si eliminamos la simplificación anterior de igual prioridad en la asignación de ancho de banda a cada conexión, el sistema pasa a ser GPS y las conexiones ATM llevan asociado un número real positivo $\phi_1, \phi_2, \ldots, \phi_N$; podemos afirmar entonces que:

En un sistema GPS con capacidad de servicio r células por unidad de tiempo, en el que durante el intervalo $(\tau, t]$ las conexiones no modifican su estado de actividad/inactividad, el tiempo HOL de una célula ATM que inicie y finalice su transmisión dentro del intervalo $(\tau, t]$, es igual a

$$t_{HOL}^{GPS} = \frac{\sum_{j\ activa} \phi_j}{\phi_i} \cdot \frac{1}{r}$$

donde i es la conexión a la que pertenece la célula ATM considerada

Las afirmaciones anteriores no permiten conocer, en todas las situaciones, la relación exacta entre el tiempo HOL de una célula ATM en un sistema PS y la tasa equitativa de la conexión en el sentido *max-min*. Unicamente en el caso de que todas las conexiones estén activas en el nodo bajo consideración, el tiempo HOL de las células de cualquiera de tales conexiones será N_{ac}/r, que es efectivamente igual a la inversa de la tasa equitativa *max-min* de las conexiones.

Analicemos cuál es la relación entre t_{HOL} y la tasa equitativa *max-min* en otras situaciones. Para ello, consideramos un escenario ideal de provisión de servicio ABR, en el que la planificación en los conmutadores es PS y en el que las fuentes emiten persistentemente a la máxima tasa posible. En este escenario ideal, en régimen estacionario se habrá conseguido una utilización del ancho de banda disponible del 100%, pero no todas las conexiones estarán activas continuadamente en todos los nodos. Veamos por qué.

Introduzcamos el concepto de *nodo estrangulador* de una conexión ABR con la ayuda de una definición recursiva. El primer conmutador del trayecto de una conexión ABR se denomina nodo estrangulador. Consideremos a continuación el resto de conmutadores en el trayecto hasta el destino de la conexión. Denominamos nodo estrangulador a uno de tales conmutadores si asigna una tasa menor que algún nodo estrangulador que lo preceda en el trayecto de la conexión. A partir de la definición de nodo estrangulador, es inmediato definir el *nodo de cuello de botella* de una conexión ABR como el nodo estrangulador más próximo al destino de la conexión.

Por definición de conexión activa, podemos afirmar que una conexión estará activa de forma continuada únicamente en aquellos nodos que la estrangulen. Además, un conmutador que sea nodo estrangulador de una conexión puede no ser estrangulador de otra conexión. En particular, no todas las conexiones tienen su nodo de cuello de botella en el mismo conmutador.

En el escenario ideal descrito, la tasa estimada en un conmutador para una conexión ABR a partir del tiempo HOL no dará como resultado un valor igual a la capacidad de servicio dividido por el número total de conexiones. Ello sólo ocurriría si todas las conexiones ABR tuviesen los mismos nodos estranguladores; por ejemplo, en el caso de una red de dos conmutadores conectados por un enlace troncal, el primer conmutador es el cuello de botella de todas aquellas conexiones que atraviesan ambos conmutadores.

Así pues, en un conmutador dado pueden existir conexiones ABR estranguladas por el conmutador pero también conexiones estranguladas por otros conmutadores que le precedan o que le sucedan en el trayecto de las conexiones. Ello conlleva que aquellas conexiones estranguladas por otros conmutadores no puedan alcanzar un estado de actividad

continuado en el conmutador considerado. Al respecto del tiempo HOL, el valor medido en cada momento para las células de cualquier conexión dependerá de la eventual presencia de células en el conmutador pertenecientes a las conexiones no estranguladas. Es decir, si N es el número total de conexiones que atraviesan el conmutador y M de ellas no están estranguladas en el mismo, el valor t_{HOL} de las células de una conexión puede tener un valor entre $N \cdot 1/r$ y $(N - M) \cdot 1/r$. Para resolver la ambigüedad en la medida, proponemos que se realice un promediado de los valores medidos de tiempo HOL para las células de cada conexión, de modo que obtendríamos un valor proporcional al número, esta vez efectivo, de conexiones activas en el sistema. Por tanto se estimaría la tasa equitativa de una conexión ABR como la inversa del promedio del tiempo HOL de las células de la conexión.

5.3.1.2 Grado de agregación de la estimación de la tasa equitativa

Supongamos, manteniendo las premisas del escenario ideal descrito en la sección anterior, una situación simplificada, en la que todas las conexiones, a excepción de una de ellas, están estranguladas en un único conmutador. Entonces, el tiempo HOL computado para la conexión no estrangulada será igual a $N1/r$, pues cada vez que llega una de sus células, el número de conexiones activas pasa de $N - 1$ a N. Por su parte, el tiempo HOL promediado para cualquiera de las conexiones estranguladas será un valor entre $N1/r$ y $(N - 1)1/r$, en función del grado de actividad de la fuente no estrangulada. Además, este valor cumple que:

- es el mismo para cualquiera de las conexiones estranguladas;
- la suma de la inversa de los valores computados para las conexiones estranguladas es igual al ancho de banda no utilizado por la conexión no estrangulada.

Este valor es, por tanto, el valor de tasa equitativa *max-min* para las conexiones estranguladas.

En una situación con más de una conexión no estrangulada en el conmutador, la afirmación respecto de la conexión no estrangulada ya no es cierta, aunque sí son ciertas las consideraciones cualitativas respecto de las conexiones estranguladas.

Así pues, podemos establecer las siguientes características del procedimiento de estimación de tasa equitativa mediante el promedio del tiempo HOL de cada conexión:

- Cuando una conexión está estrangulada en el nodo, el valor estimado de tasa equitativa es el valor de tasa equitativa en el sentido *max-min*.
- Cuando una conexión no está estrangulada en el nodo, sólo podemos afirmar que:
  - el valor mínimo de la estimación de tasa equitativa para la conexión se da cuando el resto de conexiones establecidas se estrangulan en el conmutador;
  - el valor mínimo de la estimación de tasa equitativa para cualquier conexión es igual al cociente entre el ancho de banda disponible y el número de conexiones establecidas;

– el valor estimado de tasa equitativa para una conexión no depende del grado de actividad ni del patrón de emisión de la propia conexión.

Estamos en condiciones pues de afirmar que el promedio del tiempo HOL de las células de una conexión estima la tasa equitativa en el sentido *max-min*, para las conexiones estranguladas en el conmutador, mientras que para aquellas conexiones no estranguladas estima un valor mayor o igual que la tasa equitativa *max-min* que obtendría en caso de estar estrangulada. Este comportamiento es deseable en un mecanismo de control de flujo por realimentación de tasa, puesto que, primero, para aquellas conexiones que demandan más de lo que un algoritmo de planificación equitativo en el sentido *max-min* les asigna —se trata de las conexiones estranguladas—, genera una señal de realimentación igual al ancho de banda asignado. Y segundo, para aquellas conexiones que demandan menos que lo que un algoritmo de planificación equitativo en el sentido *max-min* les asignaría —se trata de las conexiones no estranguladas—, genera una señal de realimentación igual al ancho de banda que les asignaría. Así pues, la evolución de un mecanismo tal es hacia una distribución global de ancho de banda equitativo en el sentido *max-min*.

Las consideraciones hechas hasta este punto nos van a permitir decidir el grado de agregación que debe tener la señal de realimentación. Hemos visto que el valor de tasa equitativa estimado por un conmutador a partir del promedio del tiempo HOL es el mismo para todas las conexiones estranguladas en un puerto de salida. No ocurre así para el valor promediado de tiempo HOL para una conexión no estrangulada, cuando existe más de una conexión no estrangulada en ese conmutador. No parece indicado, pues, que el promedio del tiempo HOL de las células se efectúe agregadamente para todas conexiones. Otros algoritmos, tales como EPRCA o CAPC, computan agregadamente un valor de tasa equitativa: en EPRCA, se trata del valor MACR, a partir del valor CCR de todas las conexiones; en CAPC, se trata del valor ERS, a partir del factor de carga agregada.

Así pues, el procedimiento de estimación de tasa equitativa que proponemos necesariamente ha de realizarse por conexión.

5.3.1.3 Filtrado de la estimación de tasa equitativa

La situación real de funcionamiento de las conexiones ABR es intrínsecamente transitoria, tanto por el hecho de que el ancho de banda disponible para ABR es variable por definición, como porque el retardo de realimentación en el control de flujo ocasiona transitorios en la evolución del sistema. Estos transitorios pueden dar lugar a situaciones tanto de sobrecarga como de infrautilización. El procedimiento de estimación de tasa equitativa debe ser válido aun en los frecuentes casos en que no se haya alcanzado el régimen estacionario.

En el caso de que el transitorio se origine por una variación en el ancho de banda disponible en un enlace troncal, se puede producir un desplazamiento de los nodos estranguladores, tanto si se trata de un aumento como de una disminución de ancho de banda. Una conexión que pasa a estar estrangulada en un conmutador en donde no lo estaba pasa a estar activa continuadamente en el conmutador; por el contrario, una conexión que deja de estar estrangulada en un conmutador en donde sí lo estaba deja de estar activa conti-

nuadamente. Por otro lado, el retardo de realimentación ocasiona que la red evolucione desde una disposición inicial de nodos estranguladores a una disposición final a través de una serie de disposiciones transitorias de nodos estranguladores.

Por último, el régimen transitorio puede estar ocasionado por el patrón de emisión de células de cada conexión. Si el patrón de emisión de células deja de ser persistente, la actividad/inactividad de las conexiones en cada nodo será una característica muy variable. Ello provoca errores en el procedimiento propuesto de estimación de tasa equitativa, que pueden ser aleatorios o sistemáticos. En el caso de que sean aleatorios, en cada medida de tiempo HOL el número de conexiones activas sería diferente. En este caso, si promediamos las medidas de tiempo HOL, resultará un número efectivo de conexiones activas.

Por el contrario, se puede producir una sincronización de eventos de llegada de células de distintas conexiones que ocasione una estimación sistemáticamente errónea del número efectivo de conexiones activas. Estos errores de medida pueden suponer bien una sobreestimación, bien una subestimación de la tasa equitativa. En el primer caso, esta sobreestimación de tasa equitativa se traduciría, tras el correspondiente retardo de realimentación, en una sobrecarga en el conmutador. Esta sobrecarga aumentaría el grado de actividad de las conexiones, con lo cual disminuiría la probabilidad de estimar menor número efectivo de conexiones activas que el número efectivo real. En el segundo caso, la subestimación de tasa equitativa, simétricamente al efecto anterior, conllevaría una disminución del grado de actividad de las conexiones, lo cual también disminuiría la probabilidad de estimar mayor número efectivo de conexiones activas que el número efectivo real.

Las apreciaciones anteriores muestran la importancia que tiene la elección de la técnica de promediado con el objeto de atenuar el efecto que, sobre la estimación de tasa equitativa, tienen los transitorios ocasionados por la red y por las fuentes. A continuación describimos nuestra propuesta de filtrado del valor medido de tiempo HOL.

Se trata de un filtrado de las medidas de tiempo HOL, no de la inversa de las medidas; veamos por qué. La inversa del tiempo HOL es una medida de tasa instantánea, por lo que parece razonable a priori promediarlas. Sin embargo, una tasa instantánea es en realidad la razón de dos cantidades: el número de células y el tiempo transcurrido. Para promediar consistentemente una razón de dos cantidades deberíamos primero sumar los numeradores y dividir el resultado por la suma de los denominadores. Para el caso que nos ocupa, el resultado es sumar 1 tantas veces como muestras tomamos y dividir por la suma de los tiempos HOL calculados en cada muestra. Nótese que es equivalente a calcular la media aritmética de los tiempos HOL y calcular la inversa del resultado. Este argumento ya ha sido expuesto en la sección 4.4.2.1, cuando se describía el algoritmo EPRCA.

Hemos escogido un filtrado FIR mediante ventana rectangular. El tamaño de la ventana es igual al número de medidas obtenidas para cada conexión entre dos células RM, es decir, un tamaño de N_{rm} células. No obstante, el valor de salida del filtro no se computa tras la llegada de cada célula, sino únicamente tras la de la N_{rm}-ésima célula. De este modo, no se precisa almacenar N_{rm} medidas de tiempo HOL por cada conexión, sino sólo un acumulador y un contador por cada conexión.

El valor de salida del filtro se computa en el instante en que llega una célula RM de vuelta al conmutador. Esta decisión de implementación tiene dos consecuencias. En pri-

mer lugar, el valor de estimación que se utiliza para la realimentación es el más reciente. No ocurriría así necesariamente si el valor de salida del filtro se computase en el instante en el que llega una célula RM de ida al conmutador. En segundo lugar, dado que ni el número ni la cadencia de llegada de células RM de ida son necesariamente iguales a los de células RM de vuelta (véase la sección 4.3.2), el filtrado puede no seguir el comportamiento esperado. Sí lo son, no obstante, bajo condiciones de operación normales, tales como las que se asumen en el capítulo 6.

5.3.1.4 Estimación de la tasa equitativa en un sistema FQ/RR

Al igual que se hizo en la sección 5.3.1.1 para un sistema PS, un sistema FQ/RR puede concebirse, sin pérdida de generalidad, como un subsistema de espera con colas separadas para cada uno de los flujos que se identifiquen, más un subsistema de servicio que planifica la ejecución de *una* tarea de entre todas las primeras de cada flujo.

Encontramos dos diferencias fundamentales entre un sistema FQ/RR y un sistema PS. En primer lugar, el sistema PS determina implícitamente el orden de ejecución de las tareas, mientras que un sistema FQ/RR planifica explícitamente la ejecución de las tareas. En el caso de WFQ, el orden se determina a través del cálculo del instante de tiempo virtual en que cada tarea abandonaría un sistema GPS sometido a la misma carga de trabajo. En segundo lugar, en un sistema PS las tareas nunca pueden llegar a la cabeza de su cola antes de que su tarea predecesora haya completado servicio, puesto que el subsistema de servicio contiene a todas las tareas de cabeza de cola. Por el contrario, en un sistema FQ/RR sí, pues el subsistema de servicio sólo contiene a la tarea que en ese instante está recibiendo el servicio.

Esta segunda diferencia entre PS y FQ/RR obligaría a redefinir el término tiempo HOL de una tarea en un sistema FQ/RR. Sin embargo, continuaremos denominando *tiempo HOL de una tarea en un sistema FQ/RR* al tiempo empleado por la tarea en el subsistema de servicio. Ahora bien, redefiniremos subsistema de servicio de un sistema FQ/RR como el conjunto de tareas de cabeza de cola más la tarea que esté recibiendo servicio menos la tarea de cabeza de cola sucesora de esta última tarea. Así, una célula ATM presente en un sistema FQ/RR se encuentra en el subsistema de espera o en el de servicio según las consideraciones siguientes:

1. si una célula llega al sistema FQ/RR y no hay ninguna célula de su misma conexión en espera ni siendo transmitida, consideramos que en ese instante entra en el subsistema de servicio;

2. si una célula llega al sistema FQ/RR y no hay ninguna célula de su misma conexión en espera pero la célula que está siendo transmitida sí es de su misma conexión, consideramos que aún no ha entrado en el subsistema de servicio y, por tanto, permanece en el subsistema de espera;

3. si una célula llega al sistema FQ/RR y hay una célula de su misma conexión en espera, consideramos que permanece en el subsistema de espera;

4. una célula pasa del subsistema de espera al subsistema de servicio cuando su célula predecesora abandona el subsistema de servicio;

5. una célula que está siendo transmitida por el sistema FQ/RR abandona el subsistema de servicio cuando acaba de ser transmitida.

Con las consideraciones anteriores, podemos concluir la misma definición para el tiempo HOL de un célula ATM en un sistema FQ/RR que en un sistema PS (véase la sección 5.3.1.1). Proponemos, entonces, **estimar la tasa equitativa de una conexión ABR como la inversa del filtrado del tiempo HOL de las células ATM de la conexión**, las cuales son conmutadas por el conmutador ATM en el puerto de salida mediante planificación FQ/RR.

La exactitud del procedimiento de estimación de la tasa equitativa de una conexión ABR bajo planificación FQ/RR depende principalmente de la equidad en la asignación del ancho de banda por parte del algoritmo de planificación que se emplee. Según se ha analizado en las secciones 3.3.2, 3.3.3 y 3.3.4, el grado de equidad que consigue cada algoritmo FQ/RR es diferente: ello se traduce en que la estimación de la tasa equitativa será más fiable en WFQ que en SCFQ o en WRR.

5.3.2 Algoritmo de control de congestión

Según argumentamos en la página 71, un control de flujo por realimentación que ajuste la tasa de emisión ha de contar con una estimación tanto de la tasa de servicio en el cuello de botella como de la ocupación de la cola en el mismo, para que sea estable y eficiente. Algunos ejemplos de mecanismos de ajuste de tasa que cumplen este requisito son *packet-pair protocol* (Keshav, 1991b) y *hop-by-hop scheme* (Mishra y otros, 1996). En particular, los mecanismos citados proporcionan a la fuente las estimaciones de tasa y de cola, las cuales se emplean en la fuente como parámetros de entrada en el ajuste que llevan a cabo.

En el marco normalizado por el ATM Forum, se contempla la posibilidad de notificar por tasa explícita el valor de tasa de servicio en el cuello de botella, pero no el valor de ocupación de la cola en el mismo. Este hecho fue puesto de relieve por Lyles (1994a) durante la definición de *ATM Forum Traffic Management 4.0*. Ello obliga a intentar satisfacer los requisitos establecidos por Altman y otros (1993) de formas alternativas, las cuales enumeramos a continuación.

Se puede realizar la notificación de la ocupación de la cola mediante un valor binario, utilizando el bit CI de la célula RM. Tal aproximación se toma en el esquema CAPC (véase la sección 4.4.2.2), en donde la activación del bit CI depende de la superación de los correspondientes valores umbrales en el llenado de la cola. Esta alternativa impide un ajuste rápido y eficiente en función del nivel de llenado de las colas. Nótese sin embargo que en CAPC, es el factor de carga el indicador primario de congestión, por lo que en el punto de trabajo del sistema la ocupación de la cola será nula. En otras palabras, la activación del bit CI por llenado de la cola es únicamente un mecanismo accesorio.

Alternativamente, se puede modificar el valor realimentado de tasa en función del nivel de ocupación de la cola. Tal aproximación se toma en el esquema EPRCA (véase la

sección 4.4.2.1), en donde los coeficientes multiplicadores de la estimación MACR dependen de la superación de los correspondientes valores umbrales en el llenado de la cola. También es la aproximación escogida en el esquema ERICA+ (Kalyanaraman, 1997), que, a diferencia de ERICA, ajusta el coeficiente *Target Utilization* en función del nivel de llenado de la cola.

Finalmente, se puede modificar el valor realimentado de tasa con el fin de prevenir el llenado de las colas. Esta alternativa se emplea en el algoritmo ERICA (véase la sección 4.4.3.1), en el que se fija un valor objetivo de ocupación del ancho de banda disponible, materializado en el coeficiente *Target Utilization*, menor que 1, lo cual rerserva de hecho una porción de ancho de banda para el vaciado de las colas tras los transitorios.

En el mecanismo de generación de señal de realimentación propuesto hemos escogido una alternativa similar a la que incorpora el algoritmo ERICA. Se ha realizado mediante la multiplicación de la estimación de tasa equitativa de cada conexión por un coeficiente *Target Utilization*, menor que la unidad. El efecto es el mismo que se perseguía en ERICA: disponer de un ancho de banda de drenaje de las colas de los conmutadores cuando éstas se llenan por los efectos transitorios del control de flujo.

Las razones de la elección propuesta son las siguientes. Al fijar un valor objetivo de aprovechamiento del ancho de banda disponible en cada enlace por debajo de la unidad, pretendemos mantener las colas vacías. De este modo, el punto de trabajo perseguido está cercano al codo en una curva retardo-*throughput* (véase la página 89), con lo que se compromete el *throughput* por debajo del óptimo a cambio de minimizar el retardo que sufre; se trata, pues, de un esquema de *congestion avoidance*. Nótese que, con esta elección, mantenemos los valores de retardo de ida y vuelta en valores mínimos, lo cual optimiza el rendimiento obtenible por los protocolos de capas superiores que descansan sobre mecanismos propios de control de flujo. Además, si las colas permanecen vacías, el retardo de ida y vuelta es menos variable; en particular, el protocolo TCP mejora su rendimiento cuanto menor y menos variable es el retardo de ida y vuelta que estima para sus segmentos.

Podría argumentarse en contra del algoritmo propuesto que la elección de un punto de trabajo entre el codo de la curva retardo-*throughput* y el *cliff* hubiese resultado en un funcionamiento más eficiente, tanto en régimen estacionario como transitorio. Esta es la elección efectuada en el algoritmo ERICA+, en el que se busca un nivel objetivo de llenado de la cola no nulo. Así, en régimen estacionario, mantener las colas parcialmente llenas conduce a una eficiencia del 100%, mientras que la elección propuesta sólo conseguiría una eficiencia igual, en tanto por uno, a *Target Utilization*. En régimen transitorio, por otro lado, cualquier aumento de ancho de banda disponible podría ser aprovechado por las células que se mantienen en la cola del puerto correspondiente.

En contra de las argumentaciones anteriores, aportamos las siguientes consideraciones. En primer lugar, el control que se ejerce a partir de la ocupación de la cola podría hacerse depender exclusivamente del nivel de llenado de la misma, empleando funciones de ajuste por umbral, lineal, hiperbólico, etc.. En tal caso, como ya se argumentó en la página 92, el control que sobre la tasa de emisión se ejerce no sería eficaz: el ajuste de la tasa de emisión sólo garantiza la tasa de crecimiento máxima de la cola, pero nunca el nivel de

llenado máximo. En cambio, el control puede ser eficaz si se basa en la derivada del nivel de llenado de la cola con respecto al tiempo, aunque ello aumenta la complejidad.

En segundo lugar, el aprovechamiento de ancho de banda disponible a una escala temporal menor que el retardo de ida y vuelta sólo es óptimamente eficiente si se conoce exactamente el retardo de realimentación. Sólo de este modo se puede prever la presencia de un número de células en el nodo igual al producto retardo de realimentación por aumento de ancho de banda, que es el valor necesario para utilizar eficientemente el ancho de banda disponible instantáneamente. Sin embargo, el conocimiento de este valor por parte del conmutador no es viable, por la complejidad que añadiría mantener una estimación desde el nodo del retardo de realimentación de la conexiones que lo atraviesan. Además, aun en el caso de conocerse, el nodo debería prever cualquier aumento de ancho de banda disponible; en particular, el aumento máximo. Es evidente que los conmutadores ATM que soportan ABR se han diseñado precisamente para evitar tener que dimensionar sus *buffers* al valor retardo por ancho de banda de la red.

Por último, un punto de trabajo entre el codo y el *cliff* de la curva retardo-*throughput* es intrínsecamente menos estable que en el codo.

Nótese finalmente que, con el algoritmo de control de congestión propuesto, el conmutador gestiona el espacio de almacenamiento con el único objetivo de hacer frente a las situaciones de congestión transitoria. Este hecho es coherente con la restricción apuntada al inicio de esta sección, en el sentido de que en *ATM Forum Traffic Management 4.0* sólo se contempla la realimentación por tasa explícita y no se realimenta la ocupación de la cola. Así también, es justificable que el algoritmo de gestión del espacio de almacenamiento propuesto en la sección 5.2.2 sólo se responsabilice de gestionar el descarte de forma equitativa *max-min* y no de garantizar asignaciones. Las asignaciones de *buffers* no son efectivas, puesto que el usuario no es consciente de ellas, dado que no se le notifican por valor explícito.

Como veremos en la sección 6.3.4, cuando se evalúe el grado de equidad del mecanismo de generación de señal de realimentación, multiplicar la estimación de tasa equitativa individualizada de cada conexión por un coeficiente reductor no es siempre equivalente a reducir de antemano en la misma proporción el ancho de banda disponible. La equivalencia sólo tiene lugar cuando todas las conexiones tienen su cuello de botella en el mismo puerto. En cualquier otro caso, el efecto del control de congestión que acabamos de describir sería disponer de un ancho de banda de drenaje de las colas igual al producto del coeficiente *Target Utilization* por la suma del ancho de banda disponible, pero tras ser reducido en la cantidad que las conexiones no estranguladas en el nodo han consumido. Inmediatamente, observamos que esta reducción es menor o igual que el producto de *Target Utilization* por el ancho disponible total en el enlace. Aun en este caso, el algoritmo de control de congestión descrito esta justificado. Si el objetivo pretendido es mantener un ancho de banda de drenaje, nótese que aquellas conexiones que podemos hacer responsables de un aumento del nivel de llenado de las colas en el puerto son únicamente las conexiones estranguladas en el mismo puerto; en consecuencia, sólo tiene sentido tener en cuenta el ancho de banda que están ocupando estas conexiones estranguladas, a la hora de reservar un ancho de banda de drenaje en el puerto. En un caso extremo, si sólo

una conexión está estrangulada en un puerto de 150 Mbit/s y el ancho de banda ocupado por las conexiones no estranguladas es 130 Mbit/s, es más razonable reservar un 10% de los 20 Mbit/s restantes y realimentar a la conexión estrangulada un valor de 18 Mbit/s, que reservar un 10% de los 150 Mbit/s totales y realimentar 5 Mbit/s, cuando sólo esta conexión es la responsable del llenado de la cola.

5.3.3 Antecedentes de los algoritmos propuestos

El mecanismo de generación de señal de realimentación propuesto en la sección 5.3.1 recoge una propuesta hecha por Brian Lyles durante el proceso de definición de los mecanismos de soporte para el servicio ABR que tuvo lugar preferentemente en ATM Forum (Lyles y Lin, 1994b), aunque también tuvo su reflejo en la labor de ANSI (Lyles y Lin, 1994a) y del UIT-T SG13 (Lyles, 1994b)(1994c). La contribución de Lyles se basa en el mecanismo de control de flujo *Packet-Pair* (PP), diseñado por Keshav (1991c) y en las aportaciones posteriormente sugeridas por Bernstein (1993).

Como se mencionó en la página 69, el esquema PP es un mecanismo de realimentación implícita. Asume que todos los nodos de la red —más propiamente, todos los nodos susceptibles de ser cuellos de botella— planifican la transmisión de los paquetes según un algoritmo WFQ sin ponderación. Bajo tal suposición, cuando dos paquetes pertenecientes a una misma conexión llegan a un conmutador a la tasa de transmisión de la línea, el intervalo temporal de separación entre ellos a la salida del mismo será inversamente proporcional a la fracción de ancho de banda asignada por el conmutador a la conexión. Nótese que esa separación será máxima cuando atraviesen el nodo de cuello de botella de la conexión. Por tanto, si el destino mide el intervalo de separación entre los instantes de llegada de los dos paquetes, podrá determinar la fracción de ancho de banda asignada por el nodo de cuello de botella a la conexión. Por supuesto, la fuente podrá determinar esto mismo si el destino devuelve reconocimiento por cada uno de los paquetes que recibe, midiendo el intervalo de separación entre reconocimientos. De este modo, una fuente que envíe todos sus paquetes emparejados, podrá obtener una estimación actualizada de la fracción de ancho de banda asignada por el cuello de botella de la conexión cada vez que recibe la pareja correspondiente de reconocimientos. Si tal fracción cambia, la fuente lo detectará automáticamente y se adaptará al cambio detectado. Por supuesto, el espaciado temporal entre parejas de paquetes ha de ser tal que la tasa media de emisión se adapte a la tasa observada.

El mecanismo de ajuste de la tasa de emisión en la fuente que diseñó Keshav (1991b) fue en esencia el siguiente. Supongamos, por simplicidad, que todos los paquetes tienen el mismo tamaño. Sea $\mu(k)$ la tasa de servicio detectada en el cuello de botella detectada por la pareja k-ésima de reconocimientos. El esquema PP predice $\hat{\mu}(k+1)$ mediante un promediado exponencial de la serie temporal $\mu(1), \mu(2), \ldots, \mu(k)$, de modo que

$$\hat{\mu}(k+1) = \alpha \cdot \hat{\mu}(k) + (1-\alpha) \cdot \mu(k), 0 < \alpha < 1$$

el coeficiente de promediado α se modifica dinámicamente para eliminar los picos espurios en $\mu(k)$ y al mismo tiempo adaptarse a los cambios duraderos.

Dado el retardo de ida y vuelta, R, y el número de paquetes pendientes de reconocimiento, S, el esquema PP estima el número de paquetes en la cola del nodo de cuello de botella, X, como la diferencia entre S y el número de paquetes en tránsito $R \cdot \hat{\mu}(k+1)$, esto es,

$$X = S - R \cdot \hat{\mu}(k+1)$$

El esquema PP toma como punto de trabajo aquél en el que la cola de la conexión en el nodo de cuello de botella tiene un nivel de ocupación B. Para que ello ocurra, la fuente ajusta su tasa de emisión $\lambda(k+1)$ en función de la ocupación estimada actual, X, y la tasa de servicio predicha, $\hat{\mu}(k+1)$,

$$\lambda(k+1) = \hat{\mu}(k+1) + (B - X)/R$$

Keshav (1991a) demostró analíticamente y mostró mediante simulación que el esquema PP constituía un sistema de control estable y sin oscilaciones.

Bernstein (1993) analizó el rendimiento del esquema PP de Keshav y aportó diversas mejoras: algunas de ellas se centraron en el mecanismo de ajuste en la fuente, mientras que otras proponían la utilización de otros algoritmos de planificación, tales como *Virtual Clock* o *Delay-EDD*, no diseñados en principio para soportar servicios *best-effort*. Abordaremos, por la importancia que tiene en el diseño de nuestra propuesta, las mejoras sugeridas al respecto de la secuencia de generación de las parejas de paquetes por parte de la fuente.

La fuente PP básica, esto es, según Keshav (1991c), emitía los paquetes emparejados, de modo que la tasa instantánea de emisión era sucesivamente la máxima tasa de transmisión permitida por el enlace de acceso a la red, y la mitad de la tasa de emisión detectada —aproximadamente—. Este hecho era el causante de que, en escenarios en los que existía más de una fuente PP, éstas no eran capaces de detectar cuál era la tasa nominal de emisión del resto. Una solución al problema planteado fue que las fuentes PP enviasen un número determinado de paquetes a la tasa nominal entre las parejas sensoras de paquetes.

Observó también que, tal y como se generaba la pareja sensoras en el esquema PP básico, el primer paquete de la pareja retrasaba su instante de emisión con el fin de emparejarse con el segundo paquete de la pareja. De este modo se daba la situación de que, antes de enviar la pareja sensora, una fuente PP reducía transitoriamente su tasa de emisión aproximadamente a la mitad de su tasa nominal. Ello tenía como efecto que la pareja sensora detectase un estado de congestión mucho menor y, por tanto, que realizase una estimación menos exacta. Para eliminar este segundo problema, sugirió que el segundo paquete de la pareja sensora adelantase su instante de emisión, en lugar de que el primero retrasase el suyo.

Finalmente, observó que, aun con las mejoras anteriores, el emparejamiento de paquetes contribuía a que el patrón de emisión de paquetes fuese esporádico, lo cual no es generalmente deseable. Propuso, entonces, que se restringiera a un valor mínimo el intervalo de separación de los paquetes emparejados, para lo cual estableció que, en lugar de enviar la pareja sensora a la tasa de transmisión máxima, se enviase al producto de la tasa nominal de emisión por el coeficiente de ráfaga (*burstiness*) permitido. Esta modificación tiene como inconveniente que la fuente no puede detectar, en un intervalo de realimentación, un ancho de banda asignado mayor que la tasa de emisión de la pareja sensora. Sin

embargo, dado que el control de flujo comporta un ajuste iterativo, el ajuste permitido por el esquema PP modificado pasa de ser instantáneo a tener crecimiento exponencial, lo cual es más que aceptable.

Lyles y Lin (1994b) esbozaron un mecanismo de soporte para ABR, en donde las elecciones de diseño se basaban en el modelo de servicio que más tarde se plasmaron en Lefelhocz y otros (1996). En lo relativo al cómputo de la señal de realimentación, su contribución fundamentalmente consistió en convertir la señal de realimentación implícita que utilizaba el esquema PP de Keshav/Bernstein en explícita. Más concretamente, en lugar de que la fuente estimase el nivel de llenado de la cola en el nodo, propuso que el conmutador computase, como parte del mecanismo de gestión de *buffers*, una asignación de *buffers* para cada conexión y que lo realimentase a la fuente (véase la página 71). Por otra parte, en lugar de que la fuente estimase la tasa de servicio recibida por la conexión, propuso que el conmutador la estimase a partir del tiempo transcurrido desde que cada célula de la conexión llega a la cabeza de su cola, hasta que finaliza su transmisión. En las escuetas palabras de Lyles:

> The fair share for a VC is calculated by measuring the interval between when a cell is queued on the calendar queue and when it completes transmission. A moments reflection will reveal that this interval reflects the number of other traffic streams that are vying for a share of the output link

Nótese que, respecto del esquema PP Keshav/Bernstein, en el algoritmo propuesto por Lyles desaparece el concepto de pareja sensora: cada célula de la conexión está, desde el punto de vista del esquema PP, emparejada con la que le precede en la conexión y, al mismo tiempo, con la que la sucede. Además, el factor de esporadicidad propuesto por Bernstein es aquí igual a 1, lo cual nos puede llevar a concluir que con este mecanismo el conmutador no puede conseguir computar tasas equitativas mayores que la tasa de emisión de la conexión. No obstante, la conclusión anterior no es cierta, dado que, a diferencia del esquema PP, la señal de realimentación no se estima a partir del intervalo temporal de separación entre los instantes de finalización de transmisión de dos células consecutivas. Con el procedimiento de estimación de Lyles se consigue una estimación igual a la que se obtendría a partir del intervalo de separación entre instantes de finalización de transmisión de dos células consecutivas si la fuente las hubiese emitido emparejadas.

5.3.4 Descripción del mecanismo final: mejora de la estabilidad y de la escalabilidad

Durante la etapa de evaluación mediante simulación del mecanismo propuesto, que se presenta en el capítulo 6, se constató que la interacción entre el algoritmo de cálculo de la tasa equitativa y el algoritmo de control de congestión daba lugar a un comportamiento oscilatorio en las tasas ACR permitidas a las fuentes y en los niveles de llenado de las colas en los conmutadores de cuello de botella. Además, tales oscilaciones aumentaban en amplitud a medida que lo hacía el número de conexiones establecidas. A continuación exponemos las razones del comportamiento que se observó y se describe la solución que se diseñó para resolver el problema.

En la descripción del algoritmo de estimación de tasa equitativa, se ha argumentado que, cuando todas las conexiones que atraviesan un nodo están estranguladas, la estimación de la tasa equitativa en el nodo mediante cómputo de tiempo HOL es exactamente la tasa equitativa *max-min*. En el caso en que sólo una de las conexiones que lo atraviesan no esté estrangulada, el promediado de las medidas de tiempo HOL para las conexiones estranguladas resulta en una estimación exacta también de la tasa equitativa *max-min*. En el caso de que más de una conexión no esté estrangulada, el valor de tiempo HOL para las conexiones estranguladas continuará siendo el valor equitativo *max-min*, pero no ocurrirá necesariamente para las conexiones no estranguladas. De hecho, el valor computado, aun después del promediado, dependerá de la ordenación temporal de la llegada de los paquetes de estas conexiones no estranguladas al conmutador.

En lo que respecta al algoritmo de control de congestión, se ha decidido fijar el valor objetivo de utilización del enlace de salida a un valor menor que la unidad. Ello conlleva que, en régimen estacionario, el nivel de llenado de la cola del puerto correspondiente será cero. Además, tras una perturbación que provoque un llenado instantáneo de la cola, si la estimación da con los valores de tasa equitativa *max-min*, el sistema evolucionará al estado estacionario vaciando las colas con una pendiente temporal aproximadamente igual a $1 - \textit{Target Utilization}$. Si las colas de los conmutadores tienden naturalmente a vaciarse, es también cierto que todas las conexiones tienden a dejar de estar estranguladas efectivamente en los nodos.

De estas consideraciones, extraemos la conclusión de que, la operación concurrente de los algoritmos de estimación y de control da como resultado un régimen de funcionamiento en el que la estimación de tasa equitativa para cualquier conexión depende permanentemente de la ordenación temporal de la llegada de las células al nodo.

Veamos dos ejemplos de funcionamiento que ilustran este fenómeno. En el primer ejemplo, sea un sistema WFQ sin ponderación en el que tomamos *Target Utilization*=0.9, lo cual implica que, una vez estabilizados los valores medidos de tiempo HOL, la suma de las tasas de llegada de células al nodo será un 90% de la capacidad del enlace de salida. Por tanto, la ocupación agregada de las colas del puerto disminuirá en una célula por cada diez que se sirvan y finalmente, el sistema quedará inactivo. A partir de este momento, la célula que llegue en primer lugar será transmitida inmediatamente, lo cual supondrá que su tiempo HOL asociado será $1/r$. Dependiendo de la ordenación temporal entre los instantes de llegada de las células de las conexiones, los valores de tiempo HOL computados a partir de este momento serán iguales a $1/r$ o menores que este valor. En todo caso, la obtención de muestras de tiempo HOL iguales a $1/r$ en el conmutador de cuello de botella de una conexión distorsiona enormemente su estimación de tasa equitativa.

Bajo las mismas suposiciones de partida que el ejemplo anterior, antes de que el sistema quede inactivo, puede darse el vaciamiento de las colas de algunas de las N conexiones; sin pérdida de generalidad, sea tal conexión la conexión A y sea N=18. Sus tasas de llegada al conmutador serán de $0.9 \cdot r/N = r/20$ aproximadamente, y el tiempo de separación entre llegadas para cada conexión será $20/r$. Dado que la conexión A es la única para la que se ha vaciado su cola, por definición se trata de la única conexión no estrangulada. Si el sistema fuese PS, independientemente del instante preciso de llegada de una célula

de A, sería transmitida inmediatamente y su tiempo de transmisión sería $N/r = 18/r$. Observemos que la transmisión de una célula de A termina antes de que llegue al sistema la siguiente. Por tanto, el tiempo HOL de cualquier célula de A será también $18/r$. En un sistema WFQ sin ponderación, en cambio, el instante de llegada de la célula de A tiene una influencia determinante en el valor del tiempo HOL. Puesto que las células de A llegan a una tasa menor que la tasa asignada por WFQ, nunca se retrasará su transmisión más de $N/r = 18/r$; ello es consecuencia de cómo se computa la marca temporal en WFQ. Por tanto, el tiempo HOL de cualquier célula de A será también igual a su tiempo de permanencia en el sistema. Sin embargo, sí puede adelantarse su instante de transmisión; en el mejor de los casos, su instante de transmisión puede coincidir con el de su llegada al sistema, con lo cual su tiempo de transmisión sería $1/r$. En general, dependiendo de la ordenación temporal entre los instantes de llegada de las células de las conexiones, el tiempo HOL de una célula de A puede tomar un valor entre $N/r = 18/r$ y $1/r$.

De los ejemplos anteriores podemos extraer las siguientes conclusiones. En primer lugar, los errores de estimación siempre son positivos, esto es, el valor estimado erróneamente es siempre mayor que la tasa equitativa *max-min*. Por tanto, los errores de estimación provocarán un aumento en la tasa de emisión de las conexiones para las que se ha cometido el error. Ahora bien, los errores de estimación sólo tendrán influencia cuando se dan en el cuello de botella de la conexión. En segundo lugar, los errores de estimación provocan que, tras el retardo de realimentación, las colas del cuello de botella tiendan a llenarse y, consecuentemente, desaparezcan los errores de estimación. En tercer lugar, la magnitud del error de estimación es potencialmente mayor cuantas más conexiones vean vaciadas sus colas.

Resumiendo, los errores de estimación causados por la interacción entre el algoritmo de estimación y el algoritmo de control provocan oscilaciones en los valores de tasa realimentados a la fuente y en los niveles de llenado de las colas de los conmutadores. Este efecto se evidenciará en las simulaciones que se presentan en la sección 6.4.2. Si bien la amplitud de estas oscilaciones está acotada, es cierto que crece con el número de conexiones que atraviesan el conmutador. Nos encontramos pues con un serio problema de escalabilidad.

La solución que proponemos al problema que acabamos de diagnosticar es la siguiente:

> Asignaremos a aquellas conexiones que estén estranguladas en el nodo y que vacíen sus colas por efecto del algoritmo de control de congestión, un ancho de banda menor que el ancho de banda señalizado por realimentación.

La realización de la solución enunciada es la siguiente. Con el fin de determinar qué conexiones de las que han visto vaciadas sus colas son conexiones estranguladas y cuáles no, estableceremos un valor umbral de tasa, denominado *MoreRateThreshold*, típicamente del 90% de la tasa equitativa estimada, superado el cual el nodo considera que la conexión está estrangulada. Más concretamente, sea t_i^k el instante de llegada de la célula k-ésima de la conexión i al sistema; sea ERS_i la estimación de tasa equitativa más recientemente realimentada a la fuente de la conexión i. En t_i^k, si la cola de la conexión i está vacía,

realizamos la comparación siguiente:

$$t_i^k - t_i^{k-1} \leq \frac{1}{MoreRateThreshold \cdot ERS_i}$$

si es cierta, la conexión i se marca.

Cuando, en virtud del algoritmo de planificación seguido, se decida la transmisión de una célula perteneciente a una conexión marcado, esto es, estrangulada con cola recién vaciada, en el instante de ser transmitida *no* es transmitida, sino que:

1. el intervalo de tiempo que esta célula ocuparía en el enlace no se aprovecha;

2. se replanifica la transmisión de la célula:

   - en un algoritmo de prioridad ordenada, tal como WFQ y SCFQ, se suma a la marca temporal de la célula un valor igual al tamaño de la célula ponderado por su peso;
   - en un algoritmo enmarcado, como WRR, a la conexión se le computa el intervalo de transmisión desperdiciado;

3. el tiempo HOL de la célula replanificada no se tiene en cuenta en la estimación de la tasa equitativa para la conexión.

Por último, cuando una célula es escogida para la retención, no podrá ser retenida en ciclos de servicio posteriores.

Obsérvese que, retardando en un ciclo de servicio la transmisión de una célula de la conexión estrangulada, conseguimos que ya haya otra célula en la ocasión siguiente en que el algoritmo de planificación decida transmitir una célula de la conexión. Por tanto, se ha conseguido eliminar una ocasión flagrante para que la estimación de tiempo HOL sea errónea.

Podría argumentarse que el desaprovechamiento del intervalo de transmisión no es necesario para conseguir la replanificación de la célula de la conexión estrangulada. Se ha constatado que, si el intervalo de transmisión es utilizado por una célula de otra conexión, no se consigue detener el vaciamiento de las colas de las conexiones estranguladas sino que finalmente todas quedan vacías y ya no surte efecto la replanificación.

5.4 Análisis del mecanismo de generación propuesto

5.4.1 Prestaciones

Pasamos a discutir razonadamente en qué grado el mecanismo de control de flujo propuesto en este capítulo satisface los aspectos deseables que se presentaron en el capítulo 4.

5.4.1.1 Escalabilidad

A continuación analizamos la influencia de la capacidad de los enlaces y del tamaño de la red sobre el vector de tasas asignadas en régimen estacionario y sobre el espacio de almacenamiento requerido en los nodos para evitar pérdidas.

Hemos visto que, según 5.3.1, el algoritmo de cálculo de tasa equitativa computa la asignación —real o, en algunos casos, máxima— efectuada por el algoritmo de planificación. Según 5.1, la señal de realimentación se envía por tasa explícita, es decir, la señal generada en el nodo de cuello de botella de la conexión se traslada directamente a la fuente. Finalmente, según el comportamiento de fuente descrito en la sección 4.3.2, ésta se adapta en una sola iteración —o en varias, según el valor RIF— al valor realimentado de tasa. Por tanto, una vez alcanzado el régimen estacionario, la fuente de cada conexión estará emitiendo al valor de ancho de banda asignado efectivamente en el nodo de cuello de botella de la conexión. Es inmediato, pues, probar que si multiplicamos por una constante c la capacidad de todos los enlaces de la red, el valor de ancho de banda asignado por el nodo de cuello de botella también resultará multiplicado por el valor c y así también el valor de tasa de emisión de todas las fuentes. El mecanismo de control de flujo propuesto es, por tanto, escalable en términos de capacidad de enlace.

A partir del razonamiento del párrafo anterior, podemos concluir asimismo que el mecanismo de control de flujo propuesto consigue un vector de tasas de emisión en régimen estacionario que es independiente del retardo de ida y vuelta del trayecto de la conexión.

En cuanto a las necesidades de almacenamiento en los nodos, estas dependen de dos factores: de la variación previsible y permisible en el ancho de banda disponible en los enlaces y de la rapidez del ajuste en los sistemas finales.

En cuanto al primero de los factores, un nodo debe garantizar, ante una reducción del ancho de banda asignado a una conexión igual a δBW, una cantidad de *buffers*, en el peor de los casos, igual a $\delta BW \cdot RTT$, donde RTT es el retardo de ida y vuelta desde la fuente hasta el conmutador. Cuando las variaciones en el ancho de banda no se limitan a un valor máximo —sino que están acotadas únicamente por la capacidad del enlace— , el espacio de almacenamiento necesario es igual al producto capacidad por retardo de ida y vuelta. Este valor es inadmisible en redes de alta velocidad. Las alternativas de solución son, bien limitar la reducción posible en cada RTT, bien confiar en que, cuanto mayor sea la capacidad de los enlaces, mayor será el número de conexiones en la red, por lo que las reducciones previsibles por establecimiento de nuevas conexiones serán menos significativas.

El caso indicado en el párrafo anterior supone que la fuente se ajusta al nuevo valor de ancho de banda disponible tras un tiempo igual al retardo de propagación desde el nodo a la fuente. En realidad, este tiempo es mayor, pues depende, por un lado, de la frecuencia de paso de células BRM y de la constante de tiempo del filtrado de estimación de tasa equitativa en el nodo (véase la sección 5.3.1.3). Efectivamente, la señal de realimentación debe esperar en el nodo al paso de una célula BRM. En el peor de los casos, esta espera supone un retraso de $N_{rm} \cdot 1/ACR$, en donde los valores N_{rm} y ACR son los propios de la conexión.

5.4.1.2 Equidad

La asignación de ancho de banda en los nodos la efectúa un algoritmo de planificación que aproxima a GPS, que es equitativo *max-min*, por ejemplo, WFQ, SCFQ o WRR. Dado que el algoritmo de cálculo de tasa equitativa estima la asignación que efectúa el nodo para cada conexión, que la estimación del nodo de cuello de botella es notificada por tasa explícita a la fuente, y que la fuente se adapta al valor realimentado, concluimos que el mecanismo de control de flujo propuesto es equitativo *max-min*. Esto es, la equidad *max-min* del mecanismo propuesto es una consecuencia de la equidad *max-min* del algoritmo de planificación y de que la realimentación generada por el algoritmo de cálculo de tasa equitativa es individual.

5.4.1.3 Resistencia

Es deseable que la asignación de tasas y de *buffers* por parte de un mecanismo de control de flujo quede garantizada independientemente del mecanismo de ajuste de la fuente de cada conexión y, en definitiva, del patrón de tráfico observado para cada conexión.

El mecanismo de control de flujo propuesto consigue proteger el ancho de banda asignado a cada conexión —que es el equitativo *max-min*— en cuanto que, aunque otra conexión no obedezca la realimentación que se le envía y decida emitir a una tasa mayor que el valor realimentado, el algoritmo de planificación equitativa garantiza tal asignación.

Asimismo, el espacio de almacenamiento se asigna también de forma equitativa *max-min*, si bien la asignación de *buffers* no se notifica a las fuentes de las conexiones. Es decir, el nodo dispone del espacio de almacenamiento únicamente para hacer frente a situaciones transitorias de congestión, en las cuales niguna conexión resulta discriminada ni favorecida frente al resto.

5.4.1.4 Estabilidad

En la sección 6.3.5 presentaremos un conjunto de escenarios de simulación que verificarán que el mecanismo de generación de señal de realimentación descrito en la sección 5.3.1, una vez incorporadas las modificaciones de 5.3.4, es estable. En tales escenarios se simularán los siguientes tests de estabilidad del mecanismo de control de flujo:

- número variable de conexiones establecidas;
- conexiones que se establecen en instantes distintos de tiempo;
- conexiones con distintos retardos de ida y vuelta;
- conexiones con cuello de botella en distintos conmutadores;
- desviaciones respecto al régimen estacionario en forma de aumento y reducción de ancho de banda en enlaces troncales.

5.4.2 Complejidad de implementación del mecanismo

La complejidad de cualquier algoritmo destinado a ser realizado mediante *hardware* en tecnología VLSI, como se vió en la página 40, depende no del número de operaciones en sí necesarias, sino principalmente de la cantidad de memoria necesaria para almacenar el estado de funcionamiento del algoritmo, y más precisamente del tiempo necesario para acceder a tal estado. Este estado lo forman el conjunto de punteros a las estructuras de datos que se manejen, por ejemplo, los punteros a las listas enlazadas en un algoritmo de planificación, así como las variables propiamente de estado, por ejemplo, las que almacenan los valores medidos para cada conexión por un mecanismo de generación de señal de realimentación.

Podemos, no obstante, caracterizar la complejidad de implementación un algoritmo a través de dos cantidades: la complejidad espacial y la complejidad temporal.

Es deseable que la cantidad de memoria que necesita un algoritmo sea mínima. La situación óptima se da cuando un algoritmo emplea una cantidad de memoria independiente del número de conexiones que se encuentran establecidas en cada momento, o bien activas; en tal caso se habla de complejidad espacial constante, o también $O(1)$. Es el caso del algoritmo EPRCA (véase la sección 4.4.2.1), que utiliza un único valor MACR, o el de CAPC (véase la sección 4.4.2.2), que utiliza un único valor ERS. El algoritmo ERICA (véase la sección 4.4.3.1), por su parte, mantiene un valor *VCShare* por cada conexión; por tanto, su complejidad espacial es $O(N)$, donde N es el número máximo de conexiones.

El algoritmo que hemos propuesto, en su versión básica, precisa las siguientes variables: una variable de tipo `float` que almacene el valor estimado para cada conexión de tasa equitativa durante el último intervalo de promediado, así como una variable de tipo `float` que acumule el valor provisional de la estimación. En su versión final, el algoritmo precisa, además, una variable de tipo `float` que almacene el instante de tiempo en el que se recibió la célula más reciente de cada conexión; de este modo calculamos el valor de la tasa instantánea de llegada. Por tanto, la complejidad del algoritmo propuesto es $O(N)$.

Por otro lado, es deseable que el número de operaciones que deba realizar el algoritmo para generar la señal de realimentación sea también constante e independiente del número de conexiones que atraviesan el conmutador. Por ejemplo, el cálculo de MACR en el algoritmo EPRCA no depende del número de conexiones; por tanto, su complejidad temporal es $O(1)$. Aparte de las operaciones necesarias cada periodo de realimentación —que en la mayoría de los casos coincide con la llegada de una célula RM—, el número de operaciones que precisa llevar a cabo el algoritmo cada vez que llega una célula de datos debe ser mínimo. En ERICA, se requiere la activación del bit de actividad cada vez que llega la primera célula de una conexión en el intervalo de promediado; además, para el cómputo del factor de carga, debe contar el número total de células que llegan durante el intervalo de promediado.

En el algoritmo que hemos propuesto, se requieren las siguientes operaciones con la llegada de una célula BRM: en primer lugar, se calcula el cociente entre el contador de células transmitidas de la conexión y la suma acumulada de medidas de tiempo HOL de la misma. A continuación, se almacena el valor calculado en la variable que lo va a mantener durante el periodo siguiente. Por último, se inicializan el contador y el acumulador de la

conexión.

En cuanto a las operaciones cada vez que es transmitida una célula de datos, en el algoritmo básico:

1. se calcula el tiempo HOL de esa célula, lo cual requiere acceder al reloj del sistema y a la variable que almacena el instante en que llegó la célula a la cabeza de su cola;

2. se incrementa en 1 el contador de la conexión;

3. se suma la medida de tiempo HOL al valor acumulado;

4. en su caso, se almacena el valor del reloj del sistema en la variable que almacena el instante de llegada de una célula a la cabeza de la cola.

En el algoritmo final, además, cuando llega una célula, se calcula el intervalo de tiempo transcurrido desde la llegada de la última célula, lo cual requiere acceder al reloj del sistema y a la variable que almacena el instante en que llegó la última célula. En su caso, cuando llega el instante de transmitir una célula, se deberá recalcular la marca temporal para conseguir retener la conexión.

5.5 Ampliación del mecanismo para soportar un modelo generalizado para el servicio ABR

El mecanismo de generación de señal de realimentación propuesto en la sección 5.3 como elemento de soporte en la provisión de servicio ABR condiciona las características del servicio ABR (véase la sección 2.3.3) en dos aspectos. Por un lado, la distribución de ancho de banda disponible entre los usuarios del servicio es equitativo de acuerdo al criterio *max-min*, el cual asume que todos los usuarios tienen igual prioridad en la asignación de los recursos. Por otro lado, no se han incorporado mecanismos que permitan garantizar a los usuarios un valor mínimo de ancho de banda, garantía que recoge el parámetro MCR.

Estas limitaciones están presentes también en la mayoría de los algoritmos propuestos en la bibliografía consultada. Además, como se expuso en la sección 2.3.3.2, cuando se decide garantizar un valor mínimo de ancho de banda MCR, el criterio de equidad en la distribución del ancho de banda disponible por encima de la suma de los valores MCR ha de ser replanteado. En las siguientes secciones, introducimos algunas mejoras en el mecanismo de generación de la señal de realimentación que permitirán superar las limitaciones arriba apuntadas.

El objetivo de las ampliaciones que presentamos para el algoritmo de conmutador propuesto en este capítulo es definir un servicio ABR generalizado, en cuanto que permitirá el soporte de otras aplicaciones, además de las aplicaciones transaccionales de datos. Efectivamente, existen algunas aplicaciones de vídeo y de audio que se soportan actualmente sobre la Internet y que reciben una calidad de servicio aceptable. Tales aplicaciones utilizan un esquema de control de flujo por realimentación a nivel de aplicación. Esta realimentación permite a la aplicación adaptar la tasa de codificacicón del *codec* en función

del estado de la red. Este tipo de aplicaciones multimedia podrían soportarse sobre ABR, puesto que ofrece las siguientes bondades. En primer lugar, ABR garantiza una baja tasa de pérdidas de células. Y en segundo lugar, aunque el servicio no garantiza parámetros de calidad de retardo, en la práctica los algoritmos ABR ofrecen una baja variabilidad del retardo de transferencia de células, sobre todo si se compara con la variabilidad que se experimenta en la Internet.

Para que las aplicaciones multimedia citadas puedan efectivamente soportarse sobre ABR es necesario que ABR sea provisto acorde con los dos requisitos siguientes. En primer lugar, la asignación de ancho de banda a cada aplicación debe poder ser ajustada en función de los requisitos de cada una. Por ejemplo, una aplicación de vídeo necesitaría en principio mayor ancho de banda que una aplicación de voz. Proponemos en la sección 5.5.1 que la prioridad de asignación de ancho de banda venga determinada por un peso relativo asignado a cada conexión y que este peso lo empleen los algoritmos de planificación equitativa para realizar una planificación ponderada.

En segundo lugar, es necesario que ABR pueda garantizar efectivamente un valor mínimo de ancho de banda garantizado, puesto que tal valor permitirá que la aplicación mantenga la interacción indispensable para el funcionamiento de la aplicación. En la sección 5.5.2, proponemos un algoritmo de ajuste dinámico de los pesos relativos de la planificación equitativa que permitirá ofrecer esta garantía.

5.5.1 Asignación ponderada del ancho de banda disponible

El algoritmo de planificación escogido en la sección 5.2 para el soporte del control de congestión en servicio ABR podía ser bien WFQ, SCFQ o WRR. Estos algoritmos aproximan con mayor o menor precisión el comportamiento de la disciplina PS, la cual asigna ancho de banda de forma equitativa según el criterio *max-min*. Por otro lado, la propuesta de estimación de tasa equitativa mediante filtrado del tiempo HOL de cada conexión permite realimentar a las fuentes la fracción equitativa *max-min* de ancho de banda asignada por el nodo cuello de botella de la conexión.

El criterio *max-min* es apropiado en aquellos casos en que los usuarios del servicio son considerados como iguales, esto es, se asume que sus requisitos de calidad de servicio en términos de *throughput* son iguales y que la prioridad en la asignación de ancho de banda para cada uno de ellos es idéntica. Cuando tales suposiciones no son ciertas, el proveedor del servicio puede distinguir a los usuarios mediante la asignación de un peso relativo indicador de sus necesidades de ancho de banda. Este objetivo se satisface si empleamos las versiones ponderadas de los algoritmos de planificación anteriores. Según se estableció en la página 38, estos algoritmos de planificación consiguen una asignación de ancho de banda equitativa ponderada en el sentido *max-min*. Podemos desarrollar un razonamiento paralelo al llevado a cabo en la sección 5.3.1, para concluir que la estimación de tasa equitativa mediante filtrado del tiempo HOL de cada conexión cuando la planificación es una discretización de la disciplina GPS, permite realimentar a las fuentes la fracción equitativa *max-min* ponderada de ancho de banda asignada por el cuello de botella de la conexión. Esta discretización de GPS está representada por WFQ, SCFQ o WRR.

En la sección 6.4.3 constataremos que, efectivamente, la distribución de ancho de banda entre conexiones con peso asociado es equitativa según el criterio *max-min* ponderado .

Téngase presente que, la asignación de un valor ϕ_i a cada conexión establecida en la red, tiene un valor ponderador efectivo diferente en cada uno de los conmutadores que atraviesa, dado que en cada uno de ellos el número de conexiones y los valores ϕ_j asociados a cada conexión son diferentes; así lo serán también los valores $\phi_i / \sum_j \phi_j$ en cada conmutador.

5.5.2 Asignación garantizada de ancho de banda mínimo

El soporte de un ancho de banda mínimo garantizado en un servicio ABR permite extender el abanico de potenciales aplicaciones usuarias, según se expuso en la sección 2.3.3.2.

El mecanismo de asignación ponderada de ancho de banda de los algoritmos WFQ, SCFQ y WRR no garantiza, por sí mismo, un valor arbitrario garantizado de ancho de banda, excepto en el caso trivial de que los pesos asignados a cada conexión sean iguales a sus valores MCR respectivos. A continuación exponemos una propuesta de soporte de anchos de banda mínimos garantizados por conexión, que se basa en los algoritmos de planificación equitativa con ponderación.

Asumimos que existe un mecanismo de control de admisión que evita que el ancho disponible en la red pueda ser menor que la suma de los valores MCR negociados en el establecimiento de cada conexión ABR.

Sean N conexiones con valores $\Phi_1, \Phi_2, \ldots, \Phi_N$ asignados en el establecimiento de la conexión y sus correspondientes valores $MCR_1, MCR_2, \ldots, MCR_N$. Cada puerto comprobará periódicamente si la tasa equitativa estimada para cada conexión, ERS_j está por encima de su valor MCR_j. Supongamos que se detecta que no ocurre así para la conexión i, esto es, que el valor $ERS_i/MCR_i < 1$. Nótese que ello puede ocurrir aunque el ancho de banda disponible sea mayor que $\sum_j MCR_j$; por ejemplo, dados $MCR_1 = 1$ y $MCR_2 = 2$ y $\Phi_1 = \Phi_2 = 1$, cuando el ancho de banda es 3.5 —que es mayor que $MCR_1 + MCR_2 = 3$— el ancho de banda asignado para cada una es 1.75, que está por debajo de MCR_2. Proponemos que, en tal caso, Φ_i aumente de valor, con lo cual el ancho de banda asignado por el algoritmo de planificación aumentará por encima de MCR_i.

Para ello periódicamente el conmutador deberá modificar los pesos efectivamente empleados en la planificación $\phi_1, \phi_2, \ldots, \phi_N$, que se inicializan a los valores $\Phi_1, \Phi_2, \ldots, \Phi_N$, según la siguiente expresión:

$$\phi_i \leftarrow max(\Phi_i, \phi_i \cdot \frac{MCR_i}{ERS_i}) \quad \forall i$$

Obsérvese que las conexiones para las cuales se esté cumpliendo $ERS_j > MCR_j$, no modificarán su valor ϕ_j, esto es, éste se mantendrá igual a Φ_j. En cambio, las conexiones para las que se cumpla $ERS_j < MCR_j$ aumentarán su valor ϕ_j. El efecto es que la fracción equitativa ponderada que asignará el algoritmo de planificación a la conexión i será mayor que el que se venía asignando.

Veamos el siguiente caso particular. Partimos de una situación en la que todas las conexiones están por encima de su valor MCR_j; por tanto, $\forall j \in [1, N], \phi_j = \Phi_j$. En un momento dado, una de ellas, la conexión i, pasa a tener $ERS_i < MCR_i$. Ello provoca que $\phi_i = \Phi_i \cdot \frac{MCR_i}{ERS_i}$, valor que es mayor que Φ_i, mientras que para el resto de conexiones $\forall j \neq i, \phi_j = \Phi_j$. Este ajuste hace que la fracción de ancho de banda que recibirá la conexión i a partir de este momento sea

$$\frac{\phi_i}{\sum_j \phi_j} = \frac{\Phi_i}{\sum_{j \neq i} \Phi_j + \Phi_i \cdot \frac{MCR_i}{ERS_i}} \cdot \frac{MCR_i}{ERS_i}$$

que es mayor que la fracción de ancho de banda que i recibía antes del ajuste

$$\frac{\Phi_i}{\sum_{j \neq i} \Phi_j + \Phi_i}$$

El periodo de actualización de los pesos $\phi_1, \phi_2, \ldots, \phi_N$ es un parámetro crítico en la estabilidad del sistema. Existen varias opciones.

Una de ellas consistiría en que cada conmutador modificase localmente los pesos que asigna en la planificación de las células. Podría fijarse en este caso que la modificación de los pesos se realizase cada vez que se actualiza el valor ERS de la conexión. Esta elección tiene un inconveniente. La modificación del peso asociado a una conexión surte efecto en la asignación de marca temporal de la próxima célula de la conexión. Si esta conexión tiene muchas células ya en la cola, estas continuarán siendo servidas según la ordenación que resultó antes de la modificación del peso. En tal situación, el valor ERS calculado continuará reflejando la ponderación que existía antes de la última modificación, lo cual provocará que, en la próxima modificación del peso de la conexión, éste resulte de nuevo aumentado. Este comportamiento puede producir inestabilidad. Proponemos que los pesos de una conexión no se vuelvan a modificar hasta que la célula para la que va tener efecto esta modificación no haya sido transmitida.

Otra opción es que la modificación de los pesos sea gestionada de forma centralizada en la red, a partir de los valores MCR de las conexiones ABR establecidas en la red y del ancho de banda disponible para ABR en la red. Esta opción plantea, sin embargo, los inconvenientes de cualquier solución no distribuida.

En relación con los criterios de equidad en presencia de MCR que se discutieron en la sección 2.3.3.2, nótese que el procedimiento que acabamos de proponer para garantizar el ancho de banda mínimo resulta en la aplicación del criterio denominado *de mínimos*.

5.6 Conclusiones

La utilización de un algoritmo de planificación equitativa en la provisión de la clase de servicio ABR permite garantizar efectivamente la asignación de ancho de banda que se decide en la red. Además, el criterio de distribución de ancho de banda puede ser más

general que el criterio *max-min*: el criterio *max-min* ponderado. Finalmente, la red puede garantizar un ancho de banda mínimo si modifica los pesos relativos en la planificación en función del ancho de banda disponible en la red.

Las tres ventajas anteriores se han incorporado en la clase de servicio ABR mediante un nuevo mecanismo de generación de la señal de realimentación. Tal mecanismo estima la asignación de ancho de banda que el algoritmo de planificación equitativa efectúa en el nodo. De este modo, se realimenta a la fuente un valor de tasa equitativo *max-min* ponderado según el criterio de mínimos.

Capítulo 6

Evaluación de la propuesta mediante simulación

En este capítulo, se evalúan las prestaciones del algoritmo de conmutador propuesto en el capítulo 5. Para ello se ha escogido el modelado de simulación por eventos discretos. En la sección 6.1 se justifica la elección de la simulación por eventos discretos. El modelo de simulación comprende aquellos elementos que operan a nivel de capa ATM y dan soporte a la clase de servicio ABR. En la sección 6.2 se describe las características del modelo de simulación, así como de los escenarios escogidos para la evaluación. Los parámetros medidos en la evaluación han sido la tasa máxima de emisión permitida, el nivel de llenado de las colas en los conmutadores y la utilización de los enlaces. A partir de estos parámetros, en la sección 6.3 se presentan los resultados de los tests, que evalúan la eficiencia, la escalabilidad, la equidad, la estabilidad y la resistencia del esquema de control de flujo basado en el algoritmo de conmutador propuesto. Finalmente, algunas alternativas de diseño que se plantearon en el capítulo 5 son simuladas en la sección 6.4.

6.1 Metodología de modelado

En la Tesis se ha escogido el modelado de simulación, como metodología para la evaluación de la contribución descrita en el capítulo 5. Se ha descartado la utilización de modelos analíticos por las razones que apuntamos a continuación.

Primero, los estudios que utilizan modelos analíticos para evaluar mecanismos de control de flujo en ATM son excesivamente sencillos. Yin y Hluchyj (1994) y Bolot y Shankar (1992) emplean la aproximación *fluid flow* (Kleinrock, 1976) para modelar un esquema de control de flujo de ajuste de tasa por realimentación binaria. Ritter (1996) extiende esta aproximación para modelar el esquema de control de flujo normalizado para ABR, esto es, operando en marcado tanto binario como explícito, pero el análisis sólo se puede aplicar al estado estacionario y los parámetros de mérito que se obtienen con la aproximación anterior se limitan al tamaño máximo de cola y al retardo máximo de transferencia. Ritter (1997) consigue un modelo más realista, pues utiliza análisis estocástico en tiempo dis-

creto, y extrae conclusiones respecto del comportamiento dinámico, esto es, determina la estabilidad del control.

No obstante, los estudios anteriores asumen que los algoritmos de planificación en los nodos es de tipo FCFS. Existen otros estudios en los que el algoritmo de planificación es de tipo equitativo. En uno de ellos, Mascolo y otros (1996) proponen un algoritmo de ajuste basado en la realimentación del nivel de ocupación de la cola de cada conexión, pero asume que el retardo de ida y vuelta es poco variable y conocido a priori, lo cual no es admisible en un escenario real. En otro estudio, Altman y otros (1993) estudian la estabilidad del control de flujo del protocolo *packet-pair*, empleando una aproximación estocástica en tiempo discreto.

Segundo, los estudios que hacen uso de modelos analíticos para evaluar las prestaciones de los algoritmos de planificación equitativos se centran en el soporte de aplicaciones de servicio garantizado. Así, Parekh (1994) obtiene valores de retardo máximo de tránsito por una red de nodos WFQ, bajo la suposición de que en la entrada se emplea conformación de tráfico tipo *leaky bucket*. Golestani (1995) hace lo propio con el algoritmo SCFQ. Sin embargo, cuando se estudia la interacción de los mecanismos de control de flujo con los algoritmos de planificación equitativa, Demers y otros (1989) y Bernstein (1993) emplean modelado de simulación por eventos discretos.

Y tercero, los modelos analíticos estudian aisladamente el comportamiento de cada conexión, modelando el efecto del resto de conexiones mediante variables estocásticas fácilmente tratables. Esta simplificación es inadmisible en el modelado del control de flujo en ABR, en donde la interacción entre los bucles de control de cada conexión es determinante en las prestaciones que obtiene cada una de ellas.

El modelado de simulación en cambio permite abordar con éxito la complejidad intrínseca del mecanismo de control de flujo propuesto como contribución de esta Tesis.

6.1.1 Herramienta de simulación: BONeS DESIGNER

BONeS DESIGNER es un paquete *software* de Alta Group of Cadence Systems, Inc. (Foster City, California) destinado al modelado y simulación de redes de comunicaciones mediante la técnica de eventos discretos.

El modelado mediante DESIGNER se lleva a cabo a través de un interfaz gráfico, mediante la creación de *bloques*, cuyo interfaz exterior se define en la forma de *puertos* de entrada y/o puertos de salida, y cuyo comportamiento se define como el procesado que el bloque lleva a cabo sobre las estructuras de datos que entran por los puertos de entrada y que eventualmente pueden salir por los puertos de salida. Algunos bloques definen su comportamiento haciendo uso de otros bloques ya definidos, lo cual permite un modelado jerárquico, mientras que otros lo hacen mediante código C/C++; estos últimos se denominan *primitivas*. Además de los bloques, se definen estructuras de datos, los cuales modelan las unidades de datos intercambiadas en las redes de comunicación. Por último, la comunicación entre los bloques se modela a través de la unión de uno o varios puertos de salida de un bloque con uno o varios puertos de entrada de otro.

La ejecución de un bloque durante la simulación se activa con la habilitación de cual-

quiera de sus puertos de entrada. Tal habilitación tiene lugar cuando una estructura de datos llega al puerto. Algunos bloques necesitan que todos sus puertos de entrada estén habilitados antes de activar su ejecución, mientras que otros sólo necesitan la habilitación de algunos de ellos. En cualquier caso, la activación de la ejecución del bloque conlleva la deshabilitación de sus puertos de entrada. Si la ejecución del bloque supone la generación de una estructura de datos por algún puerto de salida, una copia de la estructura de datos se envía a cada uno de los puertos de entrada de los bloques conectados al puerto de salida y además, estos quedan habilitados. De este modo, la ejecución durante la simulación se va propagando por el modelo hasta que se alcanza un bloque cuya ejecución no pueda ser activada porque no se cumplan sus condiciones de activación. Este tipo de ejecución tiene lugar en tiempo cero, esto es, no provoca el avance del reloj de tiempo de simulación.

No ocurre lo mismo con los sucesos asíncronos. Los sucesos asíncronos desatan la ejecución de los bloques en un instante dado de tiempo de simulación y son generados básicamente por dos tipos de bloques: los bloques de retardo, que, tras un intervalo de tiempo, reanudan la ejecución de simulación en el mismo punto del modelo del sistema, y los bloques temporizadores, que, a diferencia de los anteriores, permiten reanudarla en cualquier punto (incluso varios) del modelo del sistema. Son estos sucesos los que hacen progresar al reloj de tiempo de simulación.

Finalmente, toda simulación requiere de una semilla para la eventual generación de números aleatorios. En DESIGNER, esta semilla base de la simulación, que puede especificar el usuario, se combina durante la inicialización de la simulación con los identificadores de los bloques en los que se requiera efectivamente la generación de números aleatorios —tales como generador de *jitter* en los enlaces— para obtener una semilla individualizada.

6.1.2 Aproximación monocapa al modelado

El modelado utilizado para la evaluación de prestaciones del mecanismo de generación de señal de realimentación propuesto en el capítulo 5 puede calificarse como aproximación monocapa al modelado de sistemas de comunicación.

Esta aproximación se basa en escoger una capa de protocolo en donde concentrar el esfuerzo de modelado y, a continuación, simplificar el modelo de simulación empleado para las capas inferiores y para las capas superiores. La elección de qué capa modelar en detalle dependerá obviamente de cuáles sean los parámetros de mérito que se desean obtener.

En el caso que nos ocupa, la capa escogida para un modelado detallado es la capa ATM, pues a este nivel opera el control de flujo para la provisión de servicio ABR y, en particular, el mecanismo de generación de señal de realimentación propuesto en el capítulo 5. Además, como veremos en este capítulo, los parámetros de mérito que presentaremos son fundamentalmente la eficiencia de uso de los recursos a nivel de capa ATM y la equidad en el reparto de tales recursos. Por otro lado, el modelo de las capas superiores se ha simplificado enormemente, de hecho no se han distinguido posibles mecanismos presentes en esas capas, ni siquiera se ha parametrizado su comportamiento. En particular, el evento básico de la simulación es la llegada de una célula ATM.

Como exponen Guijarro y otros (1998), esta aproximación a la evaluación de prestaciones de sistemas de comunicaciones complejos está cada vez menos justificada, por la presencia de mecanismos de control de congestión tanto a nivel de capa ATM como a otros niveles superiores, tales como la capa de transporte. No obstante, la aproximación monocapa escogida en este capítulo se justifica por las razones mencionadas anteriormente. Además, la investigación de mecanismos de soporte ABR ha utilizado exclusivamente esta misma aproximación.

6.2 Descripción del modelo de simulación

6.2.1 Modelado de las capas inferiores

Las capas inferiores, que son las que proporcionan servicio a la capa ATM, son la capa física y el medio físico. La unidad de datos que maneja el modelo de las capa inferiores es una célula ATM. Así, cada enlace se modela como un generador de retardo de propagación de valor 5 μs/Km, más un retardo de transmisión igual al tiempo de transmisión de una célula ATM, esto es, igual a 424/LR μs, siendo LR la capacidad del enlace en Mbit/s.

El acceso al medio físico, tanto desde las estaciones como desde los conmutadores, se modela mediante un ranurado síncrono de periodo igual al tiempo de transmisión de una célula ATM. Se trata de un modelo sencillo por cuanto que únicamente modela la siguiente característica común a todos los sistemas de transmisión: garantiza que las células ATM nunca podrán ser transmitidas a una cadencia mayor que la permitida por la capacidad de transmisión del sistema. Se dejan de lado los formatos detallados de las tramas de transporte utilizados en cada sistema de transmisión como, por ejemplo, la trama STM-1 de SDH.

El ranurado descrito está activo incluso en ausencia de células que transmitir. Por un lado, el modelado de este comportamiento es más sencillo. Por otro, ello ocasiona que, cuando una célula ATM llegue a un puerto de salida vacío, deba esperar al inicio de la ranura siguiente. De todos modos, cuando el número de conexiones establecidas no es despreciable el grado de desocupación de los puertos de salida es mínimo, por lo que este comportamiento no conservativo no tiene efecto apreciable. Con el fin de evitar una sincronización patológica de eventos en la simulación, el instante de tiempo en que tiene lugar la primera ranura de cada enlace se toma de una distribución aleatoria uniforme entre 0 y el tamaño de la ranura.

Finalmente, el retardo de transmisión se aleatoriza mediante la adición de una componente positiva aleatoria. Se trata de una variación, o *jitter*, que contribuye a modelar de forma más realista la capa física y a eliminar la sincronización de eventos en la simulación. Tal variación se genera mediante un valor aleatorio extraído de una distribución exponencial cuya esperanza se toma igual a la décima parte del tamaño de la ranura. Con el fin de preservar la secuencia en la transmisión de las células a través del medio, tal distribución está truncada al valor del tamaño de la ranura. La variación es siempre positiva, con lo cual siempre se aumenta el retardo de transmisión de cada célula considerada individualmente. Sin embargo, la diferencia entre los retardos de transmisión de dos células

consecutivas se distribuye según una variable aleatoria que es la diferencia de dos variables aleatorias exponenciales con la misma esperanza; así, el intervalo entre transmisión de células consecutivas tiene una esperanza nula, o en otras palabras, la tasa de transmisión de células no se desvía, en media, de su valor nominal.

6.2.2 Modelado de las capas superiores

Las capas superiores, que son el usuario de la capa ATM, se han modelado sencillamente como un generador infinito de unidades de datos. Se asume que sólo hay una entidad de capa usuaria en cada terminal. Esta simplificación es consecuencia directa de la decisión de modelar la capa usuaria como un generador infinito de unidades de datos. La unidad de datos que maneja el modelo de las capas superiores es igual a la carga útil de una célula ATM.

Con este modelo de capa usuaria, la capa ATM dispone de una célula en cada oportunidad de transmisión, según permite el algoritmo de control de flujo ABR, es decir, cada 1/ACR μs, en donde ACR está en unidades de células por μs.

Este modelo de capa usuaria no es realista. Primero, las aplicaciones de datos, que son las primeras candidatas a utilizar la clase de servicio ABR, son intrínsecamente esporádicas; por ejemplo, un acceso remoto a un servidor WWW. Segundo, el protocolo de transporte más utilizado para soportar aplicaciones de datos es TCP; este protocolo controla el flujo mediante ventana deslizante adaptativa, lo cual se traduce en un envío en bloque de segmentos cada vez que se abre la ventana de congestión. Y tercero, la unidad de datos que se pasa a la capa AAL siempre provoca segmentación, lo cual se traduce en la generación de un bloque de células; por ejemplo, el tamaño por defecto de un segmento TCP encapsulado en IP es 576 bytes, al cual se le añaden 8 bytes tras ser procesado por AAL5, lo cual supone un número de 13 células ATM por segmento TCP.

Sin embargo, en las siguientes situaciones el modelo sí es realista. Que el terminal lo constituya una unidad de interconexión de red LAN de datos a red ATM, en la que el acceso sea del tipo *classical IP* o *LAN emulation*; en tal situación, los *buffers* del *router* o del *bridge* suelen estar desocupados con muy baja probabilidad. Que el terminal multiplexe un número de conexiones de datos originadas en el mismo terminal, con lo que, de forma análoga al caso de la unidad de interconexión, los *buffers* de acceso a la capa ATM estarán casi siempre semiocupados.

6.2.3 Modelado de la capa ATM

A efectos del modelado de la capa ATM, se han asumido las siguientes suposiciones:

1. Sólo se establecen conexiones ATM bajo la clase de servicio ABR.

2. Las conexiones ATM ya se encuentran en fase de transferencia de datos; no se modela por tanto las fases de establecimiento y de liberación y, consecuentemente, no se modela ningún aspecto de señalización ni de encaminamiento.

3. Por simplicidad de modelado, la identificación de cada conexión ATM se realiza mediante un identificador VPI/VCI de significación global, esto es, único en toda la red.

En el modelado de los conmutadores, se ha asumido una disposición de colas en los puertos de salida, necesaria para la aplicación del mecanismo de control de flujo propuesto en la sección 5.3. La matriz de conmutación no provoca bloqueo. Además, las posibilidades de conmutación de células se han reducido, con el objeto de simplificar el modelado. Como se muestra en la figura 6.1, los puertos de entrada/salida del conmutador se han dispuesto en el modelo en dos grupos: izquierda y derecha. Las conexiones que entran por alguno de los puertos del grupo de la izquierda, que numeramos con apóstrofe, sólo pueden ser conmutadas a cualquiera de los puertos del grupo de la derecha, que numeramos sin apóstrofe, y viceversa.

Los algoritmos de planificación equitativa modelados han sido WFQ (véase la sección 3.3.2) y SCFQ (véase la sección 3.3.3). En el modelado de ambos, se ha empleado una estructura de datos consistente en una lista ordenada por conexión. El modelo de WFQ implementa el algoritmo de eliminación iterativa descrito en la página 48. Por su parte, el modelo de SCFQ simplemente incorpora una decisión de implementación, que exponemos a continuación.

Dado que todas las células tienen el mismo tamaño y que todas las conexiones tienen el mismo peso, se da la situación de que las células de cabeza de cola tienen uno de dos valores de marca temporal, según se discutió en la sección 3.3.3. Ello se traduce en que, cuando se escoge la siguiente célula a transmitir, muchas células de cabeza tienen la misma marca temporal, por lo que el orden de búsqueda tiene mucha influencia en el orden efectivo de transmisión. Para eliminar esta fuente de inequidad, se ha decidido iniciar la búsqueda de la siguente célula a transmitir a partir de la cola de la conexión cuya célula acaba de ser transmitida. Un momento de reflexión mostrará que esta es la mejor alternativa.

6.2.3.1 Modelado del soporte ABR

En el modelo, las conexiones ABR sólo transmiten datos en un sentido, desde el terminal origen hacia el terminal destino, por lo que el modelo de simulación del terminal origen sólo incorporará las reglas de comportamiento de fuente (véase la página 76), mientras que el modelo del terminal destino, las reglas de comportamiento de destino (véase la página 83)

La célula RM se modela según *ATM Forum Traffic Management 4.0*; en concreto, incluye los campos DIR, BN, CI, NI, ER, CCR y MCR (véase la sección 4.2.1)

Por su parte, el comportamiento de fuente ABR se modela a partir de la implementación propuesta en el *Informative Appendix I* de *ATM Forum Traffic Management 4.0*; el modelo se muestra en forma de pseudo-código en la figura 6.2. Las variables empleadas se muestran en la tabla 6.1, así como sus valores iniciales. En el modelo de la fuente ABR se han tomado las siguientes decisiones de implementación:

- Cuando el valor ACR llega a ser menor que el valor TCR=10 cell/s, sólo se envían

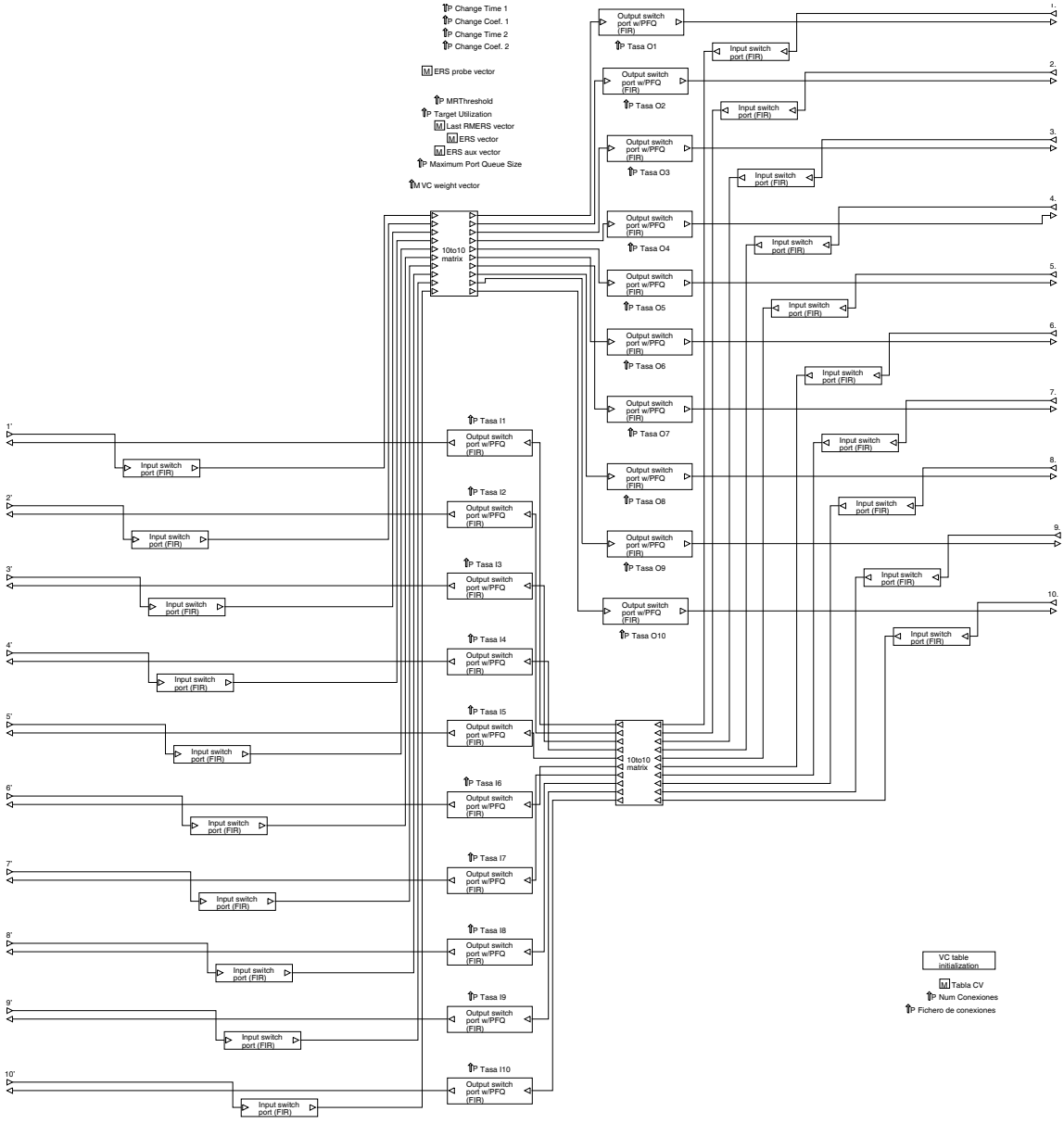

Figura 6.1. MODELO BONES DE CONMUTADOR ATM

Variable	*Significado*	*Valor inicial*
`ACR`	ACR	ICR
`count`	número de células desde la última FRM	Nrm
`unack`	número de FRM desde la última BRM	0
`time-to-send`	instante de emisión de la célula siguiente dentro de tasa	-
`last-RM`	instante de emisión de la última FRM	now-Nrm/ICR
`CI-VC`	estado EFCI de la conexión	0

Tabla 6.1. VARIABLES LOCALES UTILIZADAS EN EL MODELO DE FUENTE ABR

células FRM y, además, se envían a la tasa TCR; esta decisión respeta la regla 11 (véase la página 83).

- Dado que los valores TBE y FRTT no son relevantes cuando las fuentes son persistentes, salvo en la fase inicial de transferencia de datos, no se han considerado en el modelado del comportamiento de la fuente ABR.
- Dado que las conexiones sólo transmiten datos en un sentido, el modelo de fuente ABR nunca recibirá células FRM y, en consecuencia, nunca precisará emitir células BRM, por lo que no contempla la condición nº 2 de la regla 3 (véase la página 78).
- Cuando se modifica el valor ACR, este nuevo valor no afecta a la célula cuya emisión ha sido ya planificada, sino que afecta a la siguiente.

Los valores por defecto de los parámetros ABR estandarizados en *ATM Forum Traffic Management 4.0* y que se han empleado en las simulaciones que presentaremos se muestran en la tabla 6.2. Nótese los siguientes aspectos:

- Como valor por defecto de PCR se ha tomado la capacidad total del enlace de acceso en cada terminal.
- El valor de ICR se ha fijado a una centésima parte del valor PCR, con el fin de limitar el grado de congestión en los nodos durante la fase de funcionamiento en bucle abierto, esto es, hasta que llega la primera célula BRM a la fuente; hecha esta salvedad, la elección del valor ICR no es crítica, a diferencia de lo que ocurriría en el caso de fuentes esporádicas con ráfagas de duración menor que un tiempo de ida y vuelta.
- Dado que las fuentes empleadas en las simulaciones son persistentes, se ha decidido eliminar el mecanismo *use-it-or-lose-it* especificado en la regla 5 (página 79), por innecesario: para ello se ha aumentado ADTF hasta 10 segundos.
- También se ha desactivado el mecanismo de protección especificado en la regla 6 (página 80), que protegía frente al bloqueo de las células RM que pudiese producirse

```
if cell-enters-empty-queue-event
    if time-to-send not scheduled
        schedule: time-to-send = now

if now >= time-to-send and data-in-queue
    if ACR < TCR
        send RM (dir=forward, CCR=ACR, ER=PCR, CI=0, NI=0, CLP=1)
        schedule: time-to-send = now + 1/TCR
    else {
        if (count <= Nrm) or
        ((count > Mrm) and (now <= last-RM + Trm))
            if (time > ADTF) and (ACR > ICR)
                ACR = ICR
            if (unack >= Crm)
                ACR = ACR - ACR   CDF
                ACR = max(ACR, MCR)
            send RM (dir=forward, CCR=ACR, ER=PCR, CI=0, NI=0, CLP=0)
            count = 0
            last-RM = now
            unack = unack + 1
        else
            send data cell (CLP=0, EFCI=0)
        count = count + 1
        schedule: time-to-send = now + 1/ACR
    }

if receive RM (dir=backward, CCR, ER, CI, NI, BN)
    if (CI = 1)
        ACR = ACR - ACR   RDF
    else if (NI = 0)
        ACR = ACR + PCR   RIF
        ACR = min(ACR, PCR)
    ACR= min(ACR, ER)
    ACR= max(ACR, MCR)
    if BN = 0
        unack = 0
```

Figura 6.2. PSEUDO-CÓDIGO DEL MODELO DE FUENTE ABR

Parámetro	*Significado*	*Valor TM 4.0*	*Valor Simulación*
PCR	Peak Cell Rate	negociado	Capacidad del enlace
MCR	Minimum Cell Rate	0	PCR/1000
ICR	Initial Cell Rate	negociado	PCR/100
RIF	Rate Increase Factor	1/16	1/16
Nrm	Nrm	32	32
Mrm	Mrm	2	2
RDF	Rate Decrease Factor	1/16	1/16
CRM	Missing RM-cell count	$\lceil TBE/N_{\mathrm{rm}} \rceil$	1000
ADTF	ACR Decrease Time Factor	500 ms	10 s
Trm	Trm	100 ms	100 ms
FRTT	Fixed-RTT	-	n/a
TBE	Transient Buffer Exposure	negociado	n/a
CDF	Cutoff Decrease Factor	1/16	1/16
TCR	Tagged Cell Rate	10 cell/s	10 cell/s

Tabla 6.2. PARÁMETROS DE FUNCIONAMIENTO ABR: VALORES POR DEFECTO EN *ATM Forum Traffic Management 4.0* Y EN LA SIMULACIÓN

por caída de enlaces y/o congestión severa en los nodos: para ello se ha fijado CRM en 1000 células.

- Se ha descartado la posibilidad de emitir células FRM fuera de tasa: para ello, como valor MCR se toma *PCR*/1000 que, para el valor utilizado de capacidad de enlace de acceso igual a 150 Mbit/s, es igual a 150/424/1000=354 cell/s. Este valor es mucho mayor que 10 cell/s, que es el valor de TCR. Por tanto, el valor ACR nunca llegará a ser igual o menor que TCR.

En cuanto al comportamiento de destino ABR, éste se ha modelado a partir de la implementación propuesta en el *Informative Appendix I* de *ATM Forum Traffic Management 4.0*; el modelo de destino ABR se muestra en forma de pseudo-código en la figura 6.3. Hemos tomado las siguientes decisiones de implementación:

- No se contempla la posibilidad de que el destino modifique los valores ER, CI y/o NI de la célula FRM en función del estado interno de congestión, únicamente en función del estado EFCI de la conexión, aunque ambas posibilidades se citan en la regla 2 (véase la página 84); además, tampoco generará células BRM, posibilidad que contempla la regla 5 (véase la página 85).

- Dado que en el sentido de vuelta de la conexión nuestro modelo sólo transmite células BRM, la tasa de llegada de células FRM será siempre mucho menor que la tasa disponible en el sentido de vuelta para células BRM; en consecuencia, las reglas 3 y 4 (véase la página 84) no se contemplan en el modelado, por no ser efectivas

```
if receive data cell
    CI-VC= EFCI state of cell

if receive RM (dir=forward, CCR, ER, MCR, CI, NI)
    send RM (dir=backward, CCR, ER, MCR, CI-VC, NI, CLP=0)
    CI-VC = 0
```

Figura 6.3. PSEUDO-CÓDIGO DEL MODELO DE DESTINO ABR

- Dado que la elección de los valores de los parámetros evita la emisión de células FRM fuera de tasa, la situación contemplada en la regla 6 (véase la página 85) no puede darse.

El comportamiento de referencia del conmutador se modela según las reglas de comportamiento de conmutador de *ATM Forum Traffic Management 4.0* (véase la página 86), con las siguientes decisiones de implementación:

- Se emplea únicamente el marcado de tasa explítica, dado que es el soportado por el mecanismo de generación de la señal de realimentación propuesto en la sección 5.3.
- El conmutador no generará células BRM, pues no lo contempla el mecanismo propuesto, aunque es un posibilidad contemplada en la regla 2 (véase la página 87).
- El conmutador mantiene la secuencia de células RM y también la secuencia global de células FRM y datos (regla 3, página 87).

En cuanto al modelado del mecanismo de generación de la señal de realimentación, cabe comentar dos aspectos. En primer lugar, cada célula a transmitir a través del puerto de salida lleva asociada un valor de tiempo HOL (véase la sección 5.3.1), almacenado en la variable HOL Time, a excepción de las células retenidas, pues estas han sufrido un retraso de planificación —en este caso se asigna el valor -1 a HOL Time— (véase la sección 5.3.4). En la figura 6.4 se ilustra el proceso de actualización de las variables ERS vector, que almacena la suma de los valores HOL Time de cada conexión, y de ERS aux vector, que almacena el número de sumandos en ERS vector para cada conexión. Cada vez que se transmite una célula por un puerto de salida de los grupos de la derecha se actualizan estas variables. Téngase en cuenta que las células BRM, que salen por los puertos de salida de los grupos de la izquierda no deben tenerse en cuenta.

En segundo lugar, cada célula BRM que llega a un puerto de entrada de un conmutador provoca la ejecución del cálculo de la estimación de tasa equitativa para la conexión correspondiente —sea la conexión i— (véase la sección 5.3.1.3). Tal valor es igual a:

$$\frac{1}{\text{ERS vector}[i]/\text{ERS aux vector}[i]}$$

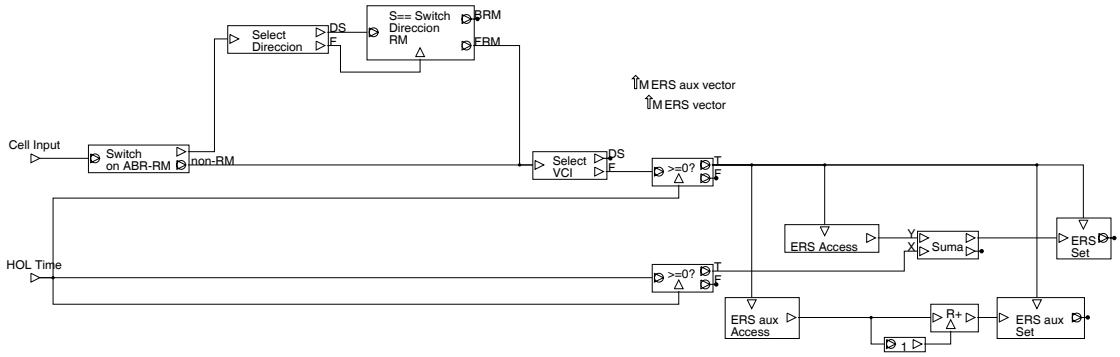

Figura 6.4. DIAGRAMA BONES PARA EL CÓMPUTO DE LA TASA EQUITATIVA EN EL PUERTO DE SALIDA

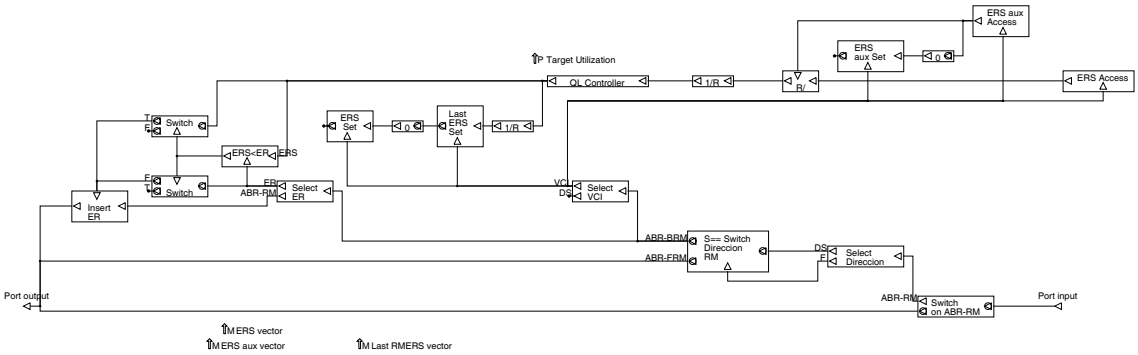

Figura 6.5. DIAGRAMA BONES PARA EL CÓMPUTO DE LA TASA EQUITATIVA EN EL PUERTO DE ENTRADA

y multiplicado por *Target Utilization*, según se propuso en el algoritmo de control de congestión en la sección 5.3.2. En la figura 6.5 se ilustra este procedimiento, así como el de comparación de la estimación con el valor ER de la célula BRM.

6.2.4 Escenarios de test

Las configuraciones de nodos, enlaces y terminales que empleamos para la evaluación de prestaciones son las siguientes:

6.2.4.1 Configuración de dos conmutadores

La configuración de 2 conmutadores se ilustra en la figura 6.6. Las capacidades de los enlaces de acceso son de 150 Mbit/s. Se trata de la configuración más sencilla de las tres

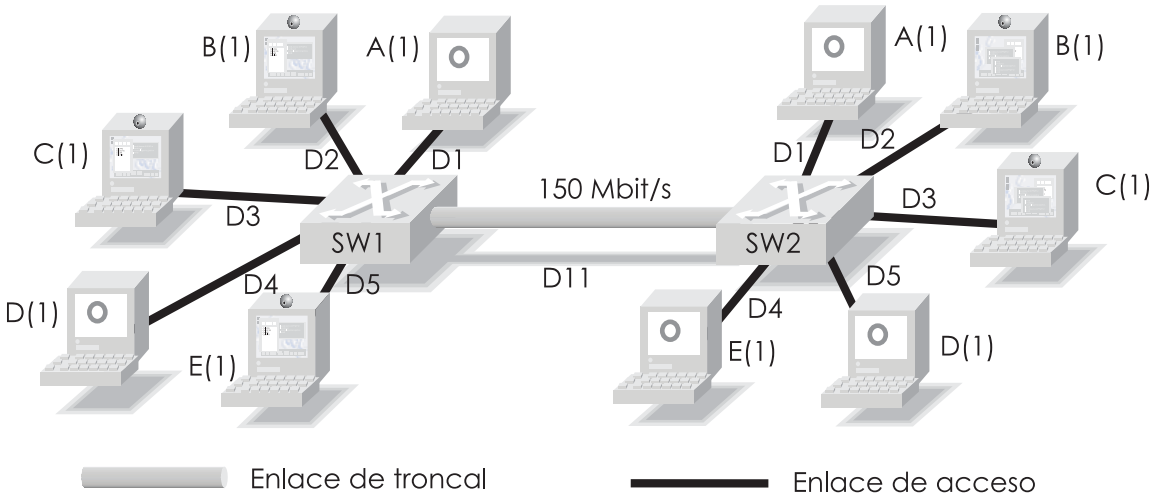

Figura 6.6. CONFIGURACIÓN DE DOS CONMUTADORES

que presentaremos, pues sólo dispone de un enlace troncal.

La notación empleada para la especificación de los extremos de las conexiones, tanto en esta configuración como en las posteriores, es la siguiente:

- las letras mayúsculas del abecedario identifican grupos de conexiones, en origen y en destino;

- los números arábigos entre paréntesis indican el número de conexiones del grupo, esto es, identifican a un subgrupo.

Así, según la notación descrita, en la configuración de la figura 6.6, la conexión B(1) llega por el puerto 2' del conmutador SW1, sale por el puerto 1 del mismo, entran por el puerto 1' del conmutador SW2 y sale por el puerto 2 de este último —a este puerto lo denotamos SW2(2)—

En esta configuración, el enlace troncal constituye el enlace de cuello de botella de las cinco conexiones establecidas, por lo que la tasa equitativa en el sentido *max-min* de cada una de las cinco conexioneses es 1/5 de la capacidad del enlace.

Se contemplan tres subconfiguraciones de dos conmutadores, según el rango de distancias de los enlaces troncales y de acceso: escenario de distancias LAN, de distancias MAN y de distancias WAN. Cada subconfiguración pretende caracterizar el rango de retardos de realimentación de una red sencilla ATM que dé soporte respectivamente a una red de área local, de área metropolitana y de área extendida. Las distancias de los enlaces en cada caso se muestran en la tabla 6.3.

Escenario	*Acceso*	*Troncal*
LAN	D1=D2=...=D10=0.2 Km	D11=2 Km
MAN	D1=D6=50 Km D2=D7=20 Km D3=D8=10 Km D4=D9=5 Km D5=D10=1 Km	D11=50 Km
WAN	ídem MAN	D11=1000 Km

Tabla 6.3. VALORES DE DISTANCIAS PARA LAS SUBCONFIGURACIONES DE DOS CONMUTADORES

6.2.4.2 Configuración genérica de equidad GFC1

La configuración GFC1 se ilustra en la figura 6.7.Fue propuesta por Simcoe (1994). La capacidad de cada enlace de acceso es de 150 Mbit/s. Está compuesta de 5 conmutadores conectados en línea por 4 enlaces troncales de distintas capacidades.

Existen 6 grupos de conexiones:

- cuatro de ellos, los grupos D, F, C y E, atraviesan un único enlace que, por tanto, constituye su enlace de cuello de botella;
- el grupo A entra en el primer conmutador y sale en el penúltimo conmutador y tiene su cuello de botella en el enlace que une el primer y el segundo conmutador, esto es, en SW1(1);
- el grupo B entra en el segundo conmutador y sale en el último conmutador y tiene su cuello de botella en el enlace que une el penúltimo y el último conmutador, esto es, en SW4(1);

El enlace bajo estudio es el enlace entre el antepenúltimo y el penúltimo conmutador, esto es, el enlace del puerto SW3(1). Este enlace está atravesado por tres grupos de conexiones: un primer grupo de conexiones (el grupo C) para el cual es cuello de botella; otro grupo (el grupo A) cuyo cuello de botella es anterior al enlace bajo estudio; y un tercer grupo (el grupo B) cuyo cuello de botella es posterior al enlace bajo estudio.

La configuración GFC1 es muy adecuada para evaluar el grado de aproximación al criterio de equidad *max-min* en la asignación de ancho de banda. La asignación teórica se muestra en la tabla 6.4, en donde el 100% de la capacidad de los enlaces se asigna a las conexiones ABR. Como se ha discutido en la sección 5.3.2, si se reservase una fracción igual a 1-*Target Utilization* en la capacidad de los enlaces, la asignación equitativa *max-min* sería la que se muestra en la tabla 6.5. No obstante, esta no es la asignación que resulta de la aplicación del algoritmo de control de congestión expuesto en la sección 5.3.2, sino que la asignación equitativa *max-min* resultante sería la que se muestra en la tabla 6.6. A esta asignación la denominaremos equitativa *max-min* efectiva.

Los enlaces de acceso tienen una distancia de 0.2 Km, mientras que los enlaces troncales son de 50 Km, esto es, se ha escogido un escenario de distancias MAN.

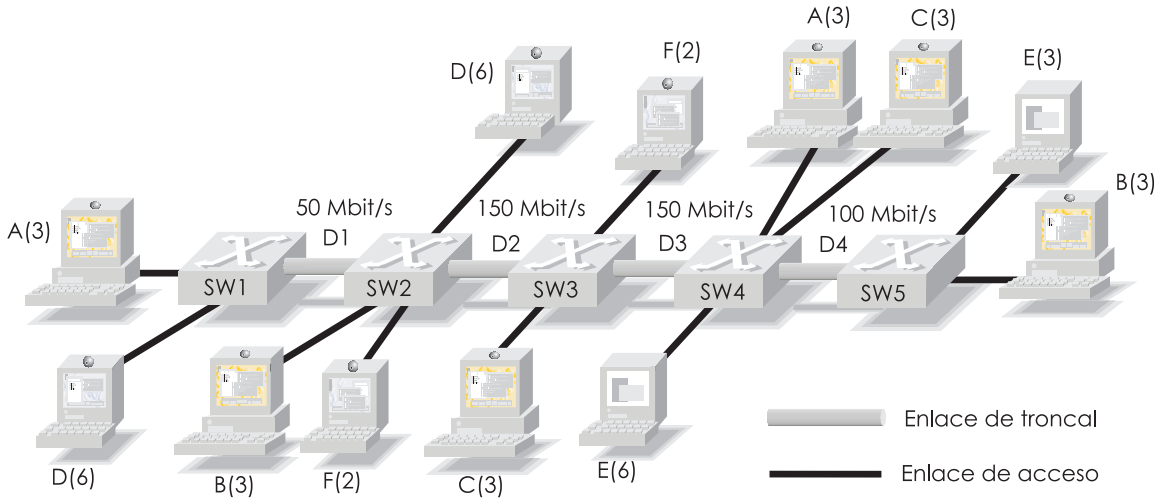

Figura 6.7. CONFIGURACIÓN *Generic Fairness Configuration 1*

Grupo	*Fracción* max-min
A	1/9 de 50 = 5.56 Mbit/s
B	1/9 de 100 = 11.11 Mbit/s
C	1/3 de (150 - 3 · 5.56 - 3 · 11.11) = 33.33 Mbit/s
D	ídem. de A
E	ídem. de B
F	1/2 de (150 - 3 · 5.56 - 3 · 11.11) = 50 Mbit/s

Tabla 6.4. VALORES TEÓRICOS DE TASA EQUITATIVA *max-min* DE CADA CONEXIÓN EN GFC1

Grupo	*Fracción* max-min
A	1/9 de 50 · 0.9 = 5 Mbit/s
B	1/9 de 100 · 0.9 = 10 Mbit/s
C	1/3 de (150 · 0.9 - 3 · 5 - 3 · 10) = 30 Mbit/s
D	ídem. de A
E	ídem. de B
F	1/2 de (150 · 0.9 - 3 · 5 - 3 · 10) = 45 Mbit/s

Tabla 6.5. VALORES TEÓRICOS DE TASA EQUITATIVA *max-min* DE CADA CONEXIÓN EN GFC1 CON CAPACIDAD DE TRANSMISIÓN REDUCIDA

6.2.4.3 Configuración genérica de equidad GFC2

La configuración GFC2 se ilustra en la figura 6.8. También fue propuesto por Simcoe (1994). La capacidad de cada enlace de acceso es de 150 Mbit/s. Está compuesta de 7 conmutadores conectados en línea por 6 enlaces troncales de distintas capacidades.

Existen 8 grupos de conexiones:

- seis de ellos, los grupos D, E, F, H, C y G atraviesan un único enlace que, por tanto, constituye su respectivo enlace de cuello de botella;
- el grupo A sale en el penúltimo conmutador, si bien cada una de sus tres conexiones entra en un conmutador distinto de los tres primeros; el grupo tiene su cuello de botella en el enlace que une el tercer y el cuarto conmutador, esto es, SW3(1);
- el grupo B sale en el último conmutador, pero sus tres conexiones entran en el primer, segundo y cuarto conmutador, respectivamente; el grupo tiene su cuello de botella en el enlace que une el penúltimo y el último conmutador, esto es, SW6(1).

El enlace bajo estudio es el enlace entre el antepenúltimo y el penúltimo conmutador, esto es, el enlace del puerto SW5(1). Al igual que en la configuración GFC1, este enlace está atravesado por tres grupos de conexiones: un primer grupo de conexiones (el grupo C) para el cual es cuello de botella; otro grupo (el grupo A) cuyo cuello de botella es anterior al enlace bajo estudio; y un tercer grupo (el grupo B) cuyo cuello de botella es posterior al enlace bajo estudio. Sin embargo, a diferencia de la configuración GFC1, cada una de las conexiones de los grupos A y B recorre un trayecto de distinta longitud. Esta disposición se conoce como *parking lot* y permite comprobar si se origina el efecto de *beat-down*. Brevemente, el problema *beat-down* aparece porque las conexiones que atraviesan muchos conmutadores tienen una probabilidad mayor de que se les notifique una reducción de tasa ACR, en comparación con aquellas conexiones que atraviesan pocos conmutadores.

La configuración GFC2 permite comprobar si la asignación de tasa es equitativa *max-min* en un entorno que es representativo como GFC1 y en el que, además, los retardos de realimentación de las conexiones de un mismo grupo —que tienen el mismo enlace de cuello de botella— son distintos. La asignación teórica de tasas se muestra en la tabla 6.7, en donde el 100% de la capacidad de los enlaces se asigna a las conexiones ABR, mientras que los valores efectivos de tasas equitativas, según definición en 6.2.4.2, se muestra en la tabla 6.8.

Los enlaces de acceso tienen una distancia de 0.2 Km, mientras que los enlaces troncales son de 50 Km, esto es, se ha escogido un escenario de distancias MAN

Grupo	*Fracción* max-min
A	(1/9 de 50) · 0.9 = 5 Mbit/s
B	(1/9 de 100) · 0.9 = 10 Mbit/s
C	(1/3 de (150 - 3 · 5 - 3 · 10)) · 0.9 = 31.5 Mbit/s
D	ídem. de A
E	ídem. de B
F	(1/2 de (150 - 3 · 5 - 3 · 10)) · 0.9 = 47.25 Mbit/s

Tabla 6.6. VALORES TEÓRICOS EFECTIVOS DE TASA EQUITATIVA *max-min* DE CADA CONEXIÓN EN GFC1

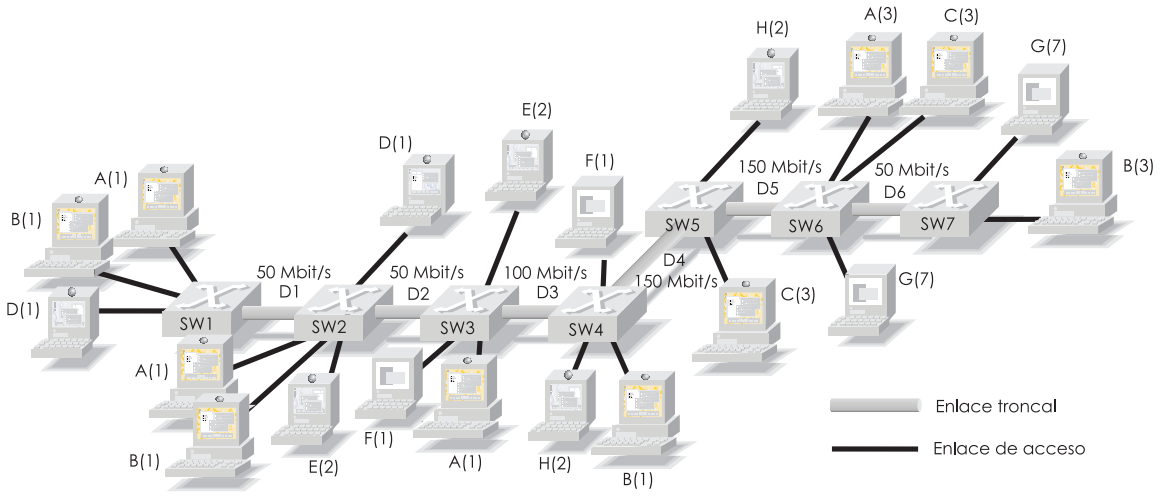

Figura 6.8. CONFIGURACIÓN *Generic Fairness Configuration 2*

Grupo	*Fracción* max-min
A	1/4 de 40 = 10 Mbit/s
B	1/10 de 50 = 5 Mbit/s
C	1/3 de (150 - 3 · 10 - 3 · 5) = 35 Mbit/s
D	(50 - 1 · 10 - 1 · 5) = 35 Mbit/s
E	1/2 de (100 - 2 · 10 - 2 · 5) = 35 Mbit/s
F	(50 - 3 · 10 - 2 · 5) = 10 Mbit/s
G	ídem. de B
H	1/2 de (150 - 3 · 10 - 3 · 5) = 52.5 Mbit/s

Tabla 6.7. VALORES TEÓRICOS DE TASA EQUITATIVA *max-min* DE CADA CONEXIÓN EN GFC2

Grupo	*Fracción* max-min
A	(1/4 de 40) · 0.9 = 9 Mbit/s
B	(1/10 de 50) · 0.9 = 4.5 Mbit/s
C	(1/3 de (150 - 3 · 9 - 3 · 4.5)) · 0.9 = 32.85 Mbit/s
D	(50 - 1 · 9 - 1 · 4.5) · 0.9= 32.85 Mbit/s
E	(1/2 de (100 - 2 · 9 - 2 · 4.5)) · 0.9 = 32.85 Mbit/s
F	(50 - 3 · 9 - 2 · 4.5) · 0.9 = 12.6 Mbit/s
G	ídem. de B
H	(1/2 de (150 - 3 · 9 - 3 · 4.5)) · 0.9 = 49.275 Mbit/s

Tabla 6.8. VALORES TEÓRICOS EFECTIVOS DE TASA EQUITATIVA *max-min* DE CADA CONEXIÓN EN GFC2

6.3 Evaluación de las prestaciones

6.3.1 Objetivos de la simulación

Los parámetros medidos en las simulaciones que se presentan a continuación son los siguientes:

1. La tasa máxima permitida para cada conexión ABR: para ello se presentará el valor instantáneo de la variable ACR de la fuente de cada conexión ABR. Este valor nos permitirá juzgar el grado de aproximación logrado por el control de flujo a una distribución equitativa en el sentido *max-min* del ancho de banda disponible en la red

2. La utilización del ancho de banda disponible en la red: para ello se medirá en cada configuración de test, la utilización de la capacidad de transmisión del enlace que se escoge como enlace bajo estudio. Esta utilización, o *throughput*, se ha medido como el cociente entre el número de células transmitidas por el enlace en un intervalo temporal de medida y la duración de este intervalo; este cociente se normaliza respecto de la capacidad de transmisión del enlace. El intervalo temporal de medida

de *throughput* escogido es 1 ms, pues permite observar con suficiente detalle la evolución del sistema para los valores de retardo de ida y vuelta y de capacidad que se han manejado en las simulaciones. Este parámetro de mérito nos permite juzgar cuán eficiente es el control de flujo que se simula, en régimen estacionario, así como evaluar la evolución del sistema en régimen transitorio, esto es, observar el efecto de una perturbación.

3. El nivel de llenado de las colas en los conmutadores: para ello se presentará el número de células en la cola del puerto de salida de los conmutadores que constituyan los cuellos de botella de las conexiones que se estudien. El valor representado es el resultado de muestrear el valor instantáneo cada 250 μs. El nivel de llenado que se ha trazado es el nivel agregado de todas las conexiones que se multiplexan en el puerto; esto se ha decidido así puesto que las colas suelen permanecer poco ocupadas, por lo que mostrar el nivel de llenado individualizado por cada conexión carece de relevancia. Este parámetro de mérito nos permitirá juzgar cuáles son las necesidades de almacenamiento en nodo que impone el mecanismo de control de flujo que se simula; cuanto mayor sea el valor máximo alcanzado por el nivel de llenado de las colas, mayor será la probabilidad de que se produzcan pérdidas y mayor será el retardo de transferencia de las unidades de datos que se entregan a la red. Estos aspectos tienen especial relevancia si hubiésemos centrado el análisis sobre el protocolo de transporte que haría uso del servicio ABR: téngase en cuenta que, cuantas más células se pierdan, mayor número de retransmisiones se generarán al nivel de capa de transporte y que, cuanto más tarden las células en atravesar la red, mayor es el tiempo que tardan los reconocimientos en llegar a la entidad emisora de capa de transporte .

Consideramos que no es relevante medir la tasa de pérdida de células, por las siguientes razones. Primero, porque la reacción de la fuente de células ante la pérdida de las células depende de las acciones que se tomen a nivel de capas superiores. Y segundo, porque la tasa de pérdida depende, en última instancia, del número de *buffers* en los nodos.

Haciendo uso de los tres parámetros de mérito anteriores, presentamos a continuación distintos escenarios de test. Estos escenarios se han diseñados con el objetivo de mostrar el grado en que el mecanismo de control de flujo presentado en el capítulo 5 cumple las propiedades que fueron discutidas en el capítulo 4:

1. La eficiencia de uso del ancho de banda disponible por parte de las conexiones ABR.

2. La escalabilidad frente a las distancias cubiertas por la red y frente al número de terminales conectados a la red.

3. La equidad del reparto de ancho de banda efectuado por la realimentación de los valores de tasa máxima permitida a cada conexión.

4. La estabilidad del sistema, en términos de ocupación de colas, *throughput* y tasas máximas permitidas, cuando se producen distintos tipos de perturbaciones.

5. La resistencia del sistema ante comportamientos no cooperativos por parte de algunas conexiones.

Las simulaciones que se presentan a continuación, salvo que se indique lo contrario, tienen los siguientes parámetros asignados por defecto:

- El tamaño máximo de las colas en cada puerto de salida de conmutador es de 1000 células; con ello, se ha evitado la pérdida de células en cualquiera de las simulaciones.
- Todas las fuentes inician la emisión de sus células en el instante de tiempo de simulación 0 ms.
- Los parámetros de funcionamiento de las fuentes ABR toman los valores por defecto presentados en la tabla 6.2 de la página 138.

Además, el algoritmo de planificación en los puertos de salida será WFQ con pesos idénticos.

6.3.2 Eficiencia de uso de ancho de banda disponible

El escenario básico de simulación lo constituye una configuración de 2 conmutadores con distancias MAN (véase la tabla 6.3 en página 142). La duración de la simulación es 40 ms.

Se han efectuado dos grupos de simulaciones:

- un primer grupo en el que se han probado diversos valores para los parámetros de funcionamiento del mecanismo de generación de la señal de realimentación propuesto:
  - el parámetro *Target Utilization* (TU), descrito en la sección 5.3.2;
  - el parámetro *MoreRateThreshold* (MR), decrito en la sección 5.3.4;
- un segundo grupo en el que se han probado diversos valores para los dos parámetros siguientes de funcionamiento de la fuente ABR:
  - el parámetro Nrm, que es el número de células entre dos células FRM emitidas más 1;
  - el parámetro RIF, que fija el porcentaje máximo de aumento de ACR con cada realimentación positiva.

Han resultado cuatro simulaciones, correspondientes a las siguientes combinaciones de valores TU, MR, Nrm y RIF, que se muestran en la tabla 6.9.

En la simulación I se evalúa el efecto del parámetro de conmutador TU sobre la tasa ACR (figura 6.9), sobre el nivel de llenado de las colas del puerto de salida del primer conmutador (figura 6.10) y sobre la utilización del enlace troncal (figura 6.11). Observamos que:

Simulación	*TU*	*MR*	*Nrm*	*RIF*
I	0.98-0.95-0.90	0.9	32	1/16
II	0.9	0.9-0.8-0.7	32	1/16
III	0.9	0.9	64-32-16	1/16
IV	0.9	0.9	32	1-1/8-1/16

Tabla 6.9. JUEGOS DE PARÁMETROS EN LAS SIMULACIONES I, II, III Y IV

- El porcentaje de utilización en régimen estacionario del enlace troncal es igual al valor TU.
- El valor de TU también determina el valor estacionario de ACR correspondiente a cada conexión, puesto que el valor estimado de tasa equitativa por cada conmutador siempre se multiplica por TU antes de ser enviado a la fuente; así para TU=0.98, el valor estacionario de ACR para cualquiera de las cinco conexiones es 0.2·0.98=0.196.
- El valor de TU impone un límite en el crecimiento inicial del valor realimentado de tasa equitativa; además, cuanto menor es el valor de TU, más rápido es el vaciado de las colas en el conmutador, puesto que, una vez alcanzados los valores de ACR en régimen estacionario, la tasa de vaciado es igual a (1-TU) multiplicado por la capacidad del enlace.

Esta última consideración nos inclina a escoger 0.9 como valor de trabajo para TU: comprometemos el *throughput* máximo obtenible de la red, a cambio de permitir un vaciado rápido de las colas. Se trata de un compomiso entre comportamiento en régimen estacionario y comportamiento en régimen transitorio.

En la simulación II se evalúa el efecto del parámetro de conmutador MR sobre la tasa ACR (figura 6.12), sobre el nivel de llenado de las colas del puerto de salida del primer conmutador (figura 6.13) y sobre la utilización del enlace troncal (figura 6.14). Observamos que:

- No se observa efecto apreciable sobre la utilización del enlace ni sobre la tasa ACR.
- El efecto sobre el llenado de la cola no es significativo.

Así pues, no siendo determinante en las prestaciones obtenibles, escogemos el valor 0.9 para el parámetro MR. De este modo, minimizamos la probabilidad de considerar erróneamente a una conexión como estrangulada y, consecuentemente, como candidata a ser retenida, según se propuso el procedimiento que se describió en la sección 5.3.4.

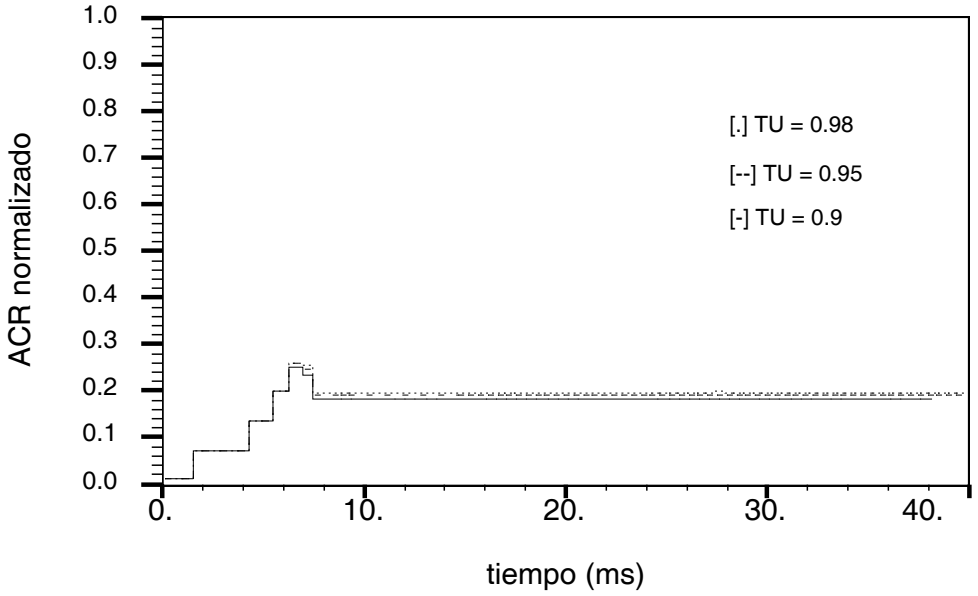

Figura 6.9. VALOR DE ACR PARA EL GRUPO DE CONEXIONES A EN SIMULACIÓN I

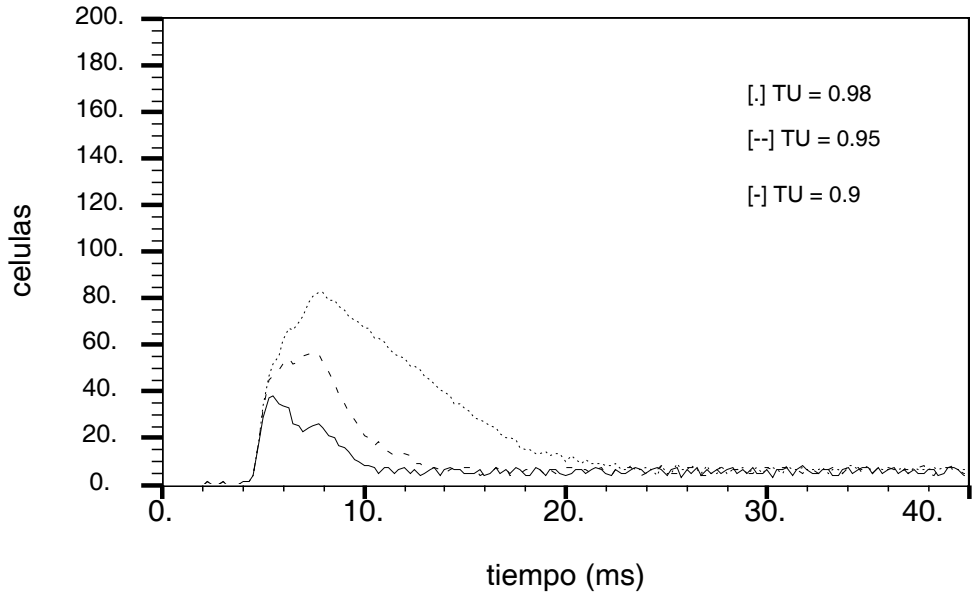

Figura 6.10. TAMAÑO DE COLA EN SW1(1) EN SIMULACIÓN I

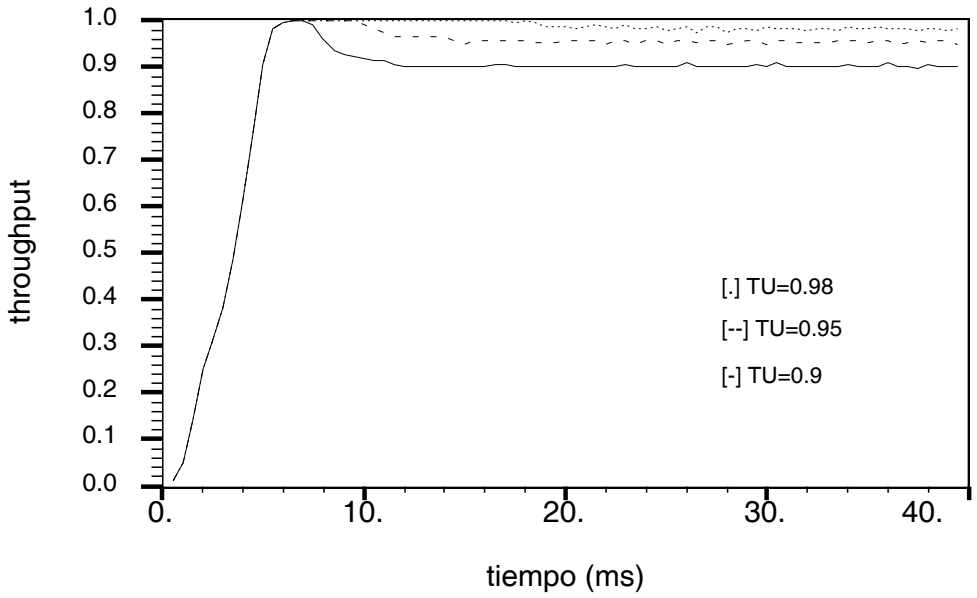

Figura 6.11. UTILIZACIÓN DEL ENLACE SW1(1) EN SIMULACIÓN I

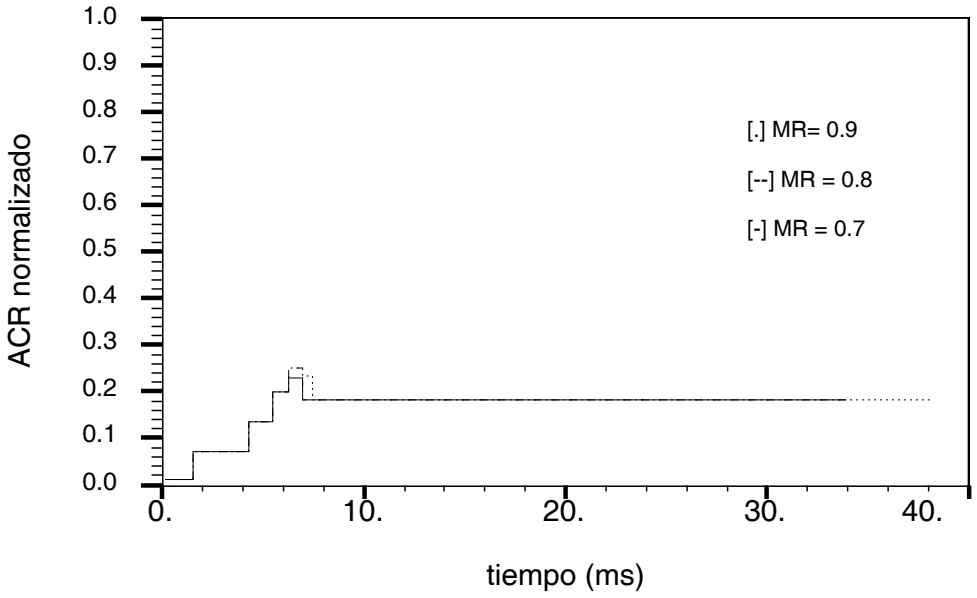

Figura 6.12. VALOR DE ACR PARA EL GRUPO DE CONEXIONES A EN SIMULACIÓN II

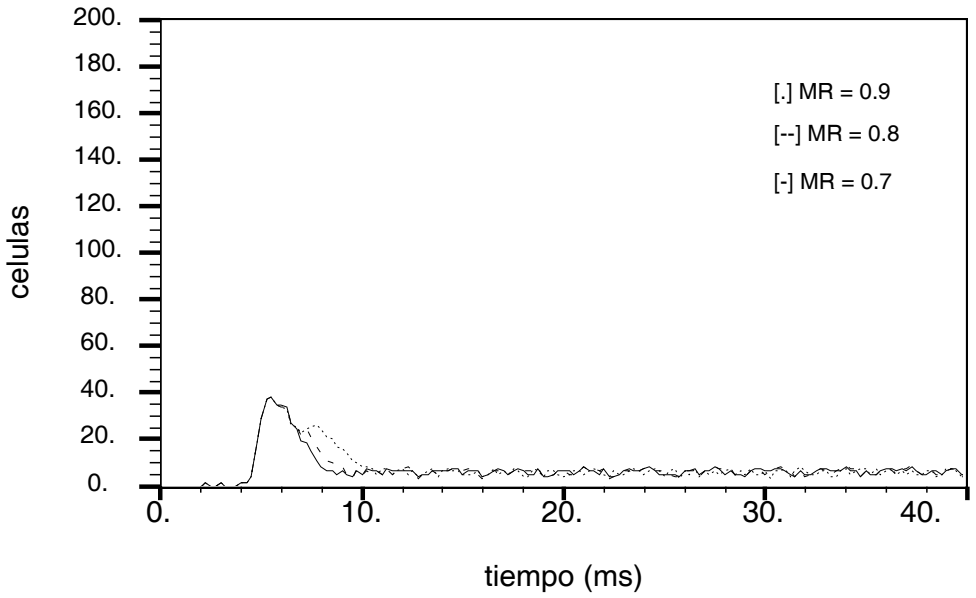

Figura 6.13. TAMAÑO DE COLA EN SW1(1) EN SIMULACIÓN II

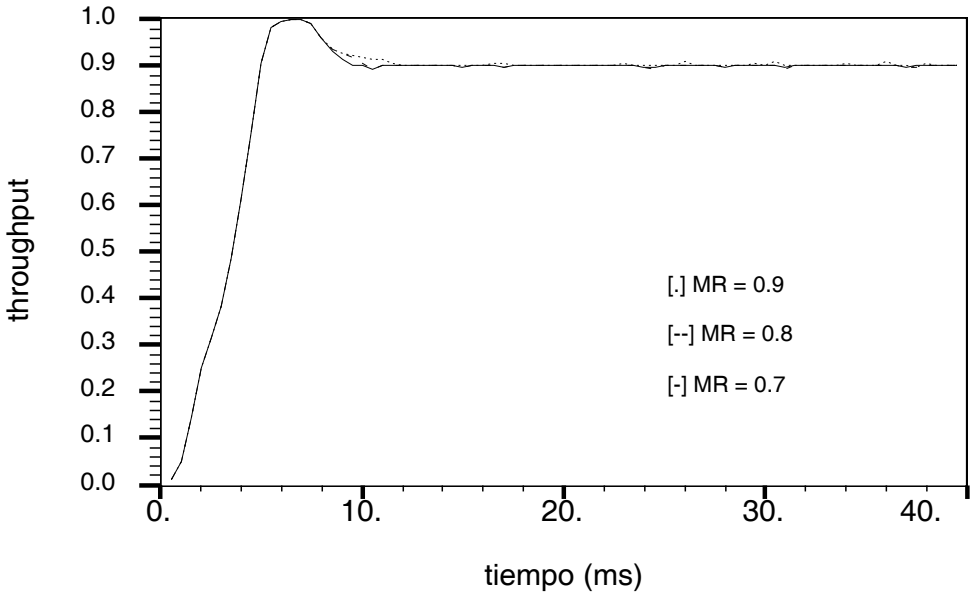

Figura 6.14. UTILIZACIÓN DEL ENLACE SW1(1) EN SIMULACIÓN II

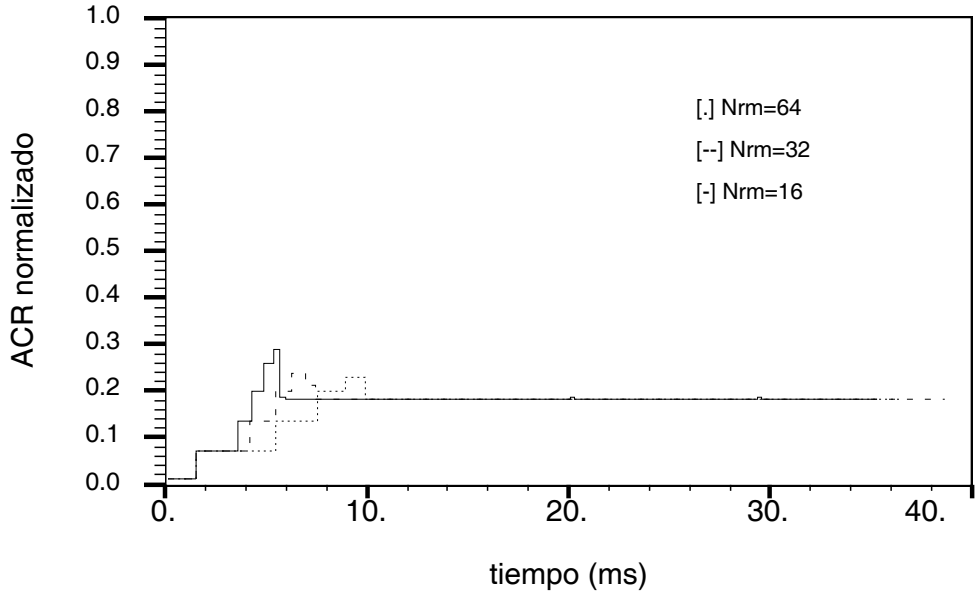

Figura 6.15. VALOR DE ACR PARA EL GRUPO DE CONEXIONES A EN SIMULACIÓN III

En la simulación III se evalúa el efecto del parámetro de fuente Nrm sobre la tasa ACR (figura 6.15), sobre el nivel de llenado de las colas del puerto de salida del primer conmutador (figura 6.16) y sobre la utilización del enlace troncal (figura 6.17). Observamos que:

- A mayor valor Nrm la frecuencia de actualización del valor ACR disminuye, lo cual se traduce en que los valores estacionarios de tasa ACR se alcanzan más tarde.
- Igualmente, a mayor valor Nrm, el desajuste entre el valor de tasa instantáneo y el valor de tasa estacionario es mayor y, por tanto, mayor es el llenado máximo alcanzado en la cola.
- Cuanto menor es el valor Nrm, más rápidamente se alcanza el valor estacionario de utilización en el enlace, aunque se aprecia una incipiente oscilación en el valor estacionario de *throughput*.

Como compromiso entre rapidez y ausencia de oscilaciones, tomamos el valor 32 para el parámetro Nrm

En la simulación IV se evalúa el efecto del parámetro de fuente RIF sobre la tasa ACR (figura 6.18), sobre el nivel de llenado de las colas del puerto de salida del primer conmutador (figura 6.19) y sobre la utilización del enlace troncal (figura 6.20). Observamos que:

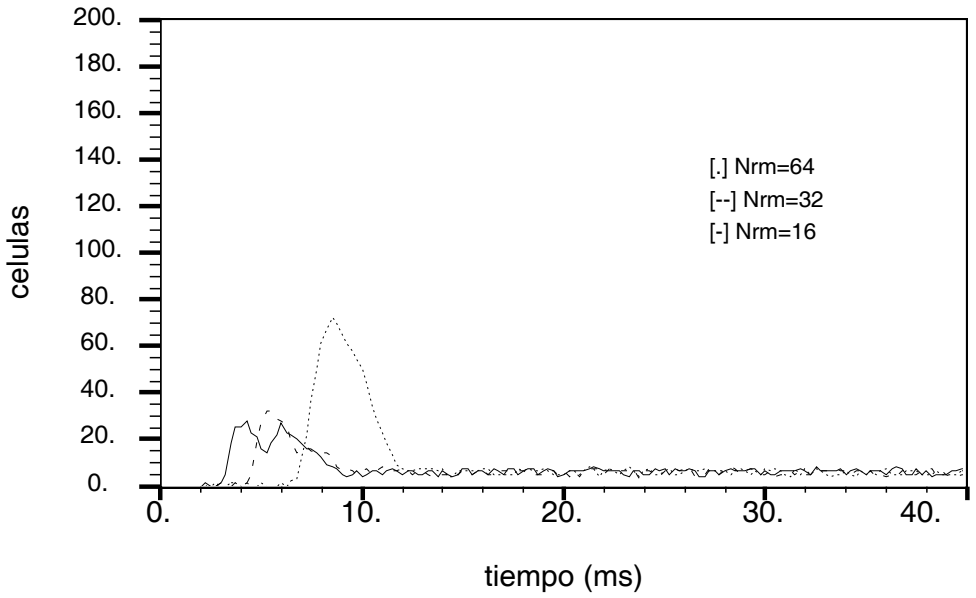

Figura 6.16. TAMAÑO DE COLA EN SW1(1) EN SIMULACIÓN III

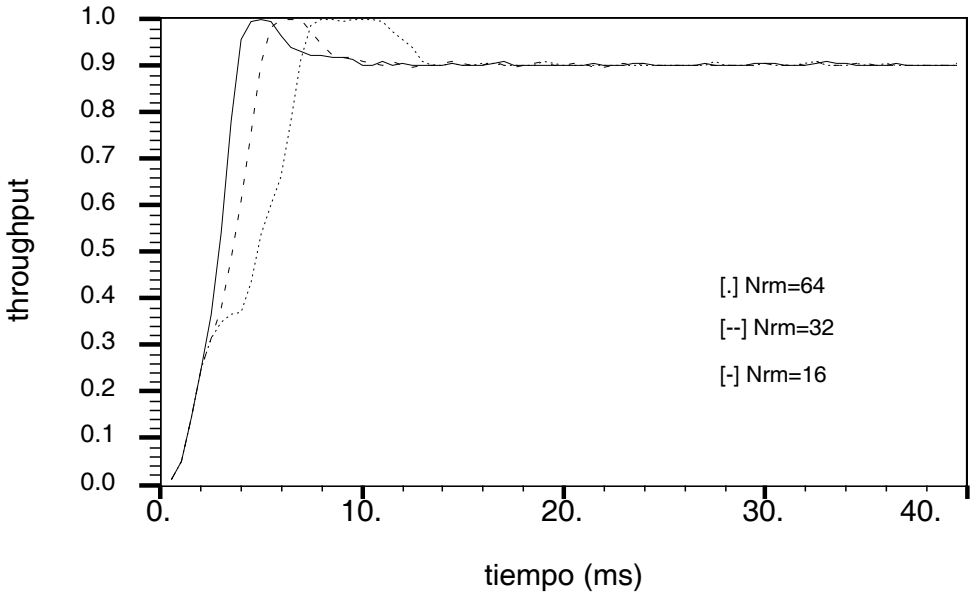

Figura 6.17. UTILIZACIÓN DEL ENLACE SW1(1) EN SIMULACIÓN III

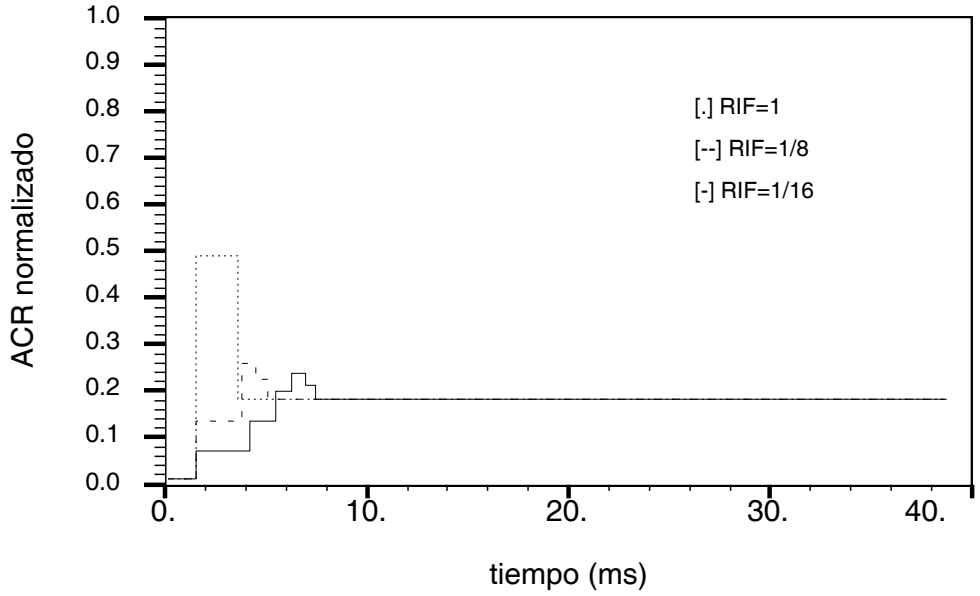

Figura 6.18. VALOR DE ACR PARA EL GRUPO DE CONEXIONES A EN SIMULACIÓN IV

- A mayor valor RIF los aumentos de tasa ACR en cada realimentación son mayores, lo cual se traduce en que los valores estacionarios de tasa ACR se alcanzan más pronto; sin embargo, las oscilaciones en el régimen transitorio son mayores en amplitud.
- Igualmente, a mayor valor RIF, el desajuste potencial entre el valor de tasa instantáneo y el valor de tasa estacionario es mayor y, por tanto, mayor es el llenado máximo alcanzado en la cola.
- Cuanto mayor es el valor RIF, más rápidamente se alcanza el valor estacionario de utilización en el enlace, aunque el llenado de las colas enmascara esta progresión rápida.

Consideramos que un valor RIF de 1/16 es garante de ausencia de oscilaciones y es un contrapeso apropiado en presencia de desajustes en la estimación.

Por último, para los valores escogidos de TU, MR, Nrm y RIF, se muestra en figura 6.21, los valores ACR de las cinco conexiones establecidas en la configuración. Observamos que todas ellas obtienen su valor equitativo *max-min*, que es 1/5 multiplicado por TU=0.9. Además, el valor equitativo de tasa permitida se mantiene estable en ausencia de perturbaciones.

En cuanto al nivel de llenado de las colas, en todos los casos (figuras 6.10, 6.13, 6.16, 6.19) se consigue mantener un nivel entre 5 y 10 células, esto es, 1 ó 2 células por conexión. Obsérvese que al mismo tiempo, la utilización del enlace en régimen estacionario es igual

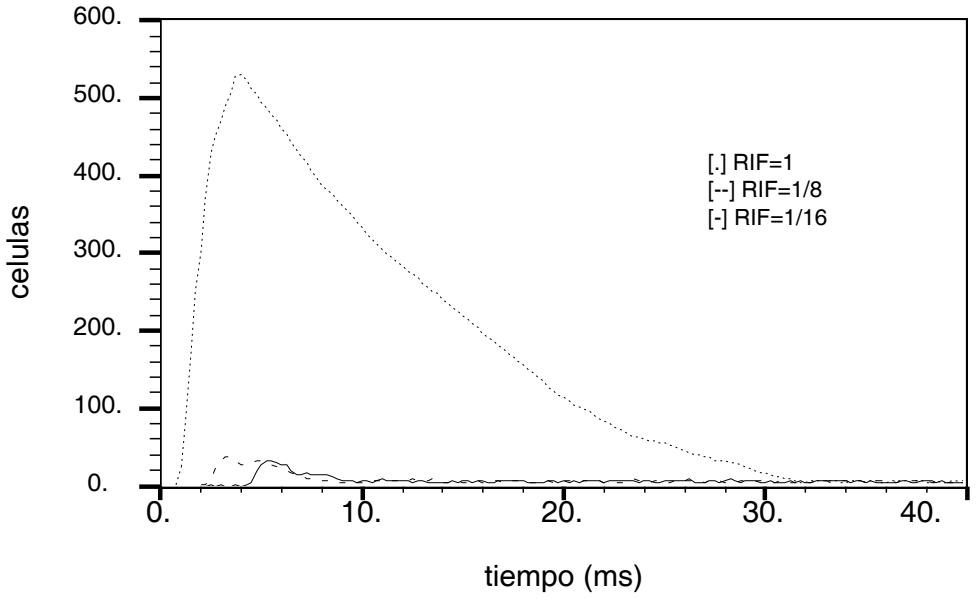

Figura 6.19. TAMAÑO DE COLA EN SW1(1) EN SIMULACIÓN IV

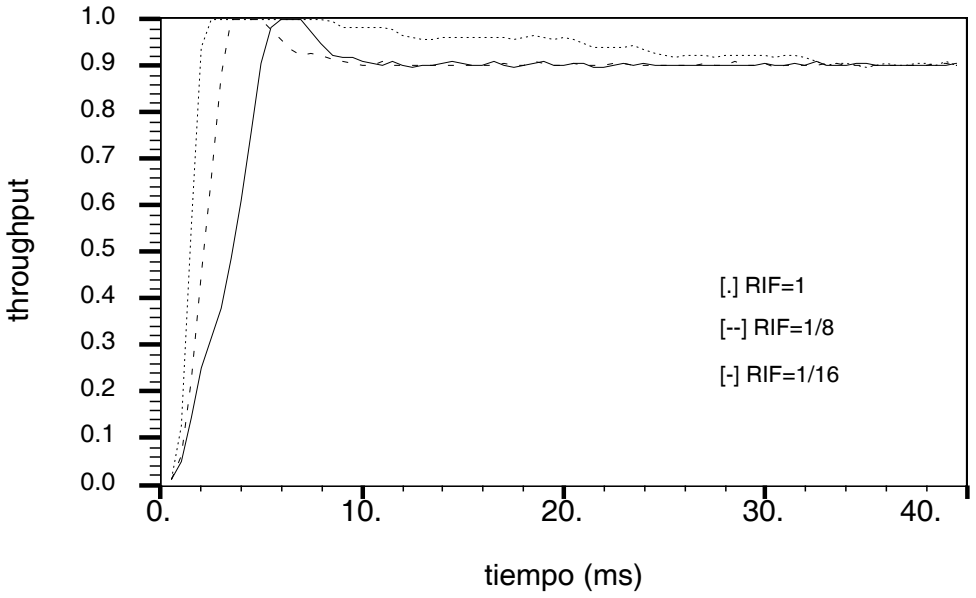

Figura 6.20. UTILIZACIÓN DEL ENLACE SW1(1) EN SIMULACIÓN IV

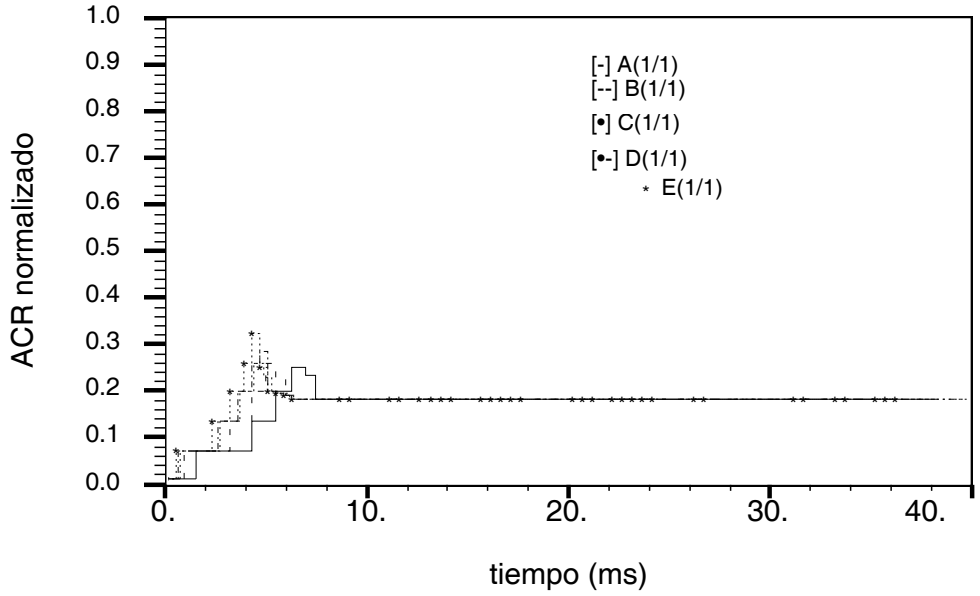

Figura 6.21. VALOR DE ACR PARA TODAS LAS CONEXIONES

al valor de TU, esto es, al 90% (figuras 6.11, 6.14, 6.17, 6.20) . Nótese que no existe contradicción entre un valor inferior a la unidad en la utilización y un valor superior a cero en el tamaño de cola, debido a que se ha incorporado un mecanismo de retención de células en los nodos que provoca el descarte de *slots* en el enlace. Además, nótese que el porcentaje de *slots* descartados es igual a (1-TU), puesto que este valor es también la probabilidad de que las conexiones estranguladas encuentren su cola vacía cuando está activo el mecanismo de retención, dado que (1-TU) es la tasa normalizada de vaciamiento de la cola en ausencia de retención.

Un último apunte acerca del nivel de llenado de las colas. En todos los casos se observa un llenado inicial de un orden de magnitud mayor que el nivel de llenado estacionario. Este comportamiento se debe a la inexactitud de la estimación de tasa equitativa durante este periodo inicial. No se ha incorporado al esquema propuesto en el capítulo 5 ningún mecanismo para evitar el comportamiento. La razón es que viene ocasionado por las particularidades de la configuración que se utilizan. En un caso más realista, como el de la simulación VIII en la sección 6.3.5, en el que las conexiones no se activan simultáneamente, veremos que el efecto sobre el llenado de las colas no es apreciable.

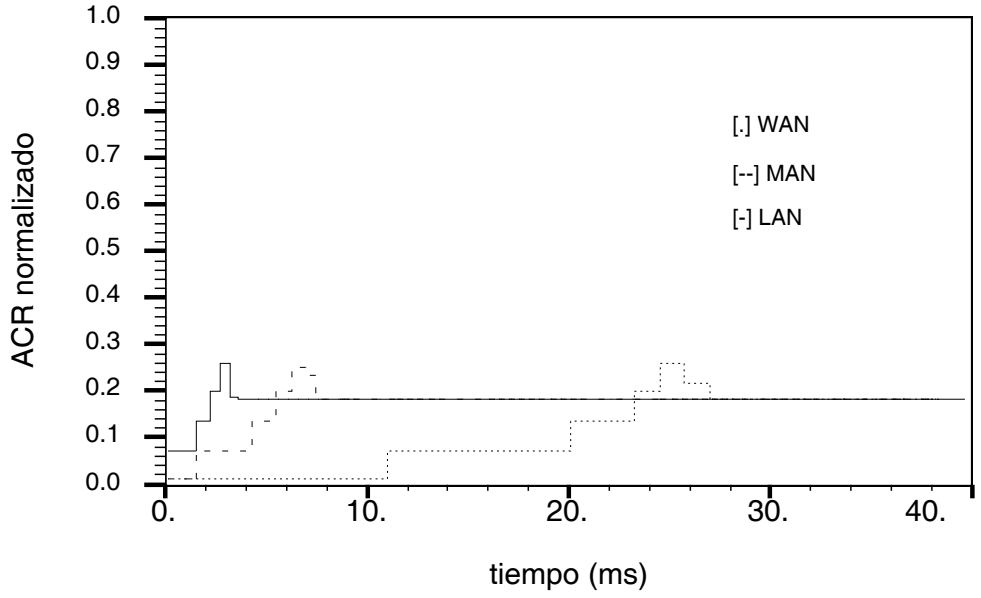

Figura 6.22. VALOR DE ACR PARA EL GRUPO DE CONEXIONES A EN SIMULACIÓN V

6.3.3 Escalabilidad temporal y espacial

Con el fin de evaluar el efecto de la distancia, esto es, del retardo del bucle de realimentación, se ha simulado la configuración de 2 conmutadores con distancias LAN y con distancias MAN (véase la tabla 6.3). La duración de la simulación V es de 40 ms.

Así, en la simulación V se evalúa el efecto de la distancia del enlace troncal sobre la tasa ACR (figura 6.22), sobre el nivel de llenado de las colas del puerto de salida del primer conmutador (figura 6.23) y sobre la utilización del enlace troncal (figura 6.24). Observamos que:

- Los valores estacionarios alcanzados de tasa ACR, de tamaño de cola y de utilización en el enlace son los mismos independientemente de las distancias cubiertas.
- Los valores estacionarios se alcanzan más tarde cuanto mayores son las distancias que se cubren.
- Dado que el desajuste del valor de tasa y del valor estacionario de tasa es mayor cuanto mayores son las distancias, el llenado máximo de la cola es mayor en distancias WAN que en distancias MAN y LAN.

Concluimos por tanto que el esquema muestra un comportamiento en régimen estacionario que es independiente de la magnitud del retardo de ida y vuelta.

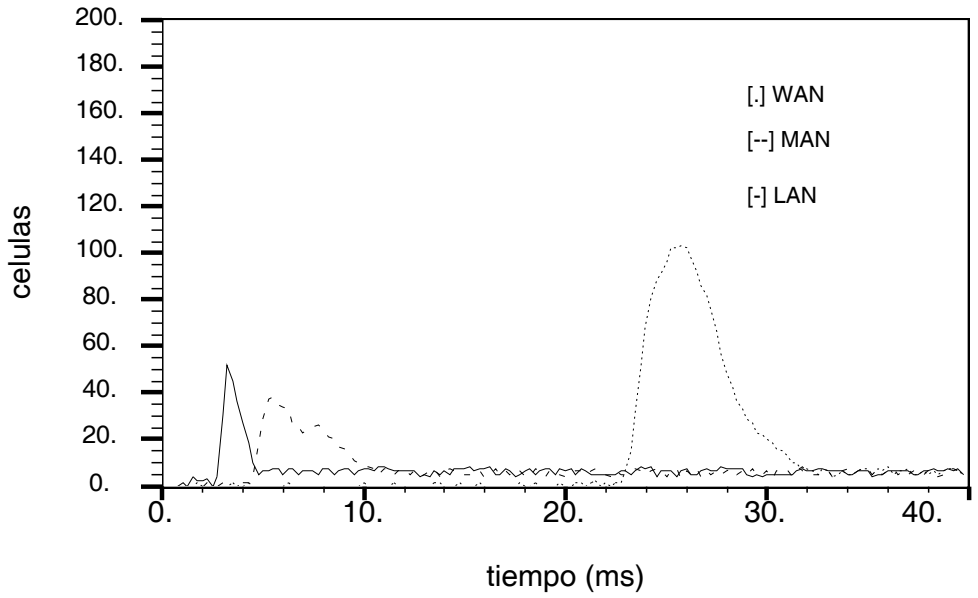

Figura 6.23. TAMAÑO DE COLA EN SW1(1) EN SIMULACIÓN V

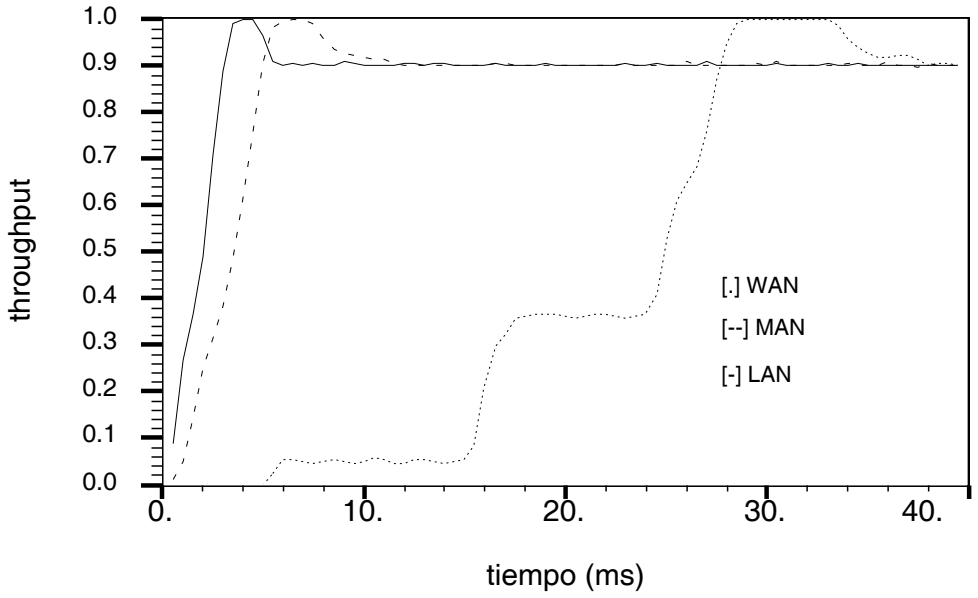

Figura 6.24. UTILIZACIÓN DEL ENLACE SW1(1) EN SIMULACIÓN V

En cuanto a la dependencia del mecanismo de control de flujo con respecto al número de conexiones establecidas en la red (escalabilidad espacial), en las secciones siguientes se simulan los escenarios GFC1 y GFC2, en los que el número de conexiones es entre 4 y 5 veces el de la configuración de 2 conmutadores. En ellos, comprobaremos que los valores en el régimen estacionario se alcanzan igualmente, lo cual demostrará que el funcionamiento del mecanismo de control no se ve afectado por el número de conexiones.

6.3.4 Equidad en la asignación de tasas permitidas de emisión

El escenario de partida para la evaluación de la equidad en la asignación de las tasas ACR es la configuración GFC1 (véase la sección 6.2.4.2 en página 142) con distancias MAN. La duración de la simulación VI es 80 ms.

El enlace bajo estudio es el correspondiente al puerto de salida SW3(1) y los grupos de conexiones bajo estudio son los tres que atraviesan el enlace. Recordemos que tales grupos son el grupo A, cuyo cuello de botella es el puerto SW1(1), el grupo B, cuyo cuello de botella es SW4(1), y el grupo C, cuyo cuello de botella es SW3(1), correspondiente al enlace bajo estudio.

Para la simulación VI, se muestran las tasas ACR de las conexiones del grupo A (figura 6.25), del grupo B (figura 6.26) y del grupo C (figura 6.27). Además, se muestra el nivel de llenado de las colas en el cuello de botella del grupo A (figura 6.28), del grupo B (figura 6.29) y del grupo C (figura 6.30). Finalmente, se presenta la utilización del enlace bajo estudio (figura 6.31). Los valores teóricos efectivos de tasa equitativa *max-min* se han dado en la tabla 6.6, que son los valores indicados en línea de puntos en las gráficas de tasas ACR.

Observamos que:

- Todas las conexiones de un mismo grupo obtienen la misma tasa máxima permitida.
- El grupo A, el grupo B y el grupo C obtienen una tasa máxima permitida igual a la tasa equitativa *max-min* efectiva.
- El valor estacionario del nivel de llenado de las colas en los cuellos de botella se mantiene en el margen de 1 ó 2 células por conexión.
- El valor de utilización del enlace bajo estudio, que es el cuello de botella del grupo C de conexiones, está por encima del valor del parámetro TU=0.9, debido a que se ha reservado un ancho de banda de drenaje igual al 10% de la suma de las tasas ACR de las conexiones del grupo C. Este valor es 3·31.5=94.5 Mbit/s, cantidad que representa 94.5/150·100=63%, por lo que el ancho de banda de drenaje es el 6.3%. Por tanto el valor esperado de utilización en el enlace es de 1-0.063=0.937, que se corresponde con el valor obtenido en la simulación.

Como segundo escenario de evaluación del grado de aproximación a la equidad *max-min* por parte del mecanismo de control de flujo propuesto, presentamos la configuración GFC2 (véase la sección 6.2.4.3 en página 144) con distancias MAN. La duración de la simulación VII es de 80 ms.

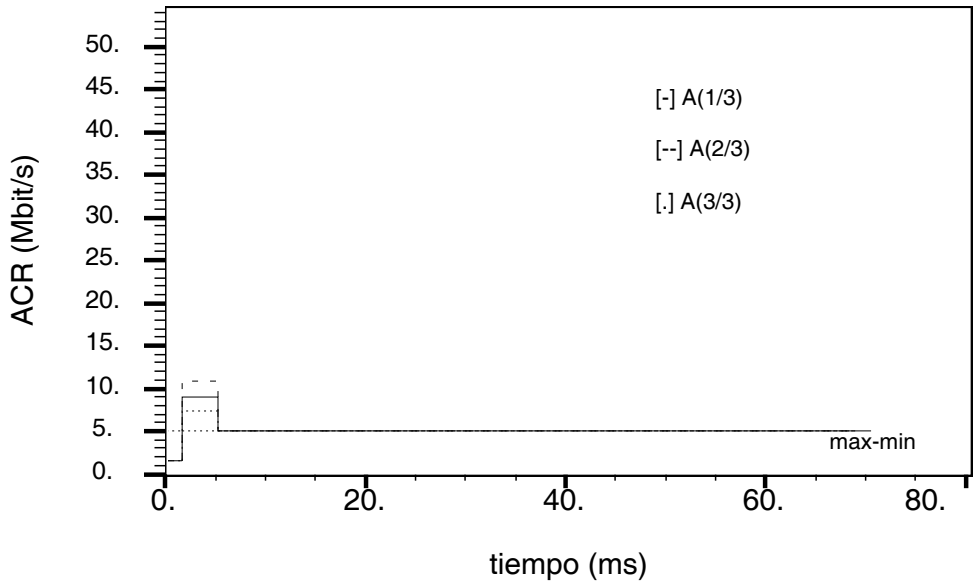

Figura 6.25. VALOR DE ACR PARA EL GRUPO DE CONEXIONES A EN SIMULACIÓN VI

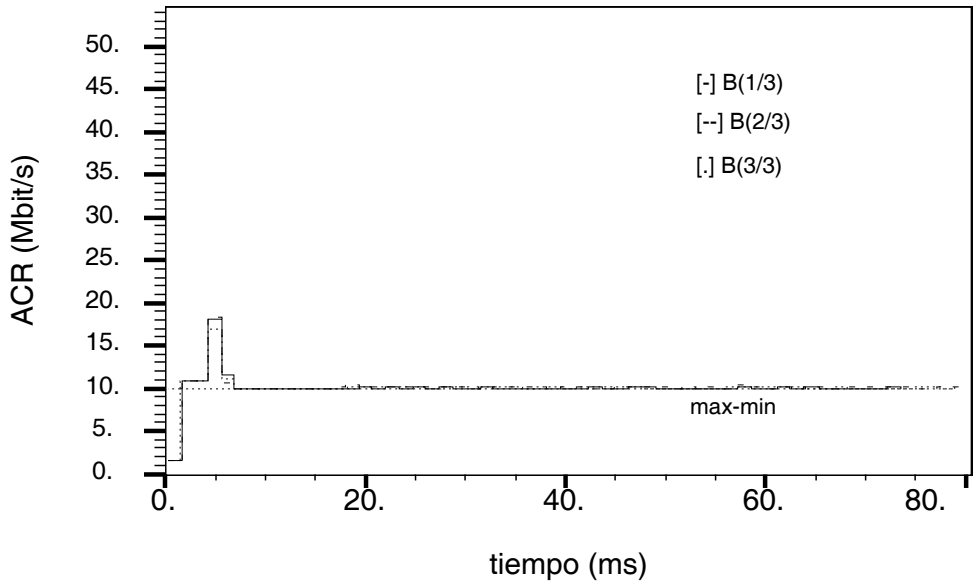

Figura 6.26. VALOR DE ACR PARA EL GRUPO DE CONEXIONES B EN SIMULACIÓN VI

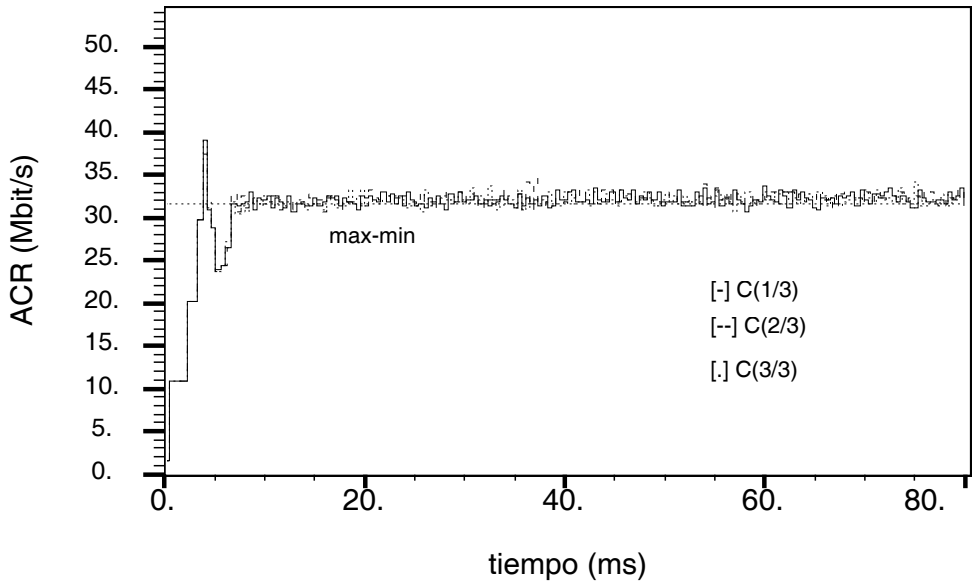

Figura 6.27. VALOR DE ACR PARA EL GRUPO DE CONEXIONES C EN SIMULACIÓN VI

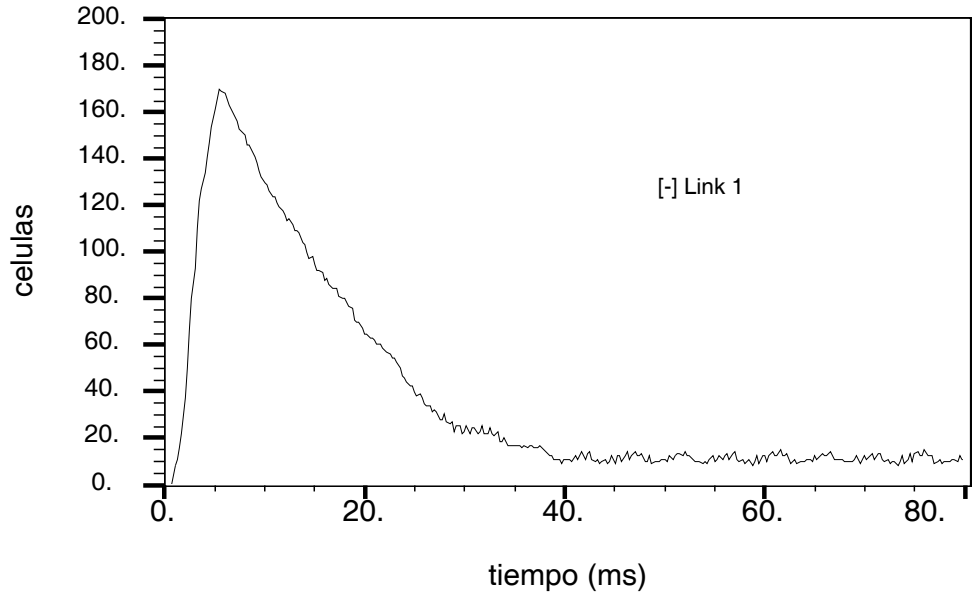

Figura 6.28. TAMAÑO DE COLA EN SW1(1) EN SIMULACIÓN VI

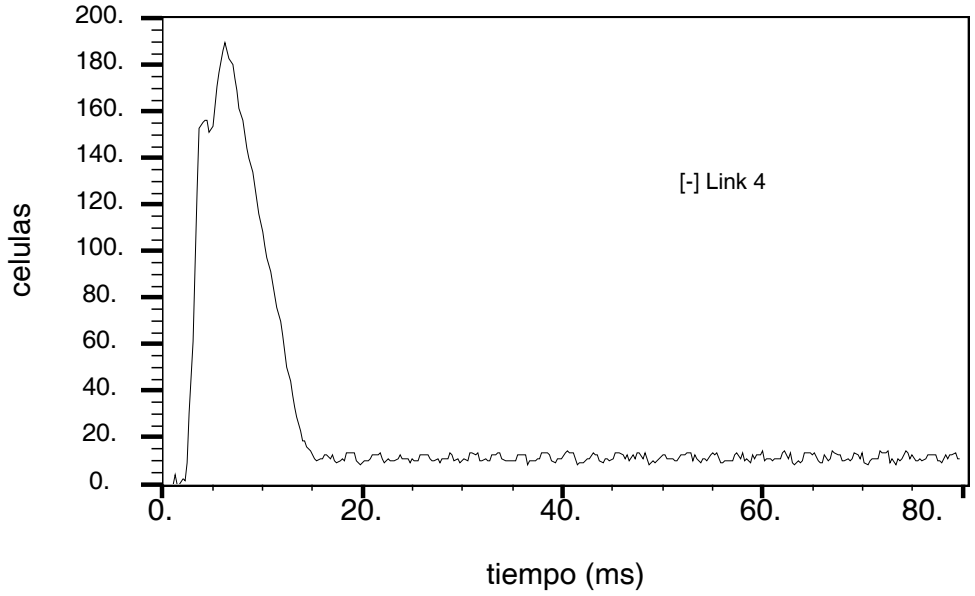

Figura 6.29. TAMAÑO DE COLA EN SW4(1) EN SIMULACIÓN VI

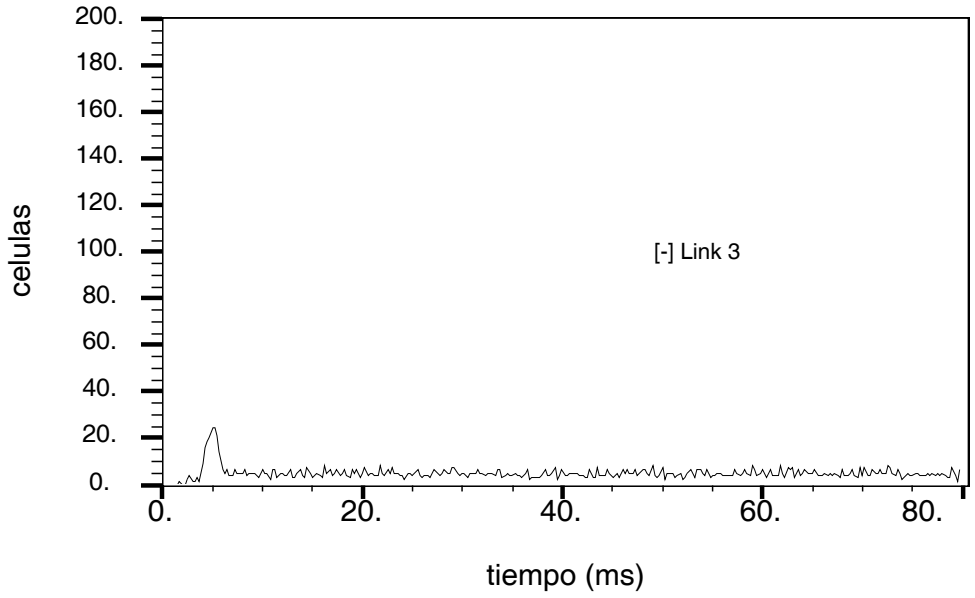

Figura 6.30. TAMAÑO DE COLA EN SW3(1) EN SIMULACIÓN VI

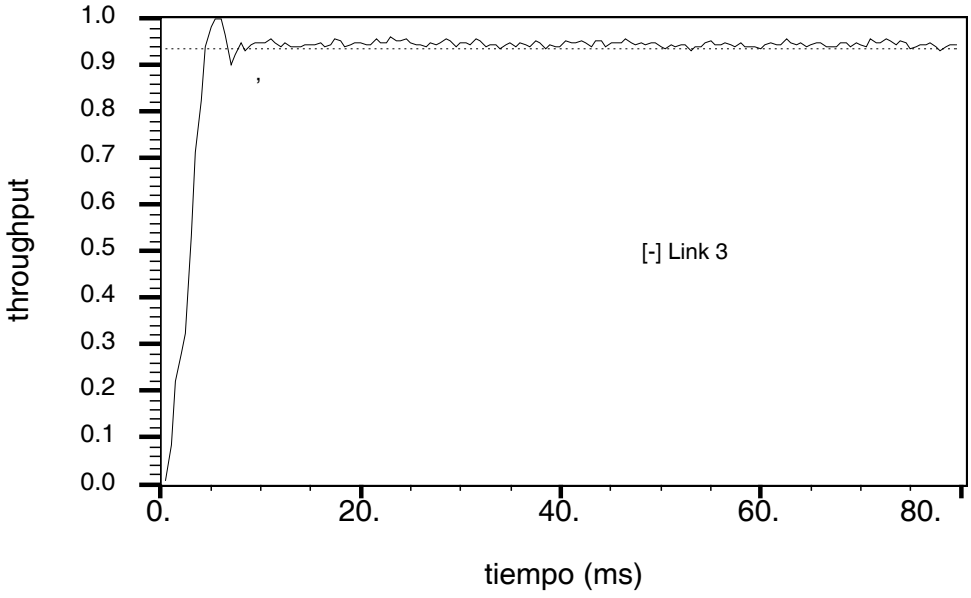

Figura 6.31. UTILIZACIÓN DEL ENLACE SW3(1) EN SIMULACIÓN VI

El enlace bajo estudio es el correspondiente al puerto SW5(1) y los grupos de conexiones bajo estudio son los tres que atraviesan el enlace. Recordemos que tales grupos son el grupo A, cuyo cuello de botella es el puerto SW3(1), el grupo B, cuyo cuello de botella es SW6(1), y el grupo C, cuyo cuello de botella es SW5(1), correspondiente al enlace bajo estudio. Además, las conexiones del grupo A entran en conmutadores diferentes, por lo que cada una tiene un retardo de realimentación diferente; así también ocurre con las conexiones del grupo B.

Para la simulación VII, se muestran las tasas ACR de las conexiones del grupo A (figura 6.32), del grupo B (figura 6.33) y del grupo C (figura 6.34). Además, se muestra el nivel de llenado de las colas en el cuello de botella del grupo A (figura 6.35), del grupo B (figura 6.36) y del grupo C (figura 6.37). Finalmente, se presenta la utilización del enlace bajo estudio (figura 6.38). Los valores teóricos efectivos de tasa equitativa *max-min* se han dado en la tabla 6.8, que son los valores indicados en línea de puntos en las gráficas de tasa ACR.

Observamos que:

- Todas las conexiones de un mismo grupo obtienen la misma tasa máxima permitida.
- El grupo A, el grupo B y el grupo C obtienen una tasa máxima permitida igual a la tasa equitativa *max-min* efectiva.
- El valor estacionario del nivel de llenado de las colas en los cuellos de botella se

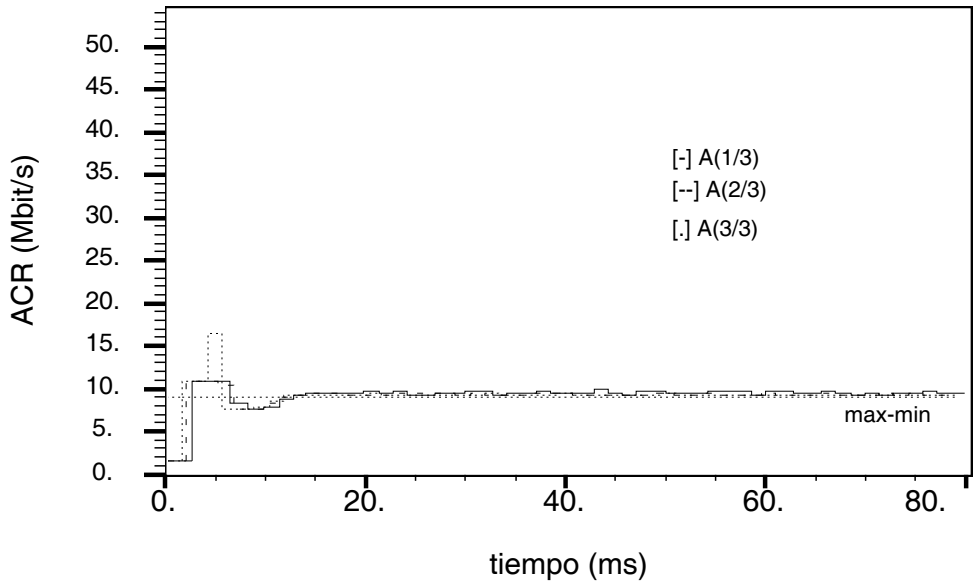

Figura 6.32. VALOR DE ACR PARA EL GRUPO DE CONEXIONES A EN SIMULACIÓN VII

mantiene en el margen de 1 ó 2 células por conexión.

- El valor de utilización del enlace bajo estudio, que es el cuello de botella del grupo C de conexiones, está por encima del valor del parámetro TU=0.9. Por un razonamiento análogo al dado para la configuración GFC1, el valor exacto de utilización efectivo es 1-0.1·(3·32.85)/150=0.9343.

En conclusión, el mecanismo de generación de señal de realimentación propuesto garantiza unos valores de tasa ACR en régimen estacionario iguales a los valores equitativos *max-min* efectivos.

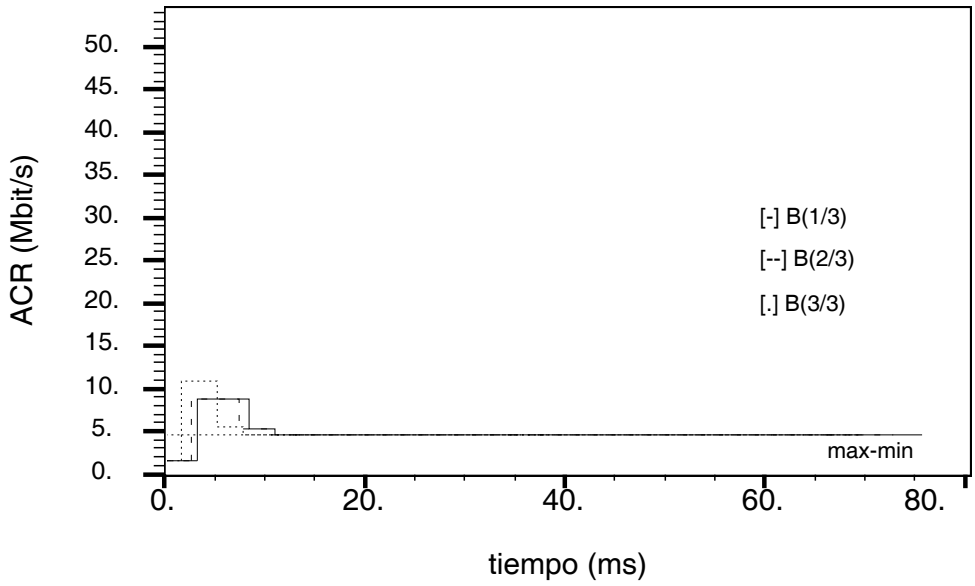

Figura 6.33. VALOR DE ACR PARA EL GRUPO DE CONEXIONES B EN SIMULACIÓN VII

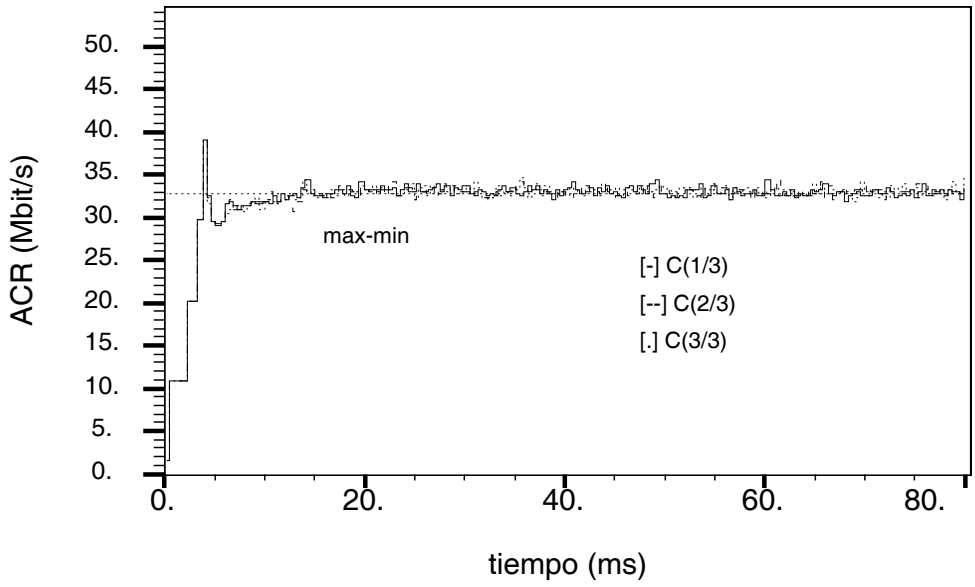

Figura 6.34. VALOR DE ACR PARA EL GRUPO DE CONEXIONES C EN SIMULACIÓN VII

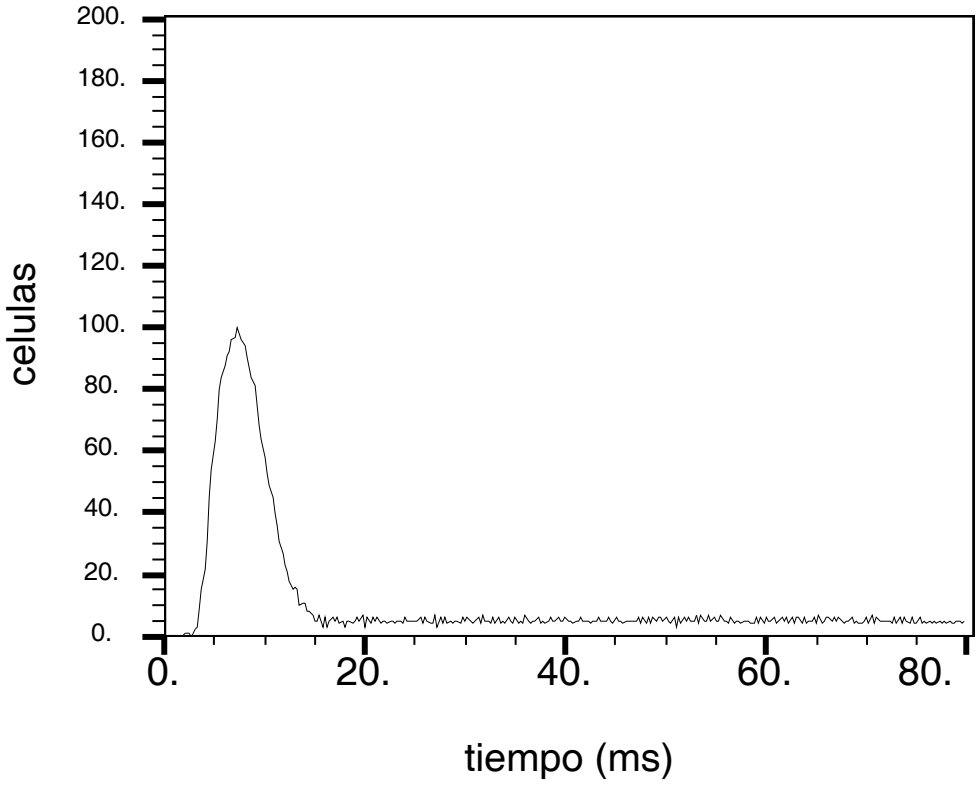

Figura 6.35. TAMAÑO DE COLA EN SW3(1) EN SIMULACIÓN VII

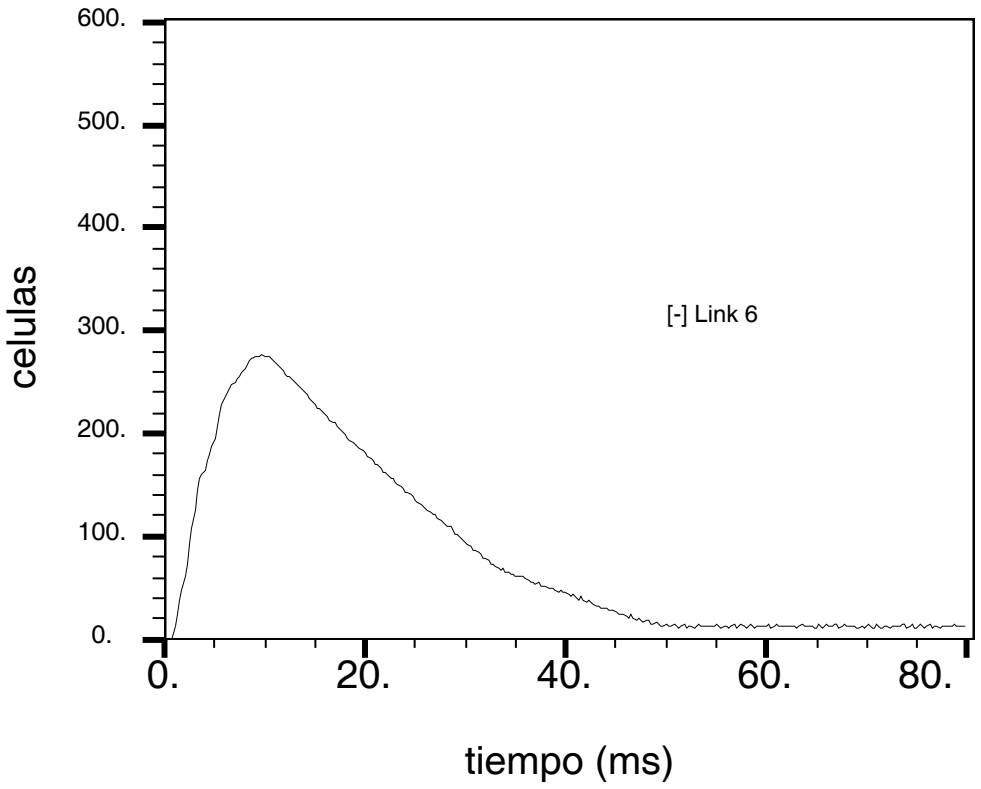

Figura 6.36. TAMAÑO DE COLA EN SW6(1) EN SIMULACIÓN VII

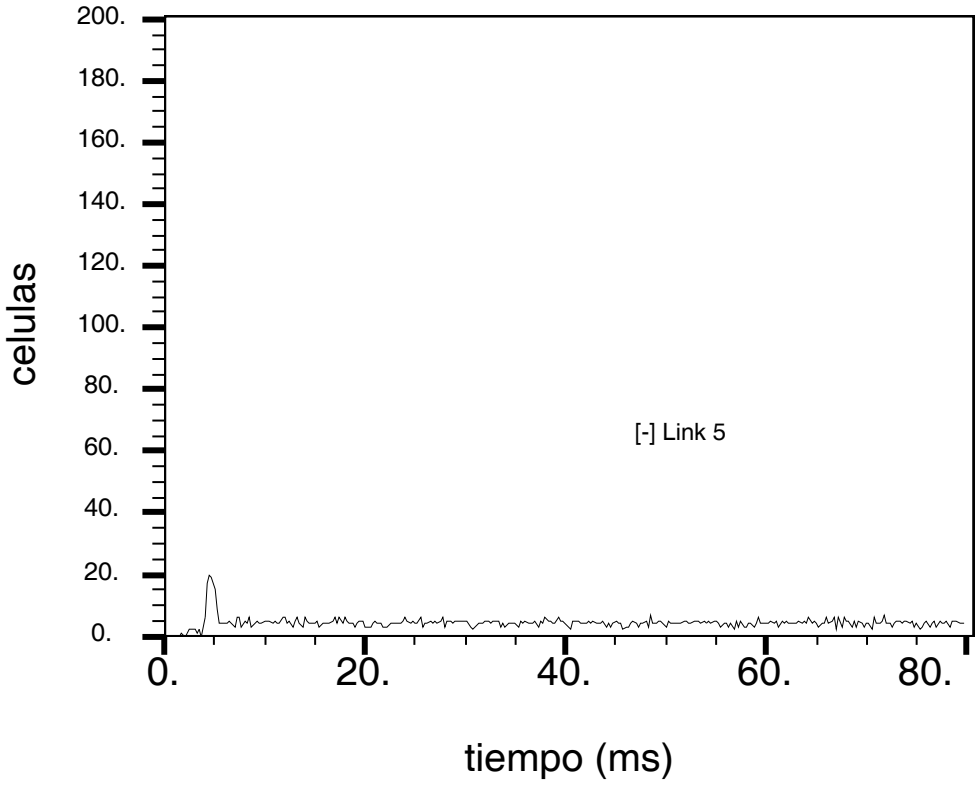

Figura 6.37. TAMAÑO DE COLA EN SW5(1) EN SIMULACIÓN VII

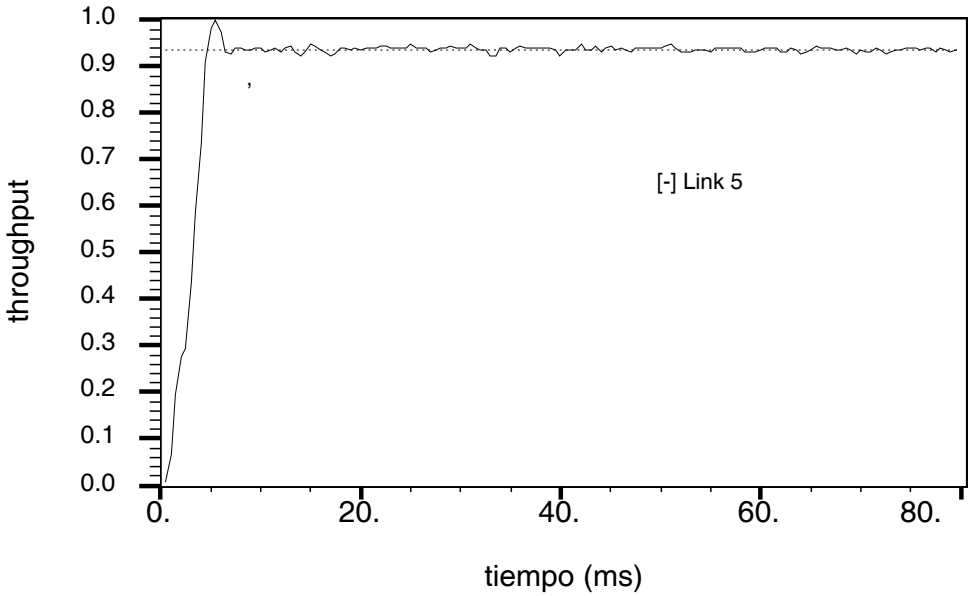

Figura 6.38. UTILIZACIÓN DEL ENLACE SW5(1) EN SIMULACIÓN VII

Conexión	*Inicio (ms)*	*Final (ms)*
A(1)	0	80
B(1)	0	100
C(1)	20	120
D(1)	40	140
E(1)	60	140

Tabla 6.10. INSTANTES DE INICIO Y FINALIZACIÓN DE EMISIÓN EN SIMULACIÓN VIII

6.3.5 Estabilidad frente a perturbaciones

Una vez evaluado el comportamiento en régimen estacionario que muestra el sistema bajo evaluación, en términos de eficiencia y de equidad, pasamos a determinar las prestaciones del mismo en régimen transitorio. Para evaluar este comportamiento, se han definido unas simulaciones en las que una vez alcanzado el régimen estacionario se han generado perturbaciones externas al sistema y se ha trazado la evolución del sistema de vuelta al régimen estacionario, en su caso.

Se han configurado dos escenarios de test para evaluar la evolución del sistema tras una perturbación:

1. un primer escenario consistente en una configuración de 2 conmutadores con distancias MAN, en donde los instantes de inicio de emisión de las cinco fuentes no coinciden;

2. un segundo escenario consistente en una configuración GFC1 con distancias MAN, en donde se producen dos variaciones tipo escalón en el ancho de banda disponible en el enlace bajo estudio

En el primer escenario, los instantes de inicio y de finalización de las conexiones que atraviesan el único enlace troncal de la configuración son los mostrados en la tabla 6.10. La duración de la simulación VIII es de 140 ms. Se consigue de este modo insertar 6 perturbaciones sobre un estado estacionario tal como el definido en la simulación I-II: tres consistentes en el establecimiento de una conexión nueva (en los instantes 20 ms, 40 ms y 60 ms) y otros tres consistentes en la liberación de una conexión establecida (en los instantes 80 ms, 100 ms y 120 ms).

Como resultado de la simulación se muestran los valores de tasa ACR de todas las conexiones (figura 6.39), del nivel de llenado de las colas del puerto SW1(1) (figura 6.40) y de la utilización del enlace troncal (figura 6.41) entre 0 ms y 120 ms. Observamos que:

- Tras cada perturbación, los valores de tasa máxima permitida alcanzan en breve un nuevo valor estacionario, correspondiente al valor de tasa equitativo *max-min* efectivo correspondiente al nuevo escenario de conexiones activas; además, no se producen oscilaciones, aunque en algunos casos sí que se produce un pequeño repunte.

- El tamaño de la cola del puerto SW1(1) no varía prácticamente durante toda la simulación.

- La entrada o salida de una conexión se refleja en la utilización del enlace troncal: una nueva conexión provoca un aumento transitorio de la utilización por encima del valor de equilibrio TU, mientras que la liberación de una conexión provoca una disminución transitoria.

Veamos con más detalles los comportamientos observados para el tamaño de la cola y para la utilización del enlace:

- Cuando se incorpora una nueva conexión, desde el momento en que llega la primera célula al puerto, se planifica su transmisión según el algoritmo WFQ y, por tanto, ya se le asigna el ancho de banda equitativo que le corresponde. Durante un intervalo de tiempo, el resto de las fuentes estarán emitiendo a la tasa equitativa anterior mientras que se servirán a la tasa equitativa actual: si no hubiésemos reservado un ancho de banda de drenaje, las colas habrían aumentado. Al haberlo reservado, observamos que, como el desajuste en las tasas de emisión no dura demasiado, este desajuste lo absorbe el enlace. Nótese que el mecanismo de retención de células continúa funcionando, por eso las colas no varían su nivel de llenado.

- Cuando una conexión abandona, desde que la última célula abandona el sistema —en el caso de WFQ, desde que deja de estar activa, esto es, desde que abandona el sistema GPS simulado— hasta que las fuentes se adaptan a la nueva tasa equitativa, el enlace está intentando transmitir a una tasa mayor que la suma de las tasas de llegada, por lo cual la utilización del enlace disminuye. El nivel de llenado de las colas no disminuirá si el mecanismo de retención de células es efectivo, lo cual depende de la duración del desajuste.

En el segundo escenario, sobre el estado estacionario definido por la simulación VI (véase la sección 6.3.4), esto es, configuración GFC1 con distancias MAN y conexiones iniciadas en el instante de tiempo 0 ms, se provocan dos perturbaciones:

1. Una disminución instantánea sobre la capacidad del enlace bajo estudio, que es SW3(1), por un factor 2, en el instante 40 ms.

2. Un aumento instantáneo en el instante 80 ms en un factor 2.

La duración de la simulación IX es 120 ms. Se consigue con este escenario modelar la variación más sencilla del ancho de banda disponible para ABR. Además, esta variación provoca un desplazamiento en el cuello de botella de uno de los grupos de conexiones.

Antes de la reducción del ancho de banda en el instante 40 ms, los valores efectivos de tasa equitativa *max-min* son los dados en la tabla 6.6. Cuando en el instante 40 ms, la capacidad del enlace SW3(1) se reduce a la mitad, esto es, a 75 Mbit/s, el cuello de botella del grupo B, que era SW4(1), pasa a ser SW3(1). Además, los nuevos valores de tasa equitativa *max-min* se muestran en la tabla 6.11.

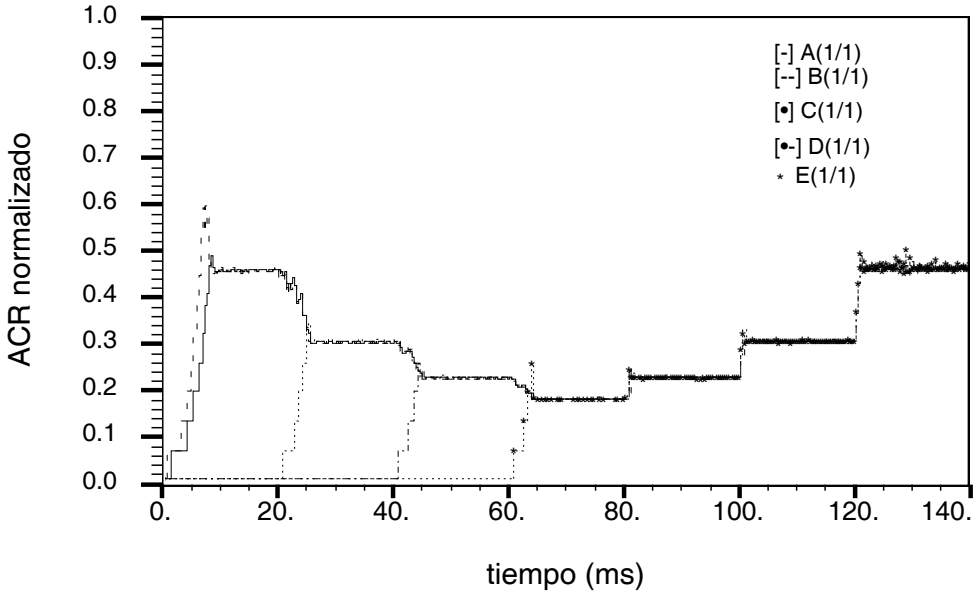

Figura 6.39. VALOR DE ACR EN SIMULACIÓN VIII

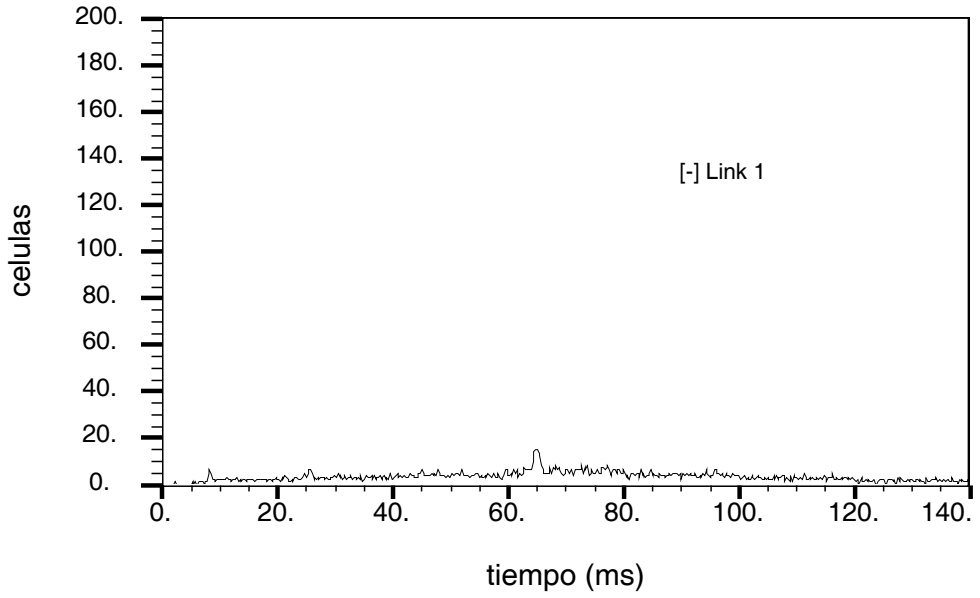

Figura 6.40. TAMAÑO DE COLA EN SW1(1) EN SIMULACIÓN VIII

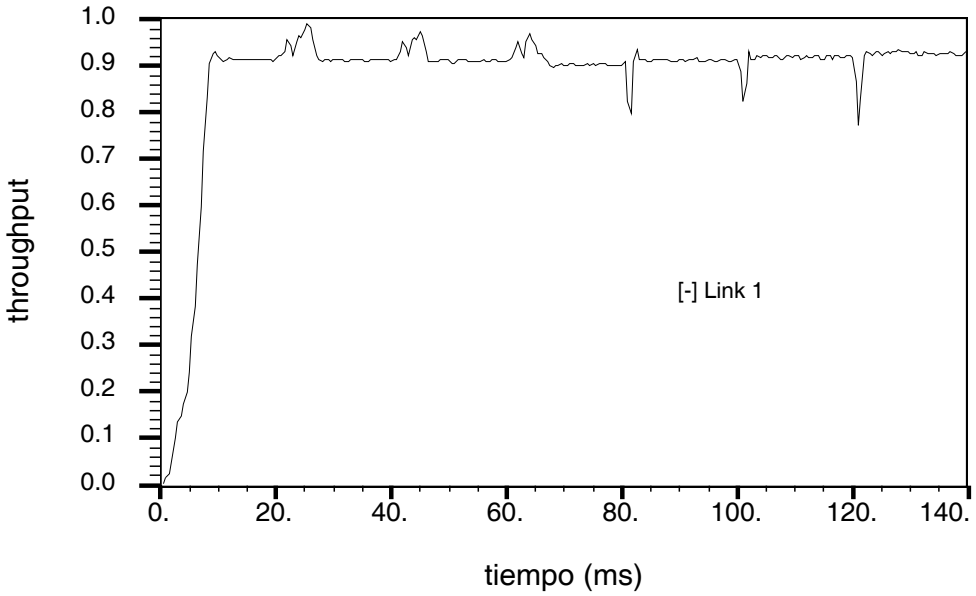

Figura 6.41. UTILIZACIÓN DEL ENLACE SW1(1) EN SIMULACIÓN VIII

Grupo	*Fracción* max-min
A	(1/9 de 50) · 0.9 = 5 Mbit/s
B	(1/6 de (75 - 3 · 5)) · 0.9 = 9 Mbit/s
C	ídem de B
D	ídem. de A
E	(1/6 de (100 - 3 · 9)) · 0.9 = 10.95 Mbit/s
F	(1/2 de (150 - 3 · 5 - 3 · 9)) · 0.9 = 48.6 Mbit/s

Tabla 6.11. VALORES TEÓRICOS EFECTIVOS DE TASA EQUITATIVA *max-min* DE CADA CONEXIÓN EN GFC1 TRAS REDUCCIÓN

Como resultado de la simulación se muestran las tasas ACR de las conexiones del grupo A (figura 6.42), del grupo B (figura 6.43) y del grupo C (figura 6.44). Además, se muestra el nivel de llenado de las colas en el cuello de botella del grupo A (figura 6.45), del grupo B (figura 6.46) y del grupo C (figura 6.47). Finalmente, se presenta la utilización del enlace bajo estudio (figura 6.48). Los valores indicados en línea de puntos en las gráficas de tasas ACR son los valores equitativos *max-min* efectivos.

Observamos que:

- Tras cada perturbación, los valores de tasa máxima permitida alcanzan en breve un nuevo valor estacionario, correspondiente al valor de tasa equitativo *max-min* efectivo correspondiente a la nueva disposición de los cuellos de botella.
- La adaptación tras la perturbación es inmediata en el caso de la reducción, esto es, en una o dos realimentaciones, mientras que en el caso del aumento la adaptación viene limitada por el crecimiento máximo permitido por el parámetro de fuente RIF.
- El tamaño de la cola del puerto SW1(1) no varía prácticamente durante toda la simulación.
- Tras la reducción de ancho de banda en el instante 40 ms, hay un llenado transitorio en la cola del puerto SW3(1), puesto que éste pasa a ser el nuevo cuello de botella de las conexiones del grupo B.
- Tras el aumento de ancho de banda en el instante 80 ms, hay un llenado transitorio en la cola del puerto SW4(1), puesto que éste pasa a ser de nuevo el cuello de botella de las conexiones del grupo B.
- La utilización del enlace bajo estudio, que es en el que se produce la modificación de capacidad, se adapta rápidamente al nuevo valor, que es 1-0.1·(6·9)/75=0.928 con respecto a 75 Mbit/s.

Concluimos por tanto que el mecanismo de generación de señal de realimentación propuesto resulta en un control de flujo que regresa al régimen estacionario eficiente y equitativo *max-min* efectivo cuando se producen perturbaciones del tipo incorporación o abandono de conexiones y del tipo disminución/aumento de ancho de banda disponible.

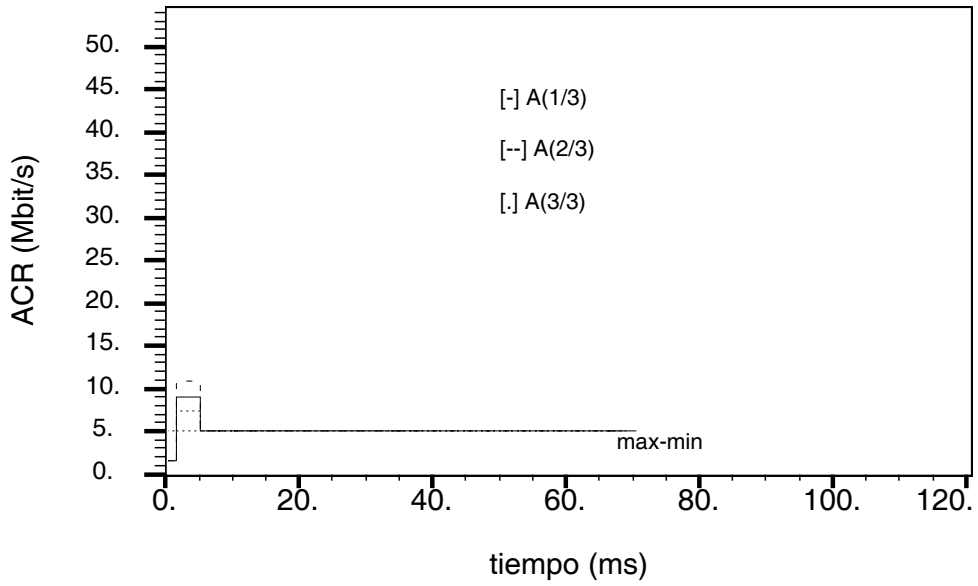

Figura 6.42. VALOR DE ACR PARA EL GRUPO DE CONEXIONES A EN SIMULACIÓN IX

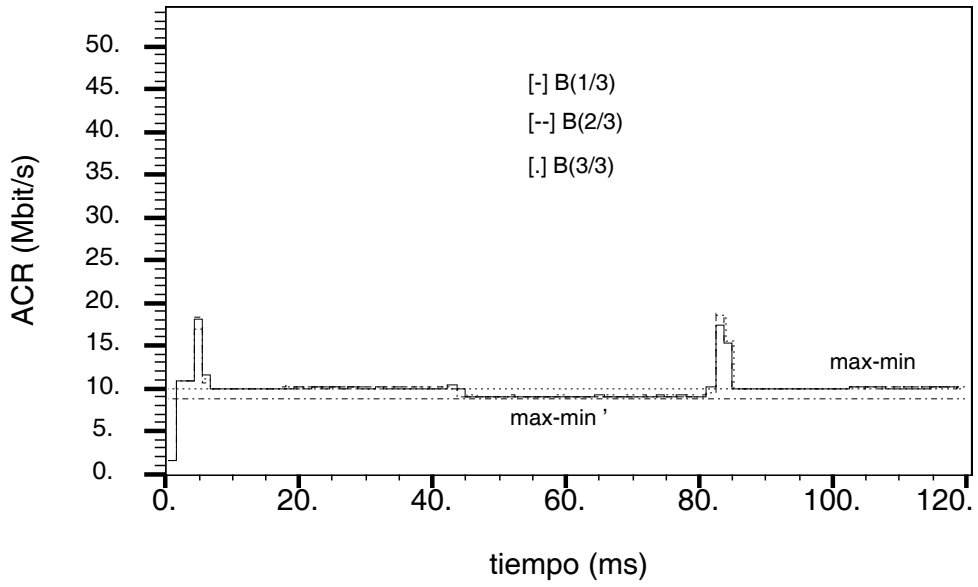

Figura 6.43. VALOR DE ACR PARA EL GRUPO DE CONEXIONES B EN SIMULACIÓN IX

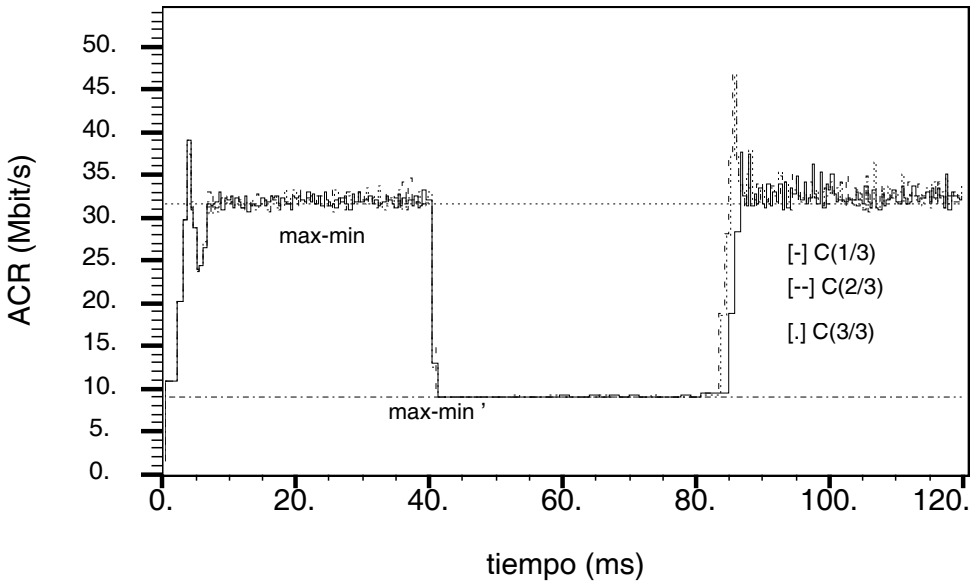

Figura 6.44. VALOR DE ACR PARA EL GRUPO DE CONEXIONES C EN SIMULACIÓN IX

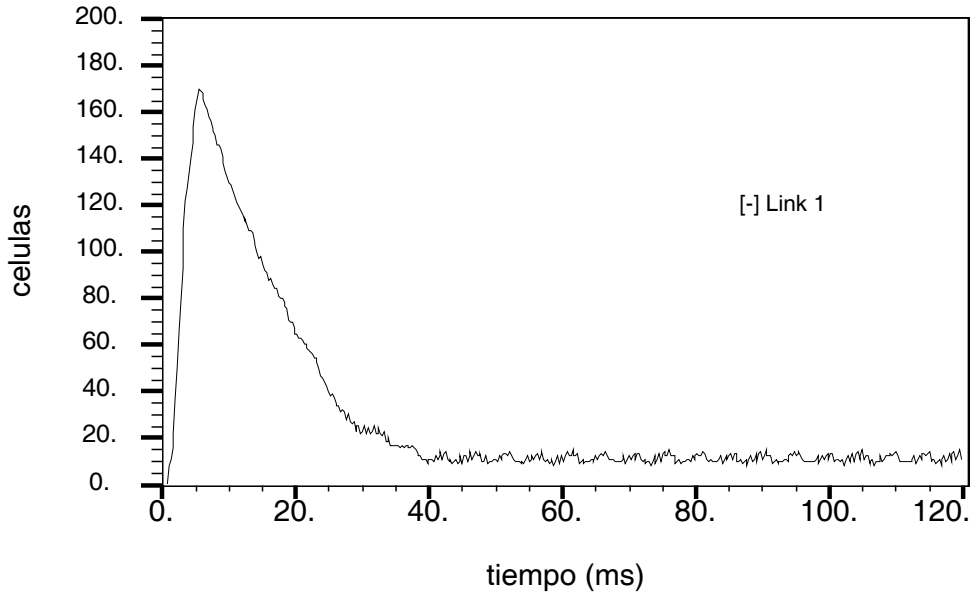

Figura 6.45. TAMAÑO DE COLA EN SW1(1) EN SIMULACIÓN IX

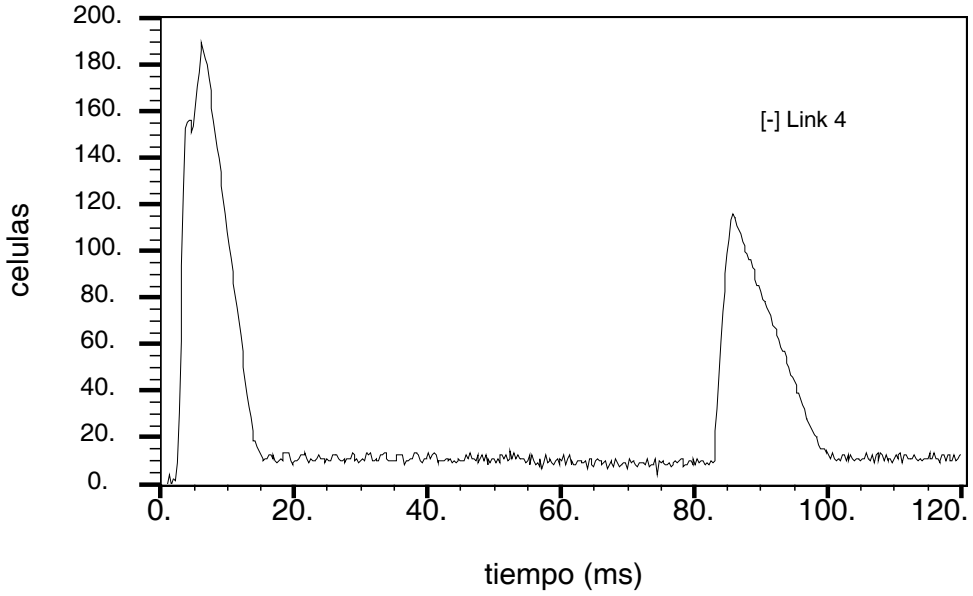

Figura 6.46. TAMAÑO DE COLA EN SW4(1) EN SIMULACIÓN IX

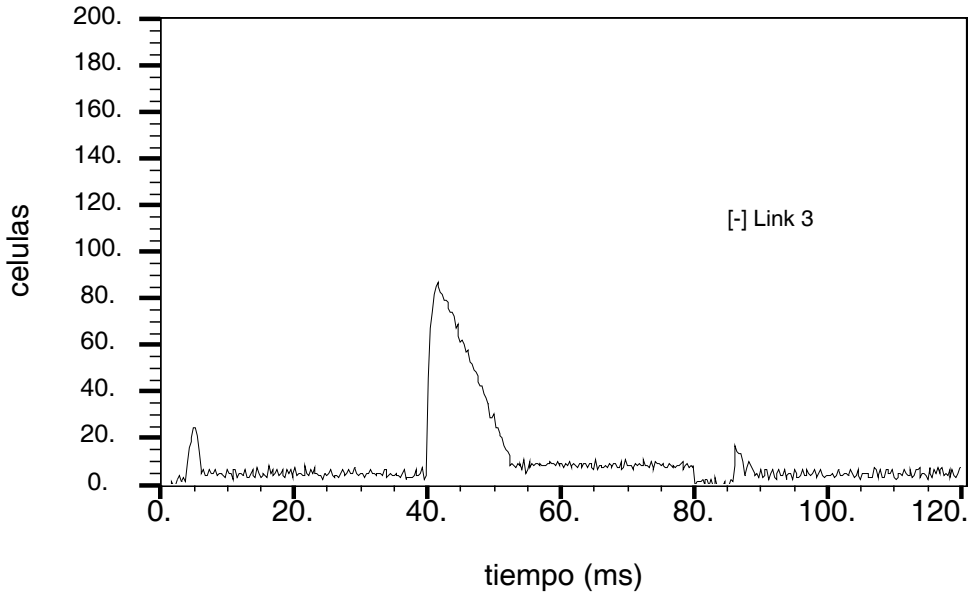

Figura 6.47. TAMAÑO DE COLA EN SW3(1) EN SIMULACIÓN IX

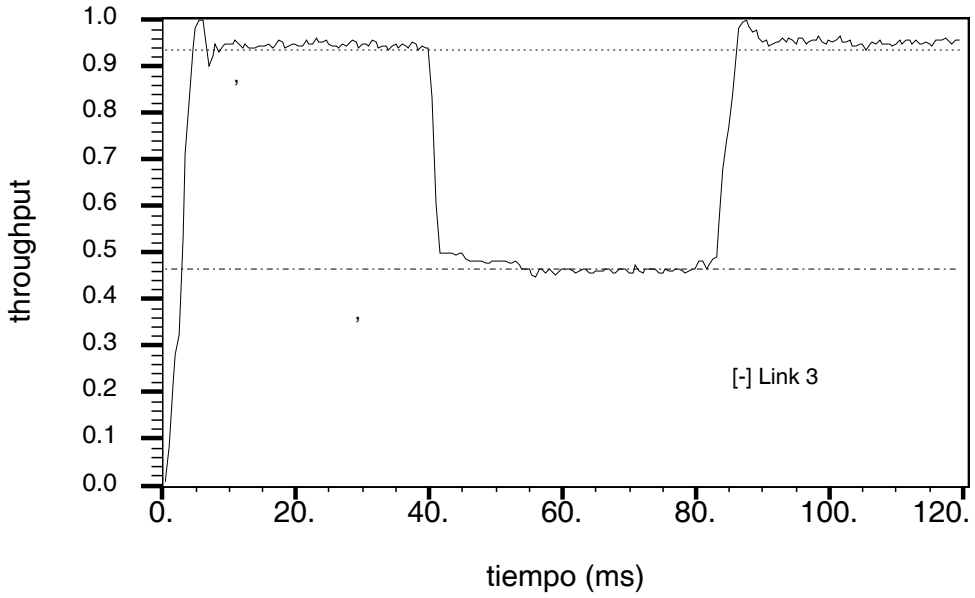

Figura 6.48. UTILIZACIÓN DEL ENLACE SW3(1) EN SIMULACIÓN IX

6.3.6 Resistencia frente a usuarios no cooperativos

Hemos afirmado en la sección 2.2 que la calidad del servicio *best-effort* que perciba un usuario no debería depender del comportamiento particular de otros usuarios; en el caso de la provisión de la clase de servicio ABR, ello se traduciría en el valor de tasa máxima permitida ACR que se realimenta a la fuente de cada conexión. Por otro lado, en las secciones 3.3.2 y 3.3.3, afirmamos que los algoritmos de planificación equitativa proporcionan protección a los usuarios en su asignación de ancho de banda. Puesto que el mecanismo de generación de señal de realimentación propuesto se basa en la estimación del ancho de banda asignado por un algoritmo de planificación equitativa, es factible la afirmación de que el mecanismo de control de flujo resultante y que da soporte de clase de servicio ABR, protege la asignación de tasa ACR realimentada a cada usuario.

Podemos aprovechar la simulación IX para mostrar la afirmación anterior. En la simulación, en el instante previo a 40 ms, las conexiones del grupo B se encuentran emitiendo a la tasa 11.11 Mbit/s, mientras que las conexiones del grupo C, a la tasa 33.33 Mbit/s. Cuando se produce la reducción de ancho de banda, las fuentes del grupo C reciben el nuevo valor de tasa equitativa *max-min* en un periodo breve de tiempo —el retardo de ida y vuelta es de 0.2 μs—, mientras que las fuentes del grupo B lo recibirán con mayor retardo —-el retardo de ida y vuelta es de 50.2 μs—: durante 0.5 ms aproximadamente, el conmutador SW3 está recibiendo células de las conexiones del grupo C a la nueva tasa 9 Mbit/s mientras que está recibiendo células de las del grupo B a la vieja tasa 10 Mbit/s. A pesar de este desajuste, como se observa en la figura 6.44, el valor de tasa máxima permitida a las conexiones del grupo C se mantiene al valor equitativo *max-min* efectivo que le corresponde. El único efecto lo sufren las conexiones del grupo B en la forma de llenado de la cola particular de cada una de sus conexiones en el puerto SW3(1), tal y como muestra la figura 6.47.

Además, se ha escogido un escenario de test para evaluar la resistencia del algoritmo de conmutador frente a usuarios no cooperativos. En tal escenario se ha empleado una configuración de dos conmutadores con distancias MAN en la que la fuente A(1) emite por encima de su tasa equitativa *max-min*. Este comportamiento no cooperativo se ha emulado asignando al parámetro MCR de esta fuente el valor 1/2 de PCR, sin notificarlo a la red. De este modo, independientemente de la realimentación que reciba la fuente A(1), ésta emitirá a la mitad del valor PCR, esto es, emitirá a 75 Mbit/s. La duración de la simulación X es de 40 ms.

Como resultado de la simulación se muestran los valores de tasa de emisión para las fuentes A(1) y B(1) en la figura 6.49, el nivel de llenado de la cola del puerto SW1(1) (figura 6.50) y el ancho de banda obtenido por la conexión A(1) y por B(1) (figura 6.51).

Observamos que:

- La tasa de emisión de la fuente no cooperativa es igual al valor PCR/2, mientras que la tasa de emisión de la fuente cooperativa se ajusta a su valor equitativo *max-min* efectivo.

- La cola del puerto de salida del conmutador se llena debido a que la fuente no cooperativa no respeta el valor realimentado por el esquema de control de flujo, que es

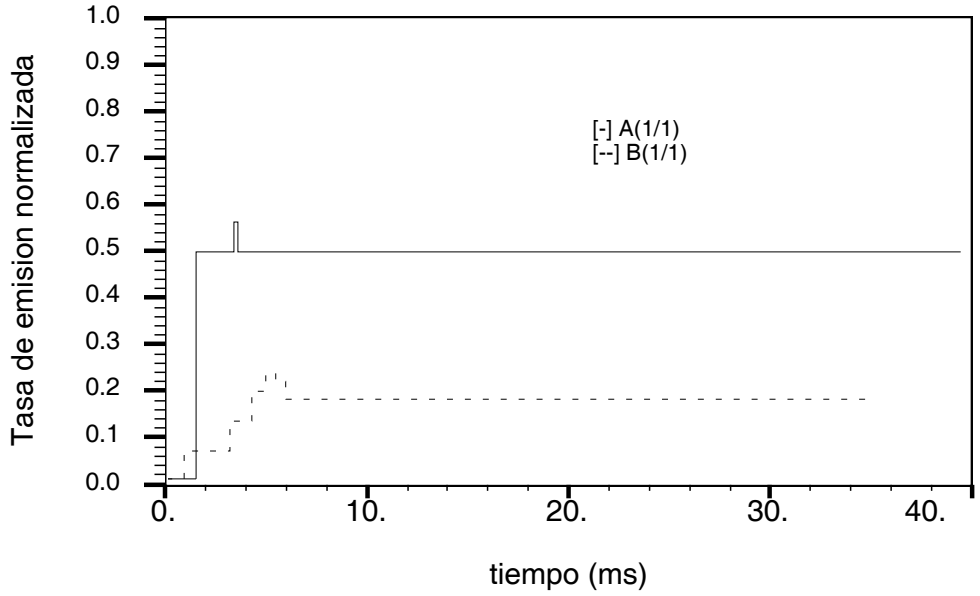

Figura 6.49. TASAS DE EMISIÓN DE CÉLULAS EN SIMULACIÓN X

el valor *max-min* efectivo.

- El ancho de banda que obtiene la fuente que coopera es igual al valor de tasa ACR que la red le comunica mediante realimentación, esto es, el valor *max-min* efectivo 0.18. Por tanto, los usuarios que sí cooperan no observan ninguna merma en la calidad de servicio que reciben a consecuencia de que otro usuario no esté respetando el valor de realimentación de la red.

- El ancho de banda que obtiene la fuente no cooperativa es igual a su valor equitativo *max-min*, que es el que le garantiza el algoritmo de planificación WFQ. Nótese que tal valor es 0.20, que no es el valor equitativo *max-min* efectivo. Ello se debe a que el algoritmo de planificación WFQ sólo fuerza anchos de banda equitativos *max-min*. La reducción para reservar ancho de banda de drenaje es un mecanismo ajeno al algoritmo de planificación, por lo que no puede forzarse su garantía en la red.

Concluimos pues que el algoritmo de conmutador propuesto en esta Tesis es resistente frente a usuarios que no respondan de acuerdo a la realimentación que genera la red.

6.4 Alternativas de diseño

En esta sección, evaluamos las prestaciones de tres alternativas de diseño al mecanismo de control de flujo que se describió en la sección 5.3.4 y que es el que ha sido evaluado en

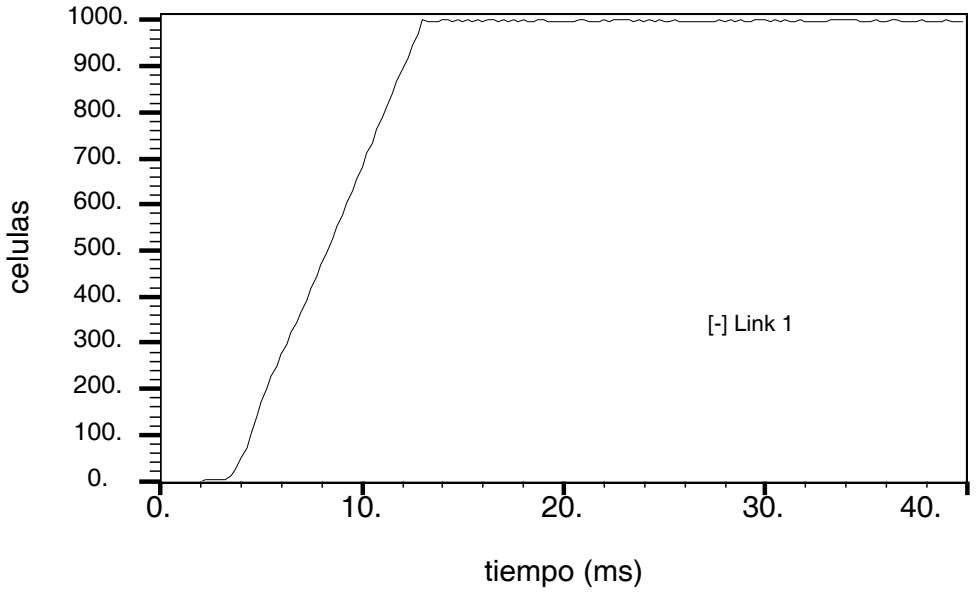

Figura 6.50. TAMAÑO DE COLA EN SW1(1) EN SIMULACIÓN X

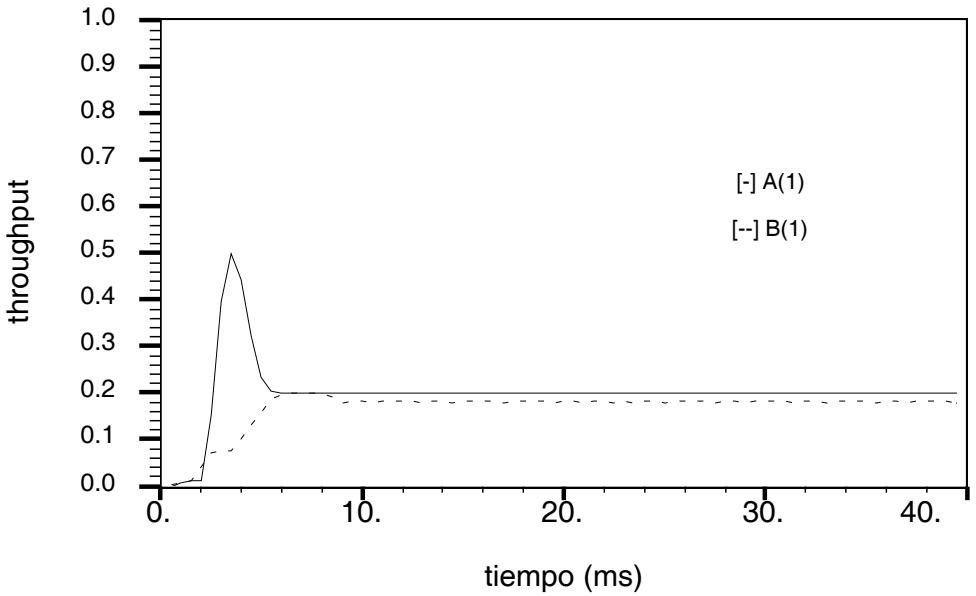

Figura 6.51. ANCHO DE BANDA OBTENIDO DEL ENLACE SW3(1) POR A Y POR B EN SIMULACIÓN X

la sección anterior.

6.4.1 Algoritmos de planificación alternativos

Hemos asumido en el mecanismo de control de flujo evaluado en la sección 6.3 que el algoritmo de planificación empleado en los puertos de salida de los conmutadores es WFQ. Vimos en la página 48 que WFQ presenta algunas dificultades de implementación. El algoritmo de planificación SCFQ fue presentado en la sección 3.3.3 como una aproximación realizable de la disciplina ideal GPS. Hemos probado el mecanismo de control de flujo propuesto con algoritmo SCFQ en los nodos; a continuación presentamos los resultados de la simulación.

Se ha escogido un escenario de simulación idéntico al de la simulación VI, esto es, una configuración GFC1 con distancias MAN. La razón de esta elección es que el algoritmo SCFQ es menos equitativo *max-min* que WFQ; por tanto, parece razonable escoger un escenario de test que evalúe el grado de equidad en la asignación de tasas ACR.

Como resultado de la simulación XI se muestran las tasas ACR de las conexiones del grupo A (figura 6.52), del grupo B (figura 6.53) y del grupo C (figura 6.54). Además, se muestra el nivel de llenado de las colas en el cuello de botella del grupo A (figura 6.55), del grupo B (figura 6.56) y del grupo C (figura 6.57). Finalmente, se presenta la utilización del enlace bajo estudio (figura 6.58). Los valores indicados en línea de puntos en las gráficas de tasas ACR son los valores equitativos *max-min* efectivos.

Observamos que:

- Los valores de tasa ACR de las conexiones de los grupos A, B y C son también los equitativos *max-min* efectivos, aunque se observa oscilación en los valores correspondientes a los grupos A y B y una mayor oscilación en los valores correspondientes al grupo C, con respecto a los valores de la simulación VI.

- Los tamaños de las colas y la utilización del enlace bajo estudio se comportan de forma similar a los de la simulación VI.

Concluimos por tanto que el mecanismo de generación de señal de realimentación propuesto muestra un comportamiento equiparable con planificación WFQ y con planificación SCFQ en términos de equidad de asignación de tasas ACR.

6.4.2 Algoritmo de control de congestión básico

En la sección 5.3.4, presentamos el mecanismo final de generación de señal de realimentación, que resolvía algunos aspectos de estabilidad y de escalabilidad que aparecían en el mecanismo básico. Recordemos que el mecanismo básico no efectuaba ninguna retención de células. A continuación presentamos la evaluación del mecanismo de control de flujo soportado sobre el mecanismo básico de generación de señal de realimentación.

Se ha escogido el escenario de simulación siguiente: configuración GFC1 con distancias MAN, tal como el utilizado en la simulación VI. Como resultado de la simulación XII

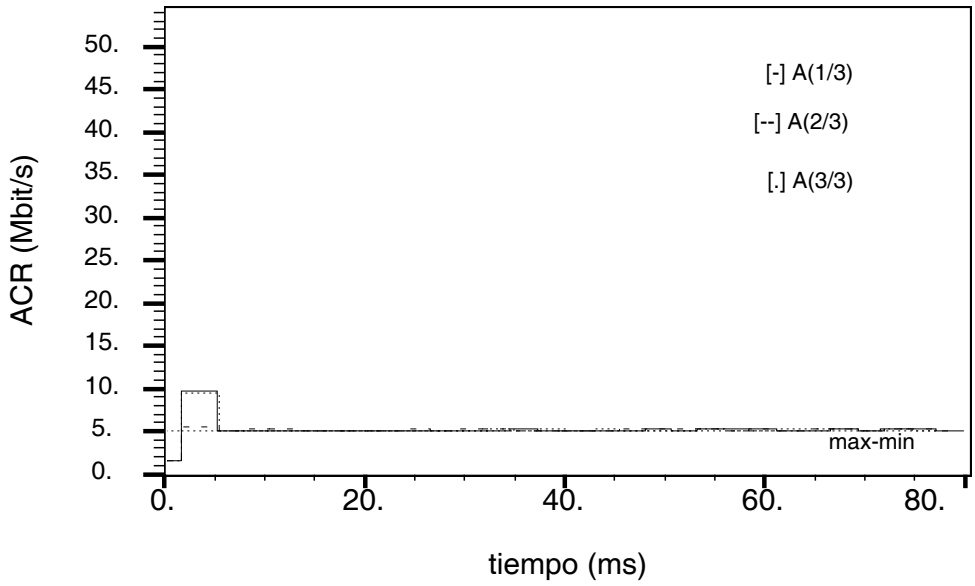

Figura 6.52. VALOR DE ACR PARA EL GRUPO DE CONEXIONES A EN SIMULACIÓN XI

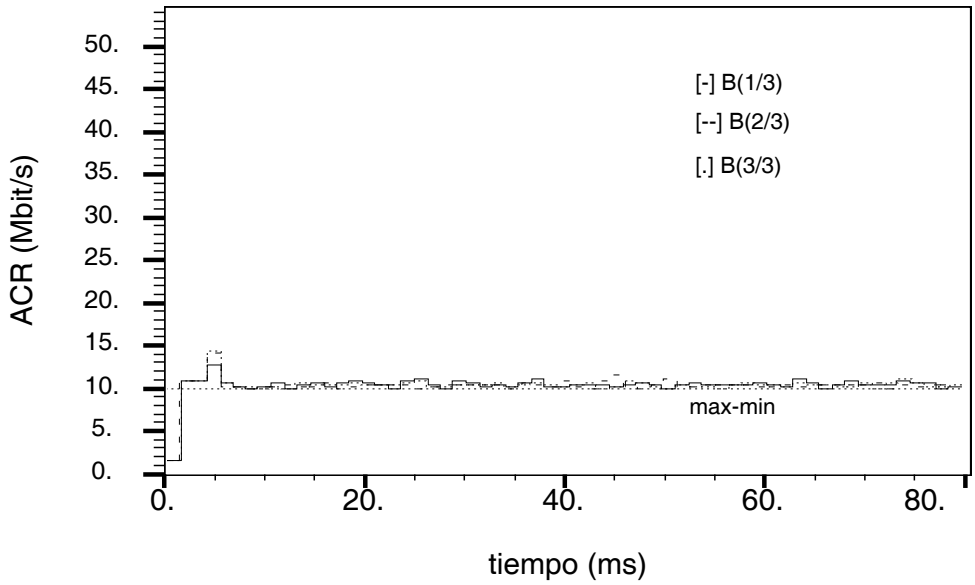

Figura 6.53. VALOR DE ACR PARA EL GRUPO DE CONEXIONES B EN SIMULACIÓN XI

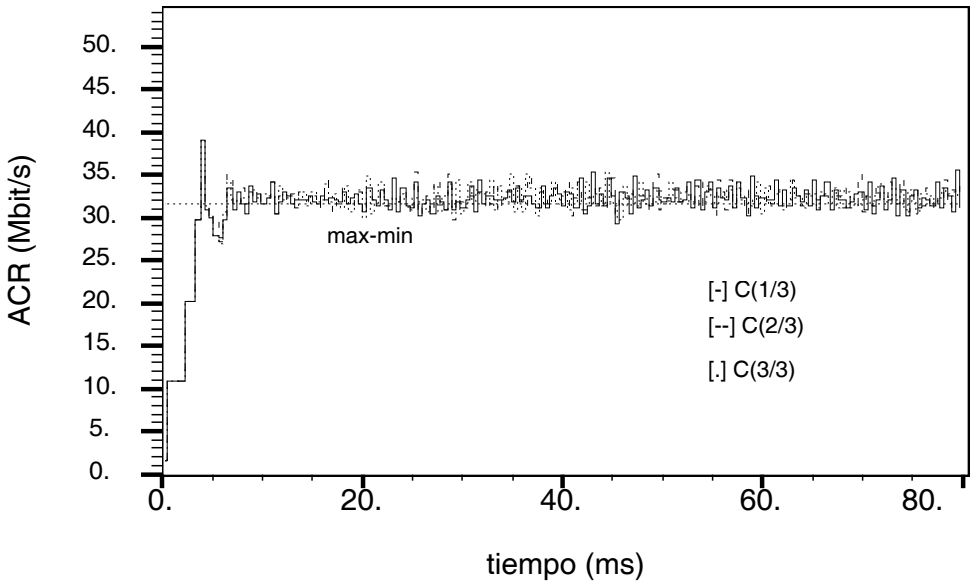

Figura 6.54. VALOR DE ACR PARA EL GRUPO DE CONEXIONES C EN SIMULACIÓN XI

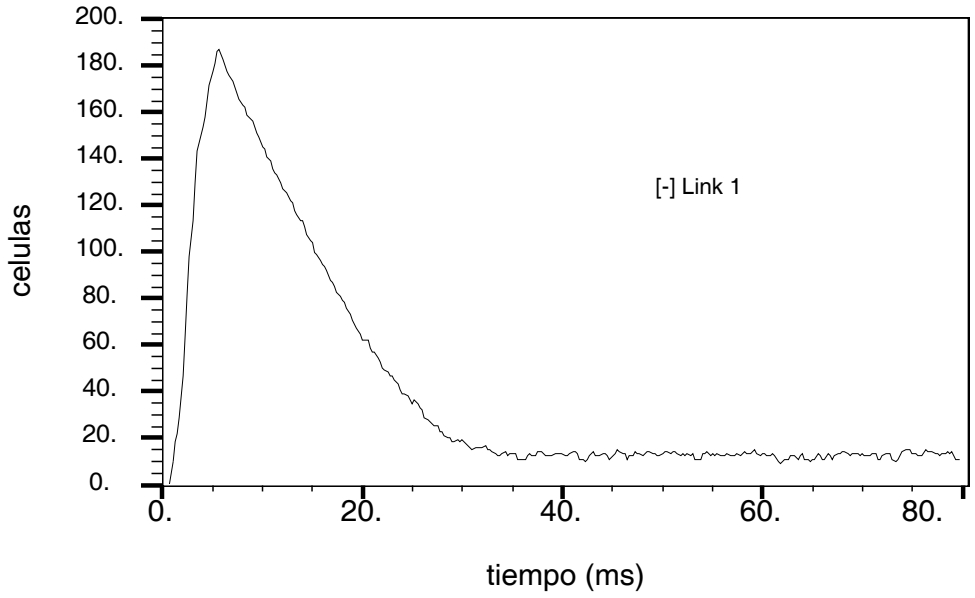

Figura 6.55. TAMAÑO DE COLA EN SW1(1) EN SIMULACIÓN XI

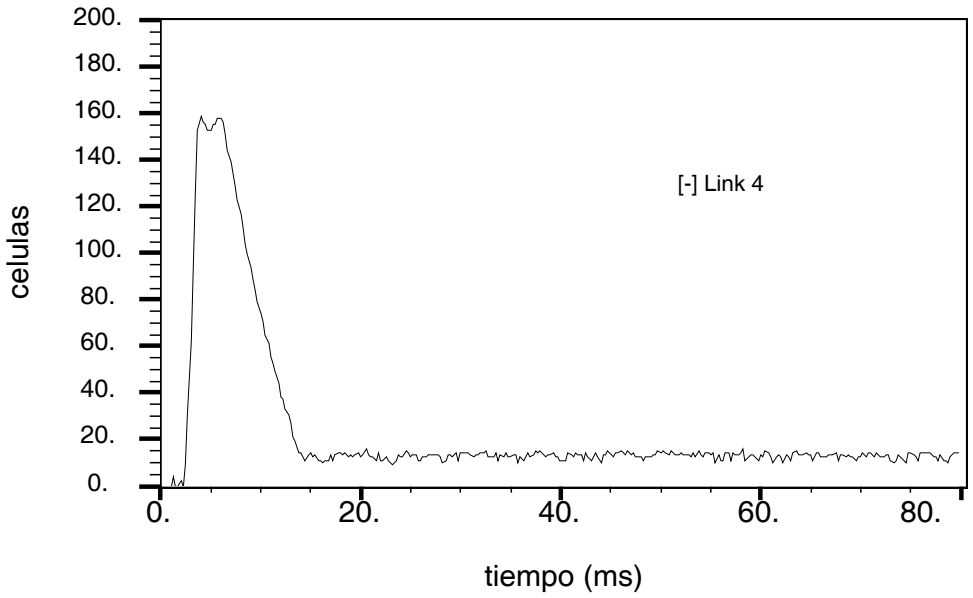

Figura 6.56. TAMAÑO DE COLA EN SW4(1) EN SIMULACIÓN XI

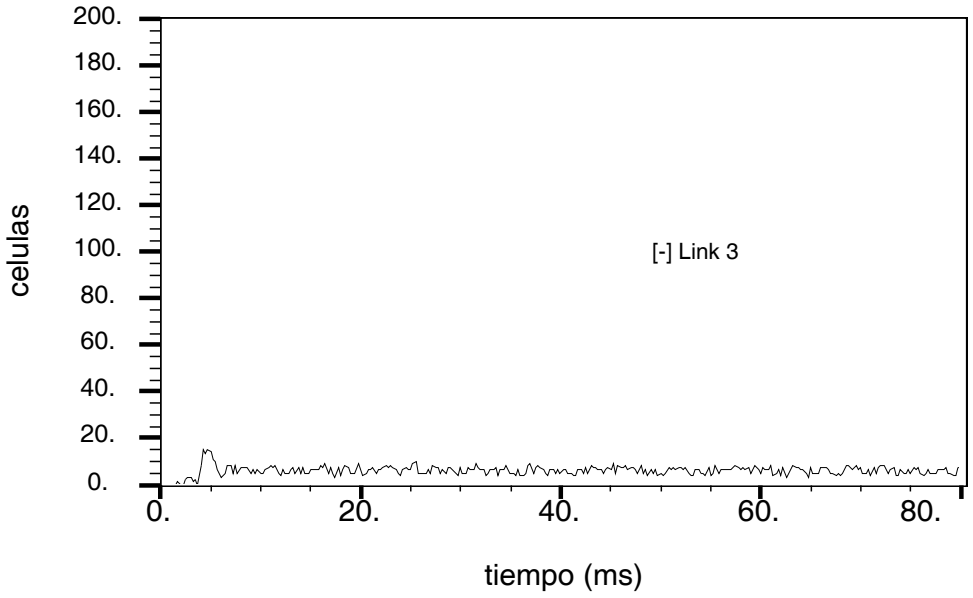

Figura 6.57. TAMAÑO DE COLA EN SW3(1) EN SIMULACIÓN XI

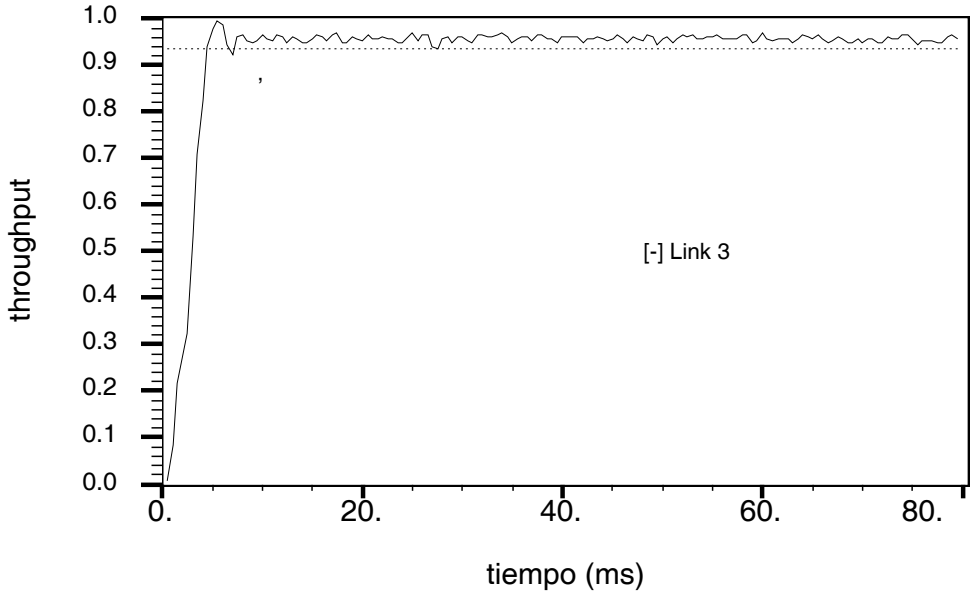

Figura 6.58. UTILIZACIÓN DEL ENLACE SW3(1) EN SIMULACIÓN XI

se muestran las tasas ACR de las conexiones del grupo A (figura 6.59), del grupo B (figura 6.60) y del grupo C (figura 6.61). Además, se muestra el nivel de llenado de las colas en el cuello de botella del grupo A (figura 6.62), del grupo B (figura 6.63) y del grupo C (figura 6.64). Finalmente, se presenta la utilización del enlace bajo estudio (figura 6.65). Los valores indicados en línea de puntos en las gráficas de tasas ACR son los valores equitativos *max-min* efectivos.

Observamos que:

- Los valores de tasa ACR de las conexiones de los grupos A, B y C oscilan significativamente, sobre todo los del grupo C, por encima del valor de tasa equitativa *max-min* efectivo.
- No se consigue mantener el nivel de llenado de las colas a un valor bajo; además, el nivel de llenado de las colas oscila en régimen estacionario.
- Siendo el nivel medio de llenado de las colas mayor que cero, al no existir retención de células ni descarte de *slot*, la utilización del enlace bajo estudio está consecuentemente próximo al 100%.

Las oscilaciones que se observan en el nivel de llenado de las colas se explican por los siguientes fenómenos:

1. Los intervalos de llenado de la cola en cada puerto vienen provocados por los aumentos de tasa ACR de las conexiones que se estrangulan en dicho puerto;

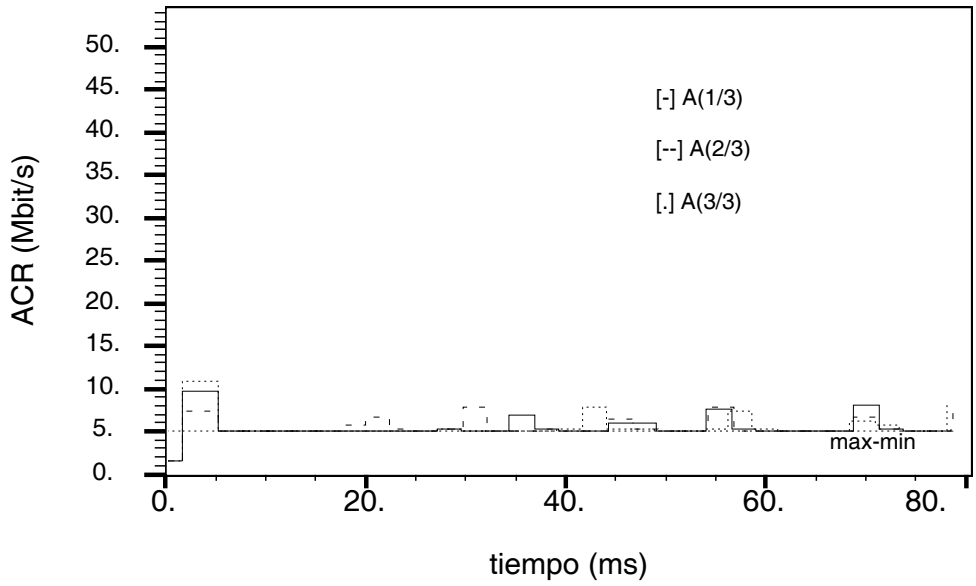

Figura 6.59. VALOR DE ACR PARA EL GRUPO DE CONEXIONES A EN SIMULACIÓN XII

2. el valor máximo de llenado de la cola se corresponde con el instante en que la tasa ACR de las conexiones que se estrangulan en el puerto se corrige a su valor equitativo *max-min* efectivo;

3. las colas se vacían con una pendiente igual al ancho de banda de drenaje;

4. la disminución del nivel de llenado de las colas, provoca que alguna de las colas de conexión se vacíe; este hecho es el causante, a su vez, del desajuste en la tasa ACR de las conexiones que se estrangulan en el puerto, con lo cual se vuelve a iniciar el ciclo de oscilación.

Además de la simulación XII, se ha llevado a cabo otra simulación del mecanismo básico de generación de señal de realimentación y se ha comprobado que la amplitud de las oscilaciones es mayor cuanto mayor es el número de conexiones establecidas en la red. Para ello se empleó una configuración de 2 nodos con inicio escalonado de las conexiones, tal como el de la simulación VIII.

Comparando los resultados de las simulaciones VIII (véase la sección 6.3.5) y XII queda justificada el mecanismo final propuesto en la sección 5.3.4.

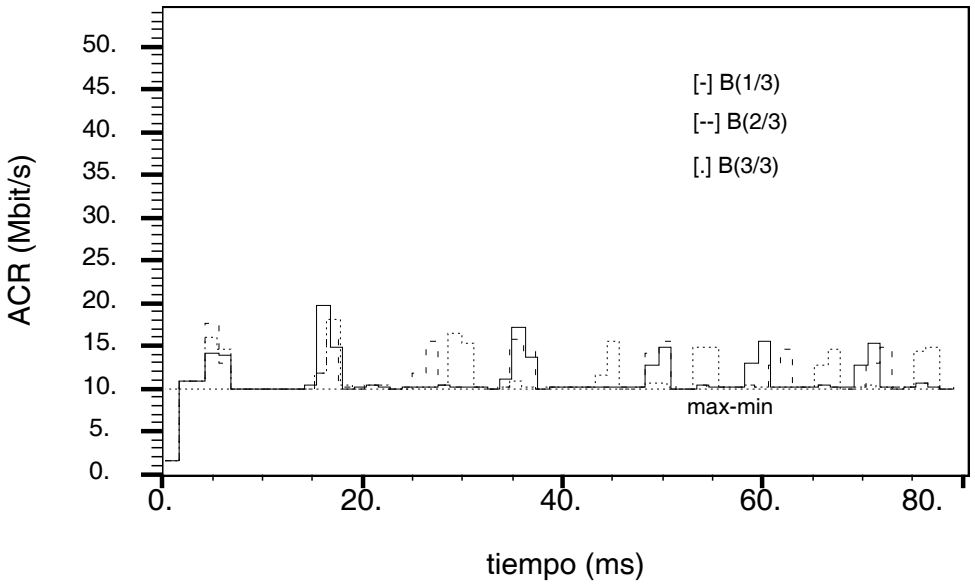

Figura 6.60. VALOR DE ACR PARA EL GRUPO DE CONEXIONES B EN SIMULACIÓN XII

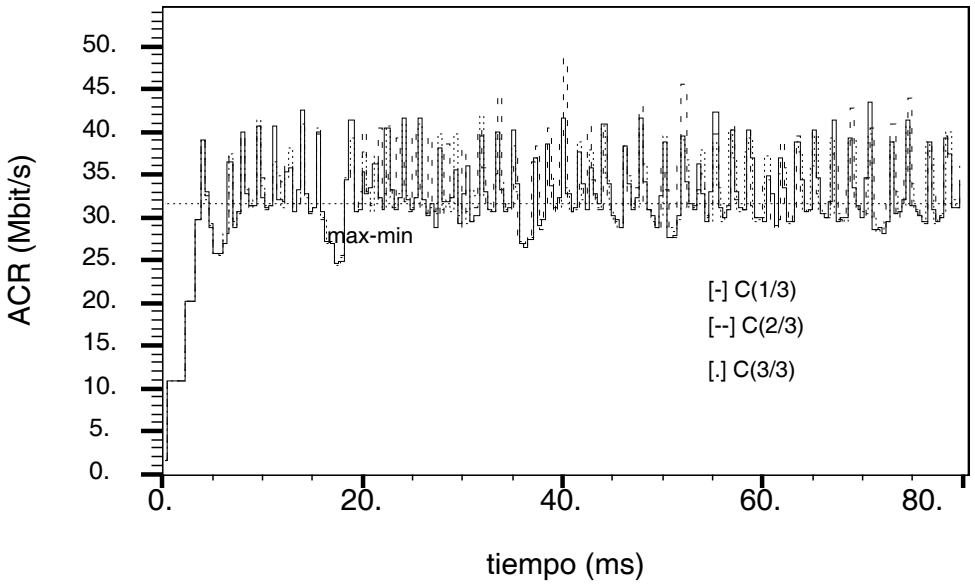

Figura 6.61. VALOR DE ACR PARA EL GRUPO DE CONEXIONES C EN SIMULACIÓN XII

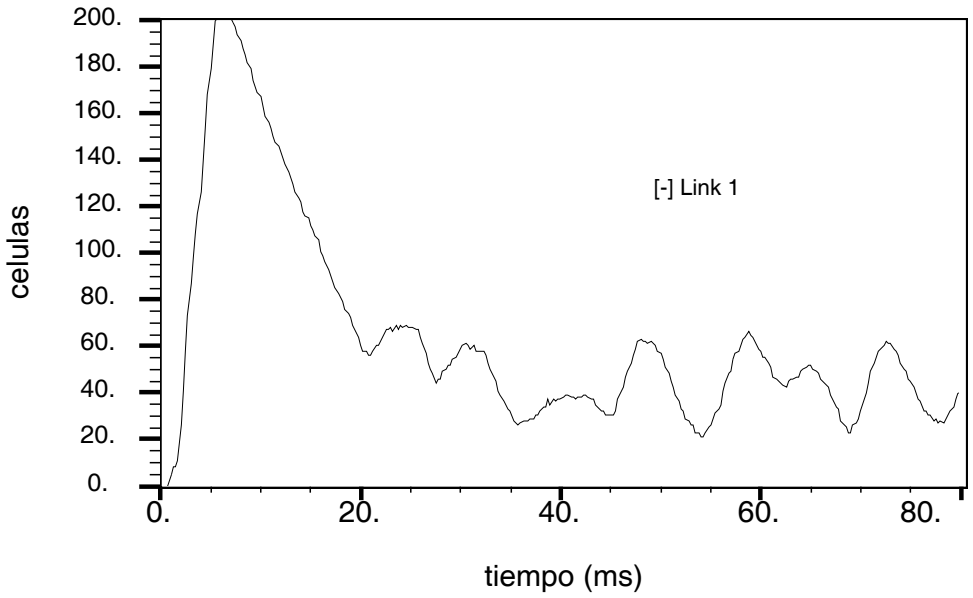

Figura 6.62. TAMAÑO DE COLA EN SW1(1) EN SIMULACIÓN XII

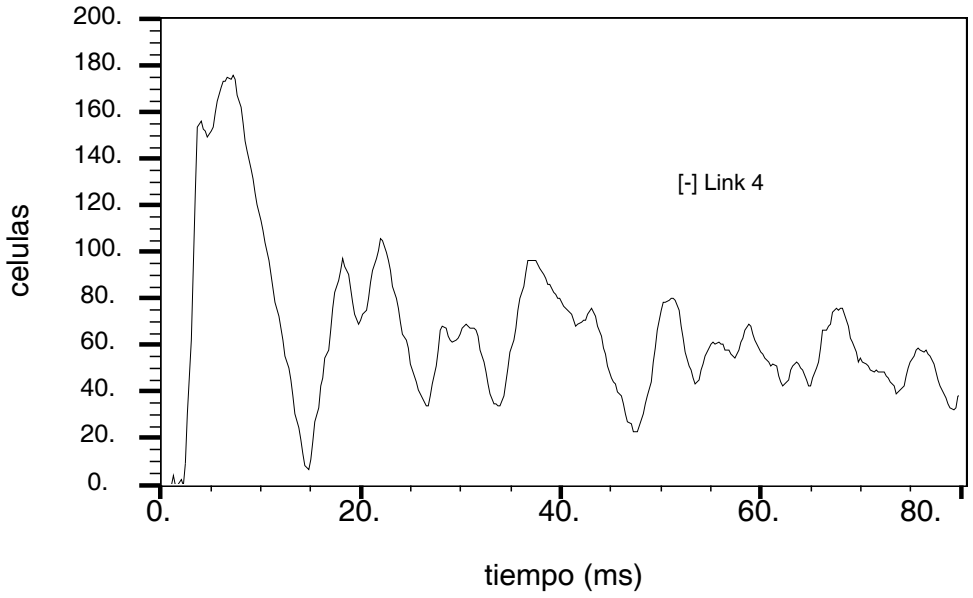

Figura 6.63. TAMAÑO DE COLA EN SW4(1) EN SIMULACIÓN XII

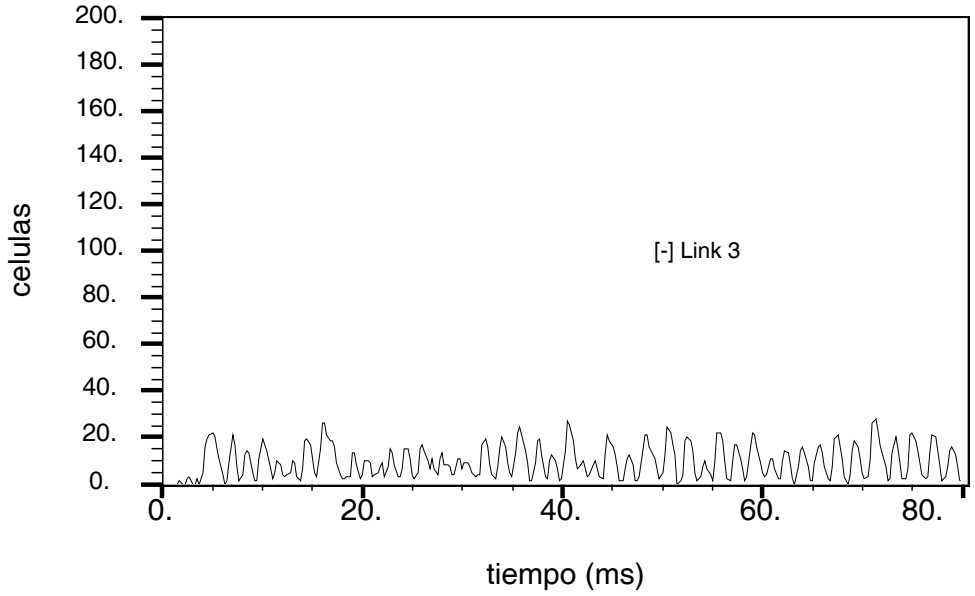

Figura 6.64. TAMAÑO DE COLA EN SW3(1) EN SIMULACIÓN XII

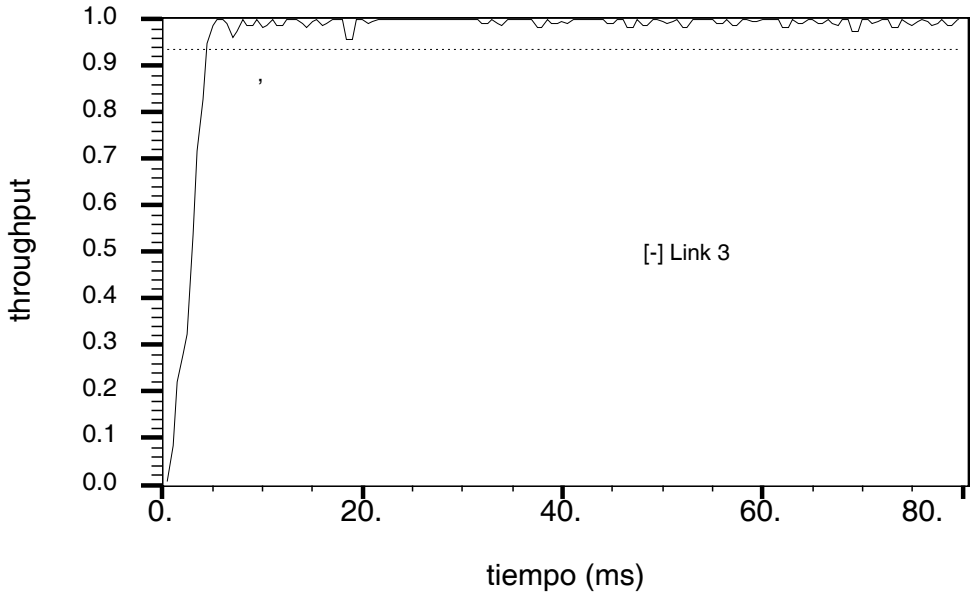

Figura 6.65. UTILIZACIÓN DEL ENLACE SW3(1) EN SIMULACIÓN XII

Conexión	*Peso*
A(1)	1
B(1)	1
C(1)	2
D(1)	3
E(1)	1

Tabla 6.12. PESOS RELATIVOS EN SIMULACIÓN XIII

Conexión	*0–20 ms*	*20–40 ms*	*40–60 ms*
A(1)	1/2 (0.45)	1/4 (0.23)	1/7 (0.13)
B(1)	1/2 (0.45)	1/4 (0.23)	1/7 (0.13)
C(1)	-	1/2 (0.45)	2/7 (0.26)
D(1)	-	-	3/7 (0.39)
E(1)	-	-	-
Conexión	*60–80 ms*	*80–100 ms*	*100–120 ms*
A(1)	1/8 (0.11)	-	-
B(1)	1/8 (0.11)	1/7 (0.13)	-
C(1)	1/4 (0.23)	2/7 (0.26)	1/3 (0.3)
D(1)	3/8 (0.34)	3/7 (0.39)	1/2 (0.45)
E(1)	1/8 (0.11)	1/7 (0.13)	1/6 (0.15)

Tabla 6.13. VALORES TEÓRICOS EQUITATIVOS EN EL SENTIDO *max-min* PONDERADO EN SIMULACIÓN XIII

6.4.3 Criterio de equidad *max-min* ponderado

Hemos escogido como escenario de test el mismo que el de la simulación VIII, esto es, una configuración de dos conmutadores con distancias MAN y con los instantes de inicio y de finalización de emisión dados en la tabla 6.10 de la página 171. Los pesos relativos asignados a cada conexión son los que se muestran en la tabla 6.12.

Los valores teóricos de tasa ACR equitativos en el sentido *max-min* ponderado en cada periodo se dan en la tabla 6.13. En paréntesis se muestra los valores teóricos efectivos.

Como resultado de la simulación XIII se muestran los valores de tasa ACR de todas las conexiones (figura 6.66), del nivel de llenado de las colas del puerto SW1(1) (figura 6.67) y de la utilización del enlace troncal (figura 6.68). Observamos que:

- Los valores obtenidos de tasa ACR se ajustan a los valores teóricos según la tabla 6.13.

- Los valores estacionarios de tamaño de cola se mantienen en el margen de 1 ó 2 células por conexión.

- La incorporación de una nueva conexión provoca un aumento de la utilización en el enlace, al igual que en la simulación VII. Cuando este aumento provoca que se alcance el valor de utilización 100%, el efecto de la incorporación se traslada a la cola, que aumenta, situación que no se dió en la simulación VII. Es posible también que la cola aumente sin que la utilización alcance el 100%, por efecto de un desajuste que afecte a una sola conexión, cuya cola aumentaría.
- La liberación de una conexión establecida provoca una reducción de utilización en el enlace, al igual que ocurría en la simulación VIII.

6.5 Conclusiones

En este capítulo se ha evaluado en qué grado la contribución principal de esta Tesis cumple las propiedades que idealmente debían poseer un esquema de control de flujo (véase la sección 4.1). Hemos comprobado que el esquema de control de flujo es eficiente, es equitativo *max-min*, es escalable, es estable y es resistente.

El algoritmo de conmutador propuesto en esta Tesis aporta dos novedades con respecto a los algoritmos existentes. En primer lugar, incorpora un algoritmo de planificación equitativa, a diferencia de los algoritmos de conmutador existentes, que incorporan algoritmos FCFS. En segundo lugar, la estimación de la tasa que realimentar a las fuentes tiene en cuenta únicamente medidas individualizadas de cada conexión, a diferencia de los algoritmos existentes, que utilizan total o parcialmente información agregada del conjunto de las conexiones —por ejemplo, el facto de carga en CAPC y en ERICA—.

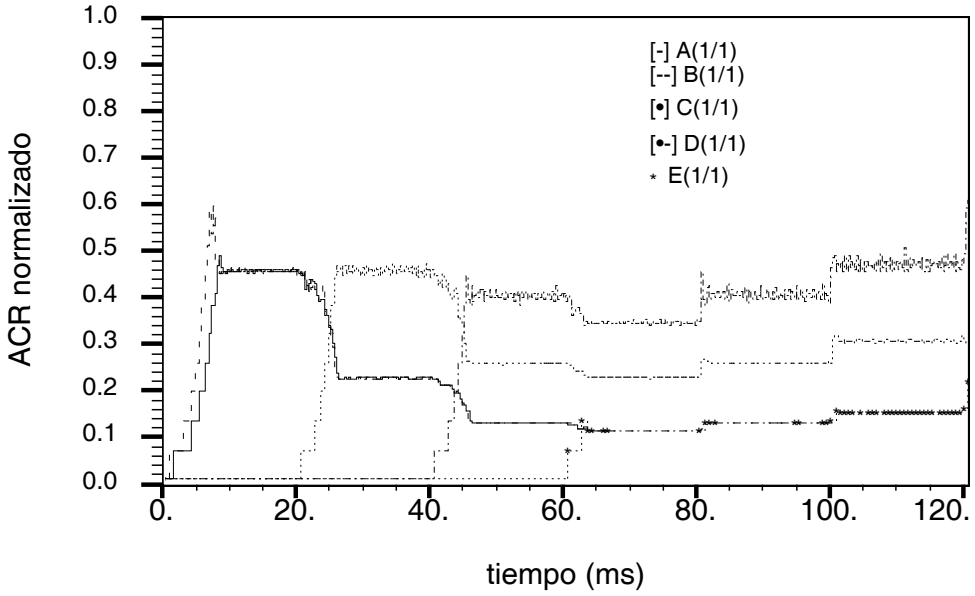

Figura 6.66. VALOR DE ACR EN SIMULACIÓN XIII

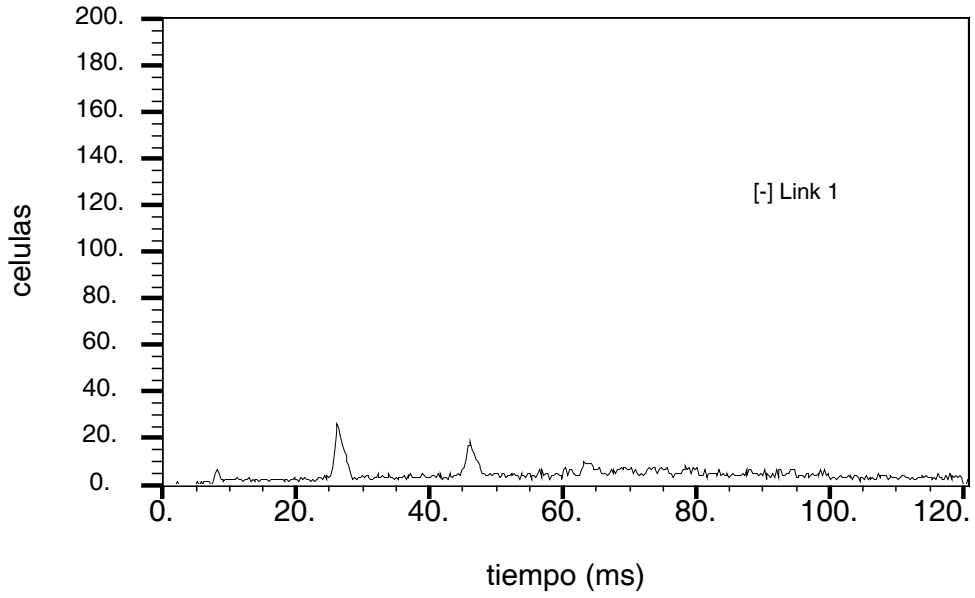

Figura 6.67. TAMAÑO DE COLA EN SW1(1) EN SIMULACIÓN XIII

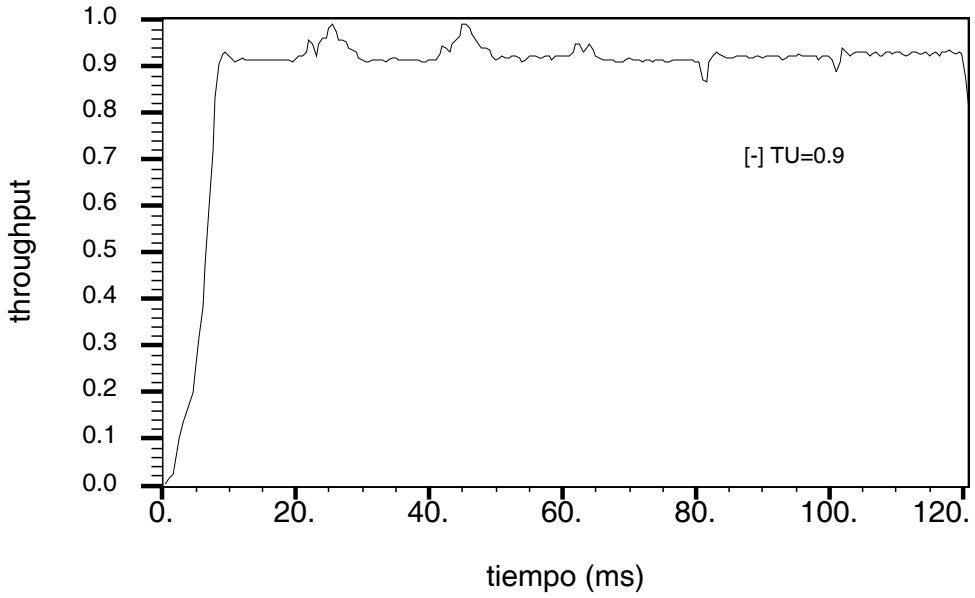

Figura 6.68. UTILIZACIÓN DEL ENLACE SW1(1) EN SIMULACIÓN XIII

Capítulo 7

Conclusiones y líneas de trabajo futuras

7.1 Conclusiones

La clase de servicio ABR provee un servicio tipo *best-effort* en el que la calidad de servicio que reciben los usuarios en cada momento les es comunicado mediante realimentación. Este control de flujo por realimentación genera el valor de tasa máxima de emisión permitida a cada una de las conexiones. De este modo, ABR establece garantías procedimentales en el servicio. No obstante, el servicio ABR debe, de acuerdo con la concepción de servicio *best-effort* desarrollada en esta Tesis y expuesta en el capítulo 2, garantizar la calidad de servicio que comunica a los usuarios, independientemente del comportamiento del resto de los usuarios, esto es, de si responden o no a la realimentación según el ajuste esperado. Esta garantía no puede ser soportada intrínsecamente por el control de flujo.

Los algoritmos de planificación son los responsables de asignar los recursos de forma que tal garantía sea efectiva. El algoritmo de planificación FCFS, que es el algoritmo de planificación que asumen todos los algoritmos de conmutador ABR existentes, no es capaz de garantizar asignaciones de recursos por sí mismo. En cambio, los algoritmos de planificación equitativa asignan el ancho de banda según un criterio equitativo en el sentido *max-min*. Hemos argumentado en esta Tesis la conveniencia de soportar la clase de servicio sobre técnicas de planificación equitativa. Asimismo, hemos demostrado que ABR puede efectivamente soportarse mediante técnicas de planificación equitativa. Para ello, hemos diseñado y evaluado un algoritmo de conmutador que estima la realimentación del control de flujo a partir de la asignación que lleva a cabo un algoritmo de planificación equitativa. En particular,

- Hemos demostrado que es factible estimar la tasa de realimentación en ABR mediante el cálculo del tiempo que permanece una célula en la cabeza de la cola de su conexión, cuando la planificación en el puerto del conmutador es de tipo *weighted fair queueing*.

- Hemos demostrado que el esquema de control de flujo presenta un funcionamiento

aceptable cuando el algoritmo de planificación es *self-clocked fair queueing*, el cual realiza cómputos menos complejos en la planificación de las células que el algoritmo *weighted fair queueing*.

- Se ha resuelto el control de la congestión en los conmutadores mediante la reserva de un ancho de banda de drenaje en cada uno de sus puertos. Este ancho de drenaje es igual a un porcentaje del ancho de banda ocupado por las conexiones estranguladas en el puerto. Ello se consigue multiplicando la estimación de tasa de realimentación por un coeficiente menor que la unidad antes de entregar la realimentación.

- Por último, se ha diseñado un algoritmo de retención de células que estabiliza el mecanismo de estimación de tasa de realimentación, sin disminuir la eficiencia del mecanismo original.

La asignación de tasas a las conexiones ABR debe ser equitativa. El criterio *max-min* asume que todas las conexiones tienen idénticas necesidades de ancho de banda. Hemos argumentado que la utilización de otros criterios de equidad permitiría que la clase de servicio ABR pueda dar soporte a otras aplicaciones, distintas de las aplicaciones transaccionales de datos, para las cuales estaba inicialmente destinada. Esas otras aplicaciones, como las de vídeo o las de audio, tienen requisitos de ancho de banda que son diferentes en función de cada aplicación. Además, algunas de ellas necesitan un valor mínimo de tasa permitida, que posibilite el mantenimiento de la interacción imprescindible para el funcionamiento de las aplicaciones. Estos dos requisitos pueden ser satisfechos por el algoritmo de conmutador propuesto en esta Tesis. Efectivamente, el algoritmo de conmutador permite que la asignación de tasas sea equitativa en un sentido *max-min* ponderado, de modo que los requisitos de ancho de banda pueden caracterizarse mediante un valor real positivo, que se emplearía como peso en la planificación. Además, incorporando un mecanismo de adaptación de los pesos empleados en la planificación, la asignación cumpliría el criterio de equidad denominado *de mínimos*, que tiene en cuenta la exigencia de un ancho de banda mínimo garantizado.

7.2 Líneas de trabajo futuras

Uno de los inconvenientes que presenta el algoritmo de conmutador propuesto en esta Tesis es la complejidad de implementación que añade al diseño de un conmutador. Dos son las fuentes de complejidad, a saber, el algoritmo de planificación propiamente dicho y el mecanismo de estimación de la tasa de realimentación. Consideramos interesante que se diseñen mecanismos que, manteniendo el principio de funcionamiento, disminuyan la complejidad de los algoritmos implicados. En cuanto al algoritmo de planificación, el algoritmo *weighted round-robin* es el algoritmo de planificación equitativa más ampliamente implantado en conmutadores ATM, por lo cual es el algoritmo que debería incorporar cualquier prototipo que implementase el algoritmo de conmutador propuesto. En cuanto al mecanismo de estimación de tasa, una contribución interesante sería encontrar una

alternativa a la necesidad de medir tiempos. Habría que considerar, por ejemplo, aproximar el tiempo de permanencia en cabeza de cola de conexión a partir del número de conexiones activas en el momento en que llega la célula al conmutador.

La propuesta realizada para soportar una clase de servicio ABR generalizada, esto es, con asignación *max-min* ponderada de tasas y con garantía de tasa mínima, debería ser verificada. Para ello, se debería escoger aplicaciones multimedia que pudiesen aprovechar la realimentación proporcionada por ABR y evaluar las prestaciones de un escenario de aplicaciones multimedia con distintos requisitos de ancho de banda soportadas sobre ABR generalizado. De hecho, existen propuestas de algoritmos de control de flujo por realimentación para el ajuste de la tasa de codificación de codecs de tasa variable (Kanakia y otros, 1993). La utilización de la realimentación que entrega el control de flujo en ABR para ajustar la tasa de generación de un flujo de vídeo o de audio sería un área de investigación prometedora, por cuanto que permitiría trasladar a las redes de banda ancha, el éxito que las aplicaciones multimedia están teniendo sobre la Internet.

El algoritmo de conmutador propuesto debería ser evaluado cuando soporta aplicaciones que utilizan el protocolo de transporte TCP, las cuales siguen siendo las más extendidas. De hecho, los algoritmos de conmutador más conocidos han sido profusamente evaluados en escenarios en los que las fuentes implementan el protocolo TCP.

Al respecto de la evaluación de TCP sobre ABR, hay que hacer notar que el protocolo TCP realiza su propio control de flujo. La consecuencia de ello es que el control de flujo en la capa de transporte interactúa con el control de flujo de capa ATM. El modelado de simulación de sistemas en los que existe más de una capa con mecanismos activos es complejo. Además, en cada capa existen diversas alternativas; por ejemplo, en la capa ATM existen distintas clases de servicio adecuadas para un mismo tipo de aplicación. Para abordar esta complejidad, proponemos (Guijarro y otros, 1998) una aproximación *multi-capa* al modelado. Esta aproximación se basa en la utilización del lenguaje SDL, normalizado por el UIT-T, para la descripción de los modelos. SDL permite una descripción modular de los sistemas, muy apropiada para los sistemas de comunicaciones. Asimismo, permite aplicar el paradigma de la orientación a objetos, muy útil para la reutilización de los modelos de entidades de capa de protocolo.

Si evaluar las prestaciones de ABR cuando soporta al protocolo TCP nos permite determinar la aplicabilidad de nuestro algoritmo de conmutador, no es menos cierto que el protocolo TCP también debería modificar su funcionamiento. La finalidad de ello sería aprovechar las nuevas características que ABR tiene como servicio *best-effort* en redes ATM: en ABR, la calidad de servicio que recibe el usuario le es comunicada de forma explícita. El protocolo TCP mejoraría sus prestaciones si, cuando se soportase sobre red ATM, desactivase el mecanismo de control de flujo por realimentación implícita e hiciese uso de la realimentación que obtiene la fuente ABR (Roberts, 1997).

En los párrafos anteriores, se ha constatado la siguiente contradicción. Por un lado, ABR abre la puerta al soporte de aplicaciones que, a diferencia de las aplicaciones transaccionales de datos que se soportan sobre TCP, puedan sacar provecho de la realimentación de control de flujo. Por otro lado, los algoritmos de conmutador en ABR son frecuentemente evaluados en escenarios de aplicaciones transaccionales de datos soportadas sobre

TCP, que ignoran cualquier comunicación explícita por parte de la red. En esta Tesis, hemos argumentado que el servicio ABR ha replanteado el escenario tradicional de provisión de los servicios *best-effort*. A diferencia de otras clases de servicio de capa ATM, tales como UBR, ABT o GFR, la clase de servicio ABR ofrece una comunicación explícita de la calidad de servicio que está recibiendo el usuario en cada momento, gracias al control de flujo por realimentación de capa ATM. Esta mejora no puede ser aprovechada por las aplicaciones tradicionales que se soportaban sobre servicios *best-effort*, tales como transferencia de ficheros o correo electrónico. Para estas aplicaciones es más apropiado un servicio *best-effort* con las características de UBR, ABT o GFR. En cambio, sí puede ser aprovechada por otras aplicaciones que no precisen del protocolo TCP para su transporte, tales como las aplicaciones multimedia. Para ello, no obstante, se necesita un criterio de distribución de ancho de banda más general que el criterio *max-min*. En esta Tesis, hemos demostrado que es factible soportar el servicio ABR utilizando el criterio *max-min* ponderado, el cual sí permite reconocer prioridades diferenciadas a cada uno de los usuarios de ABR.

Bibliografía

ALTMAN, E., F. BACCELLI y J.-C. BOLOT, «Discrete-time analysis of adaptive rate control mechanisms», en *Proceedings of 5th International Conference on Data Communications* (Raleigh, NC) (octubre 1993), 121–140.

ARULAMBALAM, A., X. CHEN y N. ANSARI, «Allocating fair rates for Available Bit Rate service in ATM networks», *IEEE Communications Magazine*, 34, núm. 11 (noviembre 1996), 92–100.

BARNHART, A., «Baseline performance using PRCA rate-control», Contribution 94-0597 (julio 1994a), The ATM Forum.

BARNHART, A. W., «Explicit rate performance evaluations», Contribution 94-0983 (octubre 1994b), The ATM Forum.

BENNETT, J. C. R., y H. ZHANG, «WF^2Q: Worst-case Fair Weighted Fair Queueing», en *Proceedings of Infocom'96* (San Francisco, California) (marzo 1996a), 120–128, IEEE.

BENNETT, J. C. R., y H. ZHANG, «Why WFQ is not good enough for integrated services networks», en *Proceedings of NOSSDAV'96* (abril 1996b).

BERNSTEIN, G. M., «Reserved bandwidth and reservationless traffic in rate allocating servers», en *Proceedings of Sigcomm'93* (julio 1993), 6–24, ACM.

BERTSEKAS, D., y R. GALLAGER, *Data Networks* (2 ed.)., Englewood Cliffs, New Jersey: Prentice-Hall (1992).

BOLOT, J. C., y A. U. SHANKAR, «Analysis of a fluid flow approximation to flow control dynamics», en *Proceedings of Infocom'92* (Florence, Italy) (mayo 1992), 2398–2407, IEEE.

BONOMI, F., y K. W. FENDICK, «The rate-based flow control framework for the Available Bit Rate ATM service», *IEEE Network Magazine*, 9, núm. 2 (marzo 1995), 25–39.

BOYER, P. E., y D. P. TRANCHIER, «A reservation principle with applications to the ATM traffic control», *Computer Networks and ISDN Systems*, 24 (1992), 321–334.

BRAKMO, L. S., y L. L. PETERSON, «TCP Vegas: end to end congestion avoidance on a Global Internet», *IEEE Journal on Selected Areas in Communications*, 13, núm. 8 (octubre 1995), 1465–1480.

CHAO, H. J., «A novel architecture for queue management in the ATM network», *IEEE Journal on Selected Areas in Communications*, 9, núm. 7 (septiembre 1991), 1110–1118.

CHAO, H. J., «A VLSI sequencer chip for ATM traffic shaper and queue manager», *IEEE Journal of Solid-State Circuits*, 27, núm. 11 (noviembre 1992), 1634–1643.

CHARNY, A., «An algorithm for rate allocation in a packet-switched network with feedback», Master's thesis, Massachusetts Institute of Technology, Massachusetts (mayo 1994).

CHARNY, A., D. D. CLARK y R. JAIN, «Congestion control with explicit rate indication», en *Proceedings of ICC'95* (Seattle) (1995), 1954–1963, IEEE.

CHEN, T. M., S. S. LIU y V. K. SAMALAM, «The Available Bit Rate service for data in ATM networks», *IEEE Communications Magazine*, 34, núm. 5 (mayo 1996), 56–71.

CHIU, D., y R. JAIN, «Analysis of the increase/decrease algorithms for congestion avoidance in computer networks», *Computer Networks and ISDN Systems*, 17, núm. 1 (junio 1989), 1–14.

CHIUSSI, F. M., J. G. KNEUER y V. P. KUMAR, «Low-cost scalable switching solutions for broadband networking: the ATLANTA architecture and chipset», *IEEE Communications Magazine*, 35, núm. 12 (diciembre 1997), 44–53.

CHIUSSI, F. M., Y. XIA y V. P. KUMAR, «Dynamic Max Rate Control Algorithm for Available Bit Rate service in ATM networks», en *Proceedings of Globecom'96* (London, UK) (noviembre 1996), IEEE.

CLARK, D. D., M. L. LAMBERT y L. ZHANG, «NETBLT: A bulk data transfer protocol», RFC 998 (marzo 1987), IETF.

CRUZ, R. L., «A calculus for network delay, part i: Network elements in isolation», *IEEE Transactions on Information Theory*, 37, núm. 1 (enero 1991a), 114–131.

CRUZ, R. L., «A calculus for network delay, part ii: Network analysis», *IEEE Transactions on Information Theory*, 37, núm. 1 (enero 1991b), 132–141.

DEMERS, A., S. KESHAV y S. SHENKER, «Analysis and simulation of a fair queueing algorithm», en *Proceedings of the Sigcomm'89* (Austin) (septiembre 1989), 1–12, ACM.

FERRARI, D., y D. C. VERMA, «A scheme for real-time channel establishment in wide-area networks», *IEEE Journal on Selected Areas in Communications*, 8, núm. 3 (abril 1990), 368–379.

FLOYD, S., y V. JACOBSON, «On traffic phase effect in packet-switched gateways», *Internetworking: Research and Experience*, 3, núm. 3 (septiembre 1992), 115–156.

FLOYD, S., y V. JACOBSON, «Random Early Detection gateways for congestion avoidance», *IEEE/ACM Transactions on Networking*, 1, núm. 4 (agosto 1993), 397–413.

GARRETT, M. W., «A service architecture for ATM: From applications to scheduling», *IEEE Network Magazine*, 10, núm. 3 (mayo 1996), 6–14.

GIACOPELLI, J.Ñ., J. J. HICKEY, W. S. MARCUS, W. D. SINCOSKIE y M. LITTLEWOOD, «Sunshine: a high-performance self-routing broadband packet switch architecture», *IEEE Journal on Selected Areas in Communications*, 9, núm. 8 (octubre 1991), 1289–1298.

GOLESTANI, S. J., «Congestion-free communication in high-speed packet networks», *IEEE Transactions on Communications*, 39, núm. 12 (diciembre 1991), 1802–1812.

GOLESTANI, S. J., «A Self-Clocked Fair Queueing scheme for broadband applications», en *Proceedings of Infocom'94* (Toronto, Canada) (junio 1994), 636–646, IEEE.

GOLESTANI, S. J., «Network delay analysis of a class of fair queueing algorithms», *IEEE Journal on Selected Areas in Communications*, 13, núm. 6 (agosto 1995), 1057–1070.

GREENBERG, A. G., y N. MADRAS, «How fair is fair queuing?», *Journal of the Association for Computing Machinery*, 39, núm. 3 (julio 1992), 568–598.

GUERIN, R., y J. HEINANEN, «UBR+ service category definition», Contribution 96-1598 (dec 1996), The ATM Forum.

GUIJARRO, L., V. PLA, J. R. VIDAL y J. MARTÍNEZ, «Multi-layer simulation approach for evaluation of data service support in ATM networks», en *Proceedings of the 1st International Conference on ATM (ICATM'98)* (Colmar, France) (junio 1998), IEEE.

HUGHES, D., «Fair share in the context of MCR», Contribution 94-0977 (octubre 1994), The ATM Forum.

JACOBSON, V., «Congestion avoidance and control», en *Proceedings of Sigcomm'88* (Stanford) (agosto 1988), 314–329, ACM.

JAIN, R., «A timeout-based congestion control scheme for window flow-controlled networks», *IEEE Journal on Selected Areas in Communications*, 4, núm. 7 (octubre 1986), 1162–1167.

JAIN, R., S. KALYANARAMAN, S. FAHMY, R. GOYAL y S.-C. KIM, «Source behavior for ATM ABR traffic management: an explanation», *IEEE Communications Magazine*, 34, núm. 11 (noviembre 1996), 50–57.

JAIN, R., S. KALYANARAMAN, R. VISWANATHAN y R. GOYAL, «A sample switch algorithm», Contribution 95-0178 (febrero 1995), The ATM Forum.

KALMANEK, C. R., H. KANAKIA y S. KESHAV, «Rate controlled servers for very high-speed networks», en *Proceedings of Globecom'90* (San Diego) (diciembre 1990), 300.3.1–300.3.9, IEEE.

KALMANEK, C. R., S. KESHAV, W. T. MARSHALL, S. P. MORGAN y R. C. RESTRICK, «Xunet 2: Lessons from an early wide-area ATM testbed», *IEEE/ACM Transactions on Networking*, 5, núm. 1 (febrero 1997), 40–55.

KALYANARAMAN, S., *Traffic management for the Available Bit Rate (ABR) service in ATM networks*, Tesis Doctoral, Graduate School of The Ohio State University, The Ohio State University (1997).

KANAKIA, H., P. MISHRA y A. REIBMAN, «An adaptive congestion control scheme for real-time packet video transport», en *Proceedings of Sigcomm'93* (San Francisco) (septiembre 1993), ACM.

KATEVENIS, M., S. SIDIROPOULOS y C. COURCOUBETIS, «Weighted Round-Robin cell multiplexing in a general-purpose ATM switch chip», *IEEE Journal on Selected Areas in Communications*, 9, núm. 8 (octubre 1991), 1265–1279.

KESHAV, S., *Congestion control in computer networks*, Tesis Doctoral, UC Berkeley (septiembre 1991a), TR-654.

KESHAV, S., «A control-theoretic approach to flow control», en *Proceedings of Sigcomm'91* (Zurich) (septiembre 1991b), 3–15, ACM.

KESHAV, S., «On the efficient implementation of Fair Queueing», *Journal of Internetworking: Research and Experience*, 2, núm. 3 (septiembre 1991c), 157–173.

KESHAV, S., *An engineering approach to computer networking. ATM networks, the Internet, and the Telephone network* (1 ed.)., Professional computing series, Reading, Massachusetts: Addison-Wesley (abril 1997).

KLEINROCK, L., *Queueing systems. Computer applications*, Volume 2, New York: John Wiley & Sons (1976).

LEFELHOCZ, C., B. LYLES, S. SHENKER y L. ZHANG, «Congestion control for best-effort service: why we need a new paradigm», *IEEE Network Magazine*, 10, núm. 1 (enero 1996), 10–19.

LEMIEUX, C., «Theory of flow control in shared networks and its application in the Canadian telephone network», *IEEE Transactions on Communications*, 29, núm. 4 (abril 1981), 299–413.

LYLES, B., «ABR control loops, network stability, and the need for explicit calculation/indication of cells stranded in the network», Contribution 94-1198 (Kyoto, Japan) (noviembre 1994a), The ATM Forum.

LYLES, B., «Existence proof of ABR mechanism», Contribution (noviembre 1994b), ITU-T, USA.

LYLES, B., «Simulations of ABR mechanism», Contribution (noviembre 1994c), ITU-T, Xerox Corporation.

LYLES, B., y A. LIN, «A Class-Y mechanism and preliminary simulations», Technical Report T1S1.5/94-207 (julio 1994a), T1.

LYLES, B., y A. LIN, «Definition and preliminary simulation of a rate-based congestion control mechanism with explicit feedback of bottleneck rates», Contribution 94-0708 (Irvine, CA) (julio 1994b), The ATM Forum.

MARTÍNEZ, J., *Provisión de servicios de datos sin conexión en la RDSI-BA*, Tesis Doctoral, Departamento de Comunicaciones, Universidad Politécnica de Valencia (marzo 1997).

MARTÍNEZ, J., M. LLOP y J. M. GALINDO, «Support of non real-time networked multimedia systems in ATM based networks», en *Proceedings of the 3rd International Workshop on Protocols for Multimedia Systems (PROMS'96)* (Madrid) (octubre 1996), 269–284.

MARTÍNEZ, J., J. R. VIDAL y L. GUIJARRO, «A low complexity congestion control algorithm for the ABR class of service», en *Proceedings of IDMS'98* (Oslo, Norway) (septiembre 1998), submitted for acceptance.

MASCOLO, S., D. CAVENDISH y M. GERLA, «ATM rate based congestion control using a Smith predictor: an EPRCA implementation», en *Proceedings of Infocom'96* (San Francisco) (marzo 1996), 569–576, IEEE.

MISHRA, P. P., H. KANAKIA y S. K. TRIPATHI, «On hop-by-hop rate-based congestion control», *IEEE/ACM Transactions on Networking*, 4, núm. 2 (abril 1996), 224–239.

NAGLE, J., «On packet switches with infinite storage», RFC 970 (diciembre 1985), IETF Network Working Group.

NEWMAN, P., «Traffic management for ATM local arera networks», *IEEE Communications Magazine*, 32, núm. 8 (agosto 1994), 44–50.

PAREKH, A. K. J., *A Generalized Processor Sharing Approach for Flow Control in Integrated Services Networks*, Tesis Doctoral, Laboratory for Information and Decision Systems. Massachusetts Institute of Technology, Cambridge, Massachusetts (febrero 1992).

PAREKH, A. K. J., «A generalized processor sharing approach for flow control in integrated services networks: The single-node case», *IEEE/ACM Transactions on Networking*, 1, núm. 3 (junio 1993), 344–357.

PAREKH, A. K. J., «A generalized processor sharing approach for flow control in integrated services networks: The multiple node case», *IEEE/ACM Transactions on Networking*, 2, núm. 2 (abril 1994), 137–150.

POSTEL, J., «Transmission control protocol», RFC 793 (Virginia) (septiembre 1981), DARPA Information Processing Techniques Office.

RAMAKRISHNAN, K. K., D. CHIU y R. JAIN, «Congestion avoidance in computer networks with a connectionless network layer — Part IV: a selective binary feedback scheme for general topologies», Technical Report DEC-TR-510 (1987), DEC.

RAMAKRISHNAN, K. K., y R. JAIN, «A binary feedback scheme for congestion avoidance in computer networks», *ACM Transactions on Computer Systems*, 8, núm. 2 (mayo 1990), 155–181.

REXFORD, J. L., A. G. GREENBERG y F. G. BONOMI, «Hardware-efficient fair architectures for high-speed networks», en *Proceedings of Infocom'96* (San Francisco, California) (marzo 1996), 638–646, IEEE.

RITTER, M., «Analysis of feedback-oriented congestion control mechanisms for ABR services», en *Proceedings of the 10th ITC specialists seminar* (Lund) (septiembre 1996), 291–308.

RITTER, M., «The effect of bottleneck service rate variations on the performance of the ABR flow control», en *Proceedings of Infocom'97* (Kobe, Japan) (abril 1997), IEEE.

ROBERTS, J. W., «Virtual spacing for flexible traffic control», *International Journal of Communications Systems*, 7 (octubre 1994a), 307–318.

ROBERTS, J. W., B. BENSAOU y Y. CANETTI, «A traffic control framework for high speed data transmission», en H. PERROS, G. PUJOLLE, y Y. TAKAHASHI(Eds.), *Proceedings of the IFIP TC6 Task Group/WG 6.4 International Workshop on Performance Communications Systems* (Martinique) (enero 1993). North Holland.

ROBERTS, L., «Enhanced PRCA», Contribution 94-0735 (agosto 1994b), The ATM Forum.

ROBERTS, L. G., «Flow control - explicit rate vs. TCP», Technical report (julio 1997), Cell-In-Frames Alliance, http://www.ziplink.net/ lroberts.

ROHRS, C. E., R. A. BERRY y S. J. O'HALEK, «A control engineer's look at ATM congestion avoidance», en *Proceedings of Globecom'95* (Singapore) (noviembre 1995), 1089–1094, IEEE.

ROMANOW, A., y S. FLOYD, «Dynamics of TCP traffic over ATM networks», *IEEE Journal on Selected Areas in Communications*, 13, núm. 4 (mayo 1995), 633–641.

SCHWARTZ, M., *Redes de Telecomunicaciones. Protocolos, modelado y análisis* (1 ed.)., Madrid: Addison-Wesley Iberoamericana (1994).

SHENKER, S., «A theoretical analysis of feedback flow control», en *Proceedings of Sigcomm'90* (septiembre 1990), 156–165, ACM.

SHREEDHAR, M., y G. VARGHESE, «Efficient fair queueing using Deficit Round Robin», en *Proceedings of the Sigcomm'95* (Boston, Massachusetts) (septiembre 1995), 231–243, ACM.

SIMCOE, R. J., «Test configurations for fairness and other tests», Contribution 94-0557 (Irvine, California) (julio 1994), The ATM Forum.

SIU, K.-Y., y H.-Y. TZENG, «Intelligent control for ABR service in ATM networks», *ACM Computer Communications Review*, 24, núm. 5 (octubre 1994), 81–106.

STALLINGS, W., *ISDN and Broadband ISDN with Frame Relay and ATM* (3 ed.)., UK: Prentice-Hall International (1995).

STILIADIS, D., y A. VARMA, «Latency-Rate servers: A general model for analysis of traffic scheduling algorithms», en *Proceedings of Infocom'96* (San Francisco) (abril 1996), 111–119, IEEE.

TANENBAUM, A. S., *Computer networks* (2 ed.)., London, UK: Prentice-Hall International (1989).

THE ATM FORUM COMMITTEE, «Traffic Management specification, version 4.0», Technical Report af-tm-0056.000 (abril 1996), The ATM Forum.

TURNER, C., y L. PETERSON, «Image Transfer: an end-to-end design», en *Proceedings of Sigcomm'92* (agosto 1992), ACM.

UIT-T, «Aspectos de servicio de la RDSI-BA», Recomendación I.211 (Ginebra) (marzo 1993a), UIT-T.

UIT-T, «Características del Modo de Transferencia Asíncrona de la RDSI-BA», Recomendación I.150 (Ginebra) (marzo 1993b), UIT-T.

UIT-T, «Traffic control and congestion control in B-ISDN», Recommendation (Temporary Document 35-E) I.371 (Ginebra) (mayo 1996), UIT-T.

VERMA, D. C., H. ZHANG y D. FERRARI, «Delay jitter control for real-time communication in a packet-switching networks», en *Proceedings of Tricomm'91* (Chapel Hill, NC) (abril 1991).

VIDAL, J. R., *Evaluación de prestaciones mediante técnicas de descripción formal de la emulación de red local sobre ATM*, Tesis Doctoral, Departamento de Comunicaciones, Universidad Politécnica de Valencia (julio 1997).

YIN, N., y M. G. HLUCHYJ, «On closed-loop rate control for ATM cell relay networks», en *Proceedings of Infocom'94* (Toronto) (junio 1994), 99–108, IEEE.

ZHANG, L., «Virtualclock: A new traffic control algorithm for packet-switched networks», *ACM Transactions on Computer Systems*, 9, núm. 2 (mayo 1991), 101–104.