

1  
2ej. 01170

# Evaluación del desempeño de técnicas de control de tráfico para el servicio ABR en redes ATM

Miguel López Guerrero

Maestría en Ingeniería Eléctrica

259993

1998

TESIS CON  
FALLA DE ORIGEN



Universidad Nacional  
Autónoma de México



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

*A la memoria de un hombre ejemplar:  
Ricardo Guerrero Montaña*

# Reconocimientos

Estoy feliz de tener esta oportunidad para poder expresar mi gratitud al Dr. Luis Orozco Barbosa, profesor del Departamento de Ingeniería Eléctrica en la Universidad de Ottawa, por haber dirigido este trabajo y por todos los detalles de atención que me brindó durante mi estancia en su laboratorio. Reconozco también que este trabajo definitivamente no hubiera sido posible sin la codirección, el gran apoyo y la motivación del Dr. Jesús Savage Carmona, profesor del programa de Posgrado en Ingeniería Eléctrica de la UNAM.

Asimismo, agradezco las aportaciones y el compañerismo del M. en C. José R. Gallardo, alumno de doctorado en la Universidad de Washington. También expreso mi sincero aprecio, por su invaluable ayuda en la validación de los resultados, al Ing. Steve Rosenberg, el experto en análisis del desempeño de sistemas ATM de *Newbridge Networks*.

Estoy también en deuda con el Fis. Pedro Ángel Cuenca Castillo, estudiante de doctorado en la Universidad de Castilla-la Mancha, quien me proporcionó algunas secuencias de vídeo MPEG y útiles comentarios durante el desarrollo del modelo de simulación.

Este trabajo también se enriqueció de las valiosas sugerencias de los siguientes profesores de la División de Estudios de Posgrado de la Facultad de Ingeniería: Dr. Rogelio Alcántara Silva, M. en C. Humberto López Gallegos, Dr. Carlos Rivera Rivera y M. en C. Raúl Santoyo Delgado.

Finalmente reconozco el esfuerzo de la Fundación UNAM al haber financiado parte de mi estancia en la Universidad de Ottawa.

Miguel López Guerrero  
Marzo 30, 1998

# Introducción

El modo de transferencia asíncrono (ATM) se perfila como la tecnología de conmutación dominante en el mercado de las telecomunicaciones de las próximas décadas. Ésta es capaz de proporcionar nuevos y mejores servicios con diversos requerimientos de calidad y en un ambiente integrado.

La aceptación inicial de esta propuesta tecnológica depende, en gran medida, de su capacidad para reemplazar a otras opciones para el transporte eficiente del tráfico que generan los programas de cómputo actuales. Para ello el *ATM Forum* creó el servicio con tasa disponible (ABR). Éste es adecuado para soportar el tráfico de aplicaciones que se caracterizan por su tolerancia al retardo y que pueden operar con un ancho de banda variante con el tiempo.

La evolución de la especificación de ABR en el *ATM Forum* ha reflejado el amplio rango de intereses de sus miembros. Durante la definición del mecanismo de control para proporcionar el servicio, se desarrollaron dos aproximaciones antagónicas: el control basado en la regulación de la tasa de la fuente y el control basado en créditos.

En 1994 el *ATM Forum* seleccionó el método de control basado en tasa para proporcionar el servicio ABR. Sin embargo, la preocupación por minimizar el riesgo de estandarizar soluciones inadecuadas ha ocasionado que no se detenga el esfuerzo de investigación en soluciones alternativas.

Los estudios realizados sobre el comportamiento de los mecanismos de control usualmente han empleado modelos de tráfico fácilmente analizables. Éstos son poco representativos de la realidad ya que hallazgos muy recientes indican que el tráfico de datos ocurre por ráfagas sobre diferentes escalas de tiempo. Esto ocasiona grandes y repentinos cambios de la carga de la red; situación no prevista por los modelos tradicionalmente utilizados.

Así, el objetivo fundamental de este trabajo es investigar el desempeño de las técnicas de control para el servicio ABR utilizando modelos de tráfico realistas.

# Capítulo 1

## Tecnología ATM

Tradicionalmente se ha utilizado una tecnología particular de telecomunicaciones para el transporte de un tipo de dato específico. Es así que actualmente contamos con redes de conmutación de circuitos para el tráfico de voz y de conmutación de paquetes para la transmisión de datos [Schwartz 1994].

En la década de los 1980's comenzó el desarrollo de tecnologías de telecomunicaciones capaces de ofrecer, en una red común, nuevos y mejores servicios de transmisión de voz, vídeo y datos. El resultado de este esfuerzo dio lugar al desarrollo de estándares sobre las redes digitales de banda ancha con servicios integrados o B-ISDN (de *Broadband Integrated Services Digital Network*). De éstas, las redes de modo de transferencia asíncrono<sup>1</sup> o ATM (de *Asynchronous Transfer Mode*) son parte integrante [Schwartz 1996].

---

<sup>1</sup> En este caso, el significado de las siglas ATM no se relaciona directamente con el concepto tradicional de una comunicación asíncrona [Campbell 1994]. La palabra "asíncrono" se utiliza para indicar que las unidades de agrupamiento de datos de una misma conexión pueden exhibir un patrón irregular de ocurrencia en el medio de transmisión.

## 1.1 Descripción general de la tecnología de conmutación ATM

La tecnología ATM se utiliza para realizar las funciones de conmutación y multicanalización de un sistema de comunicaciones. El modelo de referencia de protocolos se muestra en la figura 1.1.

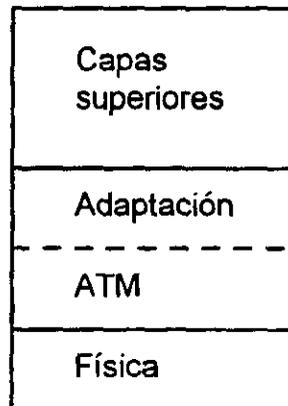


Figura 1.1 Modelo de referencia ATM

En la capa ATM los datos se agrupan en pequeñas unidades<sup>1</sup> de longitud fija<sup>2</sup> llamadas celdas. Uno de los beneficios resultantes es una reducción del tiempo de procesamiento por cada unidad, lo que ocasiona tiempos de conmutación muy cortos [Stallings 1995]. Cada una de las celdas contiene 53 bytes, de los cuales, los primeros 5 (cabecera) contienen datos de control utilizados por la red y los restantes 48 son de información.

Los protocolos de las capas superiores (como TCP, NetWare, AppleTalk, etc.) utilizan paquetes de longitud variable. Éstos son casi siempre más grandes que el campo de información de la celda ATM, por lo que para establecer el intercambio de datos, se utiliza una capa de adaptación. Ésta segmenta los

<sup>1</sup> La selección del tamaño de la celda estuvo fuertemente influenciada por el interés de lograr una integración satisfactoria de la actual infraestructura telefónica con la tecnología ATM. Dos aspectos dominan la dinámica de la interacción: a) la red telefónica tiene la característica de regresar una fracción de la señal a la fuente; b) el retardo en las conversaciones de voz varía con el tamaño de la celda. Tomando en cuenta la combinación estos dos factores, el retorno de la señal de voz debe mantenerse a menos de 10ms para evitar un eco perceptiblemente molesto al oído humano. Así, se seleccionó un tamaño pequeño de la celda para evitar este problema [Stevenson 1993].

<sup>2</sup> Existe una ventaja implícita al utilizar un tamaño fijo de las celdas. Cuando un nodo de conmutación ATM se analiza como un sistema M/D/1, se puede demostrar que el tiempo medio de espera se minimiza cuando la varianza en el tiempo de servicio es cero.

paquetes provenientes de los protocolos de alto nivel para su transferencia a través de la red. Los segmentos se colocan en celdas; se les añade, en la cabecera, la información que identifica la conexión virtual; y se transmiten a la red.

Cuando la celda llega a un nodo, el conmutador ATM utiliza los datos de la cabecera y el número del enlace por el que llegó para acceder una tabla. Como resultado de esta consulta, se obtiene el número de la interfaz de salida por la que la celda debe enviarse.

Cuando la celda finalmente llega a su destino, la capa de adaptación invierte el proceso de segmentación para formar el paquete original de datos a partir de las celdas colectadas. Una vez que éste se ensambla, se examina y puede ser descartado si se le encuentra algún error. Cuando esto ocurre, el protocolo de alto nivel detecta la pérdida e inicia procedimientos de recuperación, tales como retransmisiones.

Para proporcionar flexibilidad y escalabilidad, ATM se diseñó para operar en forma independiente del medio físico y con anchos de banda escalables.

## **1.2 Servicios de la red ATM**

La red ATM proporciona un servicio orientado a conexión. Un equipo terminal conectado a la red requiere el establecimiento de una conexión con otro antes de que pueda transmitir datos. Todas las conexiones son virtuales en el sentido de que la capacidad de un enlace no se les asigna permanentemente; pero la red les proporciona el ancho de banda requerido cuando sea necesario.

Las conexiones virtuales se identifican en términos de circuitos virtuales (VC de *virtual circuit*) y trayectorias virtuales (VP de *virtual path*). Los VC pueden iniciar y terminar entre dos puntos arbitrarios de la red y se pueden agrupar en un VP si comparten la misma ruta de transmisión (vea la figura 1.2).

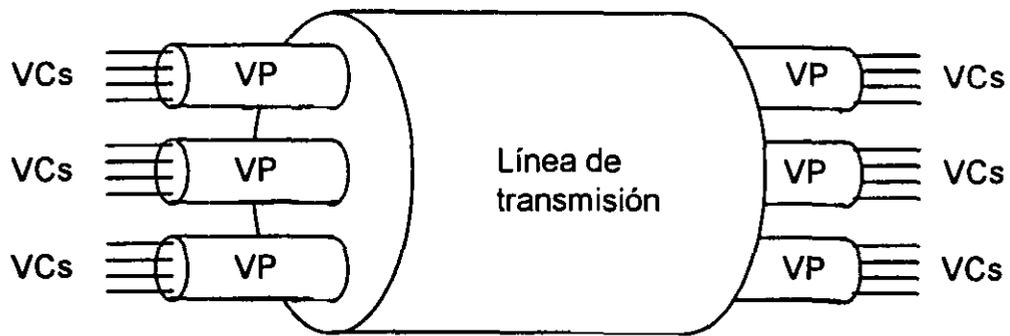


Figura 1.2 Canales virtuales y trayectorias virtuales

Sobre estas conexiones virtuales, diferentes tipos de aplicaciones requieren niveles específicos de servicio de la red. Por ejemplo, aplicaciones de voz y vídeo son muy sensibles al retardo y sus variaciones de magnitud; pero no se afectan por una pérdida pequeña de celdas. Por otro lado, las aplicaciones de datos no dependen de retardos o sus variaciones; pero son críticamente sensibles a la pérdida de celdas.

Para cumplir con los requerimientos específicos de cada aplicación, el nodo que inicia el proceso de establecimiento de la conexión debe informar a la red las características de servicio deseadas. La información proporcionada puede incluir:

- Número identificador del nodo al que se llama
- Requerimiento promedio de ancho de banda
- Requerimiento máximo de ancho de banda
- Máximo porcentaje aceptable de pérdida de celdas
- Máxima variación aceptable en el retardo de transferencia de las celdas

La red utiliza estos valores para seleccionar los enlaces que formarán la conexión virtual. Si no se puede proporcionar el servicio con la calidad deseada, se rechaza el requerimiento de conexión.

Las clases de servicios genéricos que la red ATM debe proporcionar se han establecido de acuerdo a los siguientes tres parámetros básicos:

- Tasa de transmisión (constante o variable )
- Sincronía entre la fuente y el destino (requerida o no requerida)
- Modo de conexión ("orientada a conexión" o "no orientada a conexión")

La tabla 1.1 muestra las cuatro principales clases de servicios definidos por el ITU-TSS (anteriormente CCITT) y otros organismos de estandarización.

Tabla 1.1 Clases de servicios en B-ISDN (tomado de [Schwartz 1996])

| Clase | Tipo de tasa | Sincronía | Modo de conexión |
|-------|--------------|-----------|------------------|
| A     | constante    | r         | oc               |
| B     | variable     | r         | oc               |
| C     | variable     | nr        | oc               |
| D     | variable     | nr        | noc              |

r=requerida, nr=no requerida, oc=orientada a conexión, noc=no orientada a conexión

El servicio clase A genera un flujo ininterrumpido de *bits* a una tasa constante de transmisión. Proporciona un ancho de banda garantizado con un retardo medio pequeño y poco variable. El porcentaje de pérdida de celdas queda limitado por una pequeña cota superior. Algunos ejemplos de aplicaciones que utilizan el servicio clase A son voz codificada en PCM, vídeo a una tasa constante y emulación de circuitos.

El servicio clase B también genera un flujo ininterrumpido; pero requiere un canal de tasa variable entre los valores máximo y mínimo especificados. Tiene un límite máximo en el retardo y su variación, así como en el porcentaje de pérdida de celdas. Entre los ejemplos de sus aplicaciones se incluyen transmisiones a tasa variable de voz y vídeo.

El servicio clase C es apropiado para el transporte de tráfico de datos en aplicaciones orientadas a conexión. Como ejemplos de éstas podemos citar servicios como X.25, Frame Relay y señalización en canal D de ISDN. El servicio clase D se define también para el tráfico de datos; pero para aplicaciones no orientadas a conexión, tales como la transferencia de paquetes en redes LAN y SMDS.

Para proporcionar los servicios descritos se han definido varias capas de adaptación o AAL (de *ATM adaptation layer*). Éstas se encargan fundamentalmente de realizar las funciones de segmentación y ensamble

descritas anteriormente. Adicionalmente, pueden realizar procesos específicos de la clase de servicio. Para separar claramente esas dos funciones, la capa de adaptación se segmenta en dos subcapas: una de convergencia y otra de segmentación y ensamble como se muestra esquemáticamente en la figura 1.3.

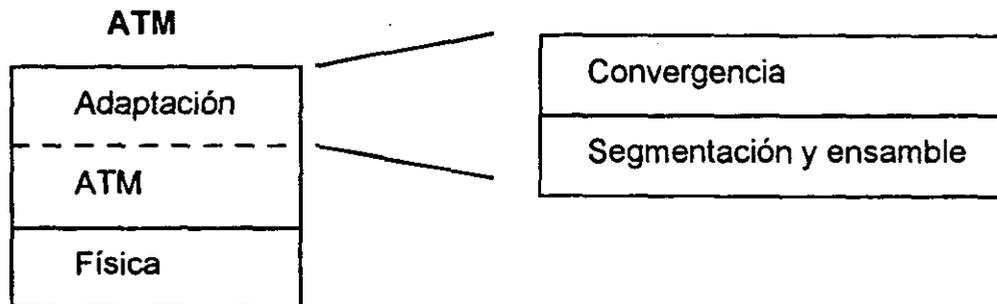


Figura 1.3 Subcapas de la capa de adaptación ATM (AAL)

Actualmente hay cuatro protocolos AAL definidos por el ITU-T:

- AAL 1 Permite la transmisión a una tasa constante y mantiene la sincronía entre los extremos de la conexión. Se diseñó para soportar el tráfico clase A.
- AAL 2 Coadyuva a la transmisión de datos a una tasa variable manteniendo la sincronía entre la fuente y el destino. El protocolo es adecuado para el tráfico de clase B.
- AAL 3/4 Soporta la comunicación en los modos "orientado a conexión" y "no orientado a conexión", a tasa variable y sin mantener la sincronía entre la fuente y el destino. Originalmente había dos protocolos AAL (3 y 4), los cuales se combinaron en uno sólo diseñado para el transporte del tráfico de las clases C y D.
- AAL 5 Proporciona una capa de adaptación simple y eficiente que soporta comunicaciones orientadas a conexión, a tasa variable y sin mantener una relación de sincronía entre la fuente y el destino. Originalmente esta capa de adaptación no era parte del conjunto original de características de ATM; pero ha logrado una mayor aceptación que AAL 3/4 para soportar el tráfico de clase C. El protocolo tiene una funcionalidad reducida; pero es más fácil de implantar y provee una mejor detección de error.

Al nivel de la capa ATM, el *ATM Forum* ha recomendado las siguientes clases de servicio para satisfacer los requerimientos de los diversos tipos de tráfico.

- Servicio con tasa constante (*continuous bit rate* o CBR). Garantiza valores máximos en el retardo y su variación. Asimismo, el porcentaje de pérdida de celdas se mantiene dentro de un límite. Este servicio es adecuado para transmisiones de voz.
- Servicio con tasa variable (*variable bit rate* o VBR). Puede proporcionarse en "tiempo real" y en "tiempo no real" referidos como RT-VBR (*real-time VBR*) y NRT-VBR (*nonreal-time VBR*) respectivamente. Ambos tienen diferentes requerimientos de servicio.

El RT-VBR puede utilizarse para transmisiones en tiempo real de voz o vídeo comprimidos. El tipo de tráfico generado tiene requerimientos de servicio similares a los de CBR: restricciones estrictas en el retardo, en la variabilidad del retardo y en la probabilidad de pérdidas. En contraste, el NRT-VBR es apropiado para aplicaciones que generan tráfico por ráfagas y que son tolerantes a variaciones de retardo. Este servicio también garantiza un valor máximo del retardo de transferencia y pequeños porcentajes de pérdida de celdas. La intensidad de las ráfagas y de la tasa de transmisión de la fuente se regulan por técnicas de control de tráfico.

- Servicio con tasa no especificada (*unspecified bit rate* o UBR). Constituye el soporte de las aplicaciones actuales que transmiten datos a base de ráfagas espaciadas y que tienen pocos requerimientos. Este servicio es adecuado para el transporte del tráfico generado por la multicanalización estadística de un gran número de fuentes.
- Servicio con tasa disponible (*available bit rate* o ABR). Éste es adecuado para aquellas aplicaciones que requieren un porcentaje bajo de pérdida de celdas y que pueden operar con intervalos de retardo y rendimiento. Las fuentes de tráfico deben ser capaces de adaptarse a las condiciones existentes en la red, aunque se les garantiza una tasa mínima de transmisión de celdas.

Se espera que el servicio ABR se utilice ampliamente para aplicaciones tales como transferencia de datos, envío de mensajes, distribución-recuperación de información y sesiones remotas [Saito *et al.* 1996].

## Capítulo 2

# Técnicas de control de tráfico para el servicio ABR

En el verano de 1993 comenzó la definición de un servicio para asignar dinámicamente el ancho de banda a conexiones en una red ATM. Antes de que finalizara el otoño de 1994, el *ATM Forum* introdujo el servicio *available bit rate* (ABR) para este propósito. La aproximación de control seleccionada se basó en la regulación de la tasa de la fuente tomando como base el intercambio de información de control entre los extremos de la conexión. La especificación actual del servicio se publicó por el *ATM Forum* en la primavera de 1996 [ATM Forum 1996]. Aquélla conservó básicamente las características ya disponibles desde 1994; pero incorporó algunas modificaciones en favor de una arquitectura más flexible [Fendick 1996].

En este capítulo se presentan las técnicas de control de mayor trascendencia en la evolución del servicio ABR y se describe en detalle la especificación actual. Asimismo, se presenta la aproximación de control basada en créditos; alternativa tradicionalmente considerada antagónica y que durante el último año ha despertado un interés creciente entre la comunidad de investigadores interesados en ATM.

## 2.1 El servicio ABR

En el proceso de establecimiento de una conexión en una red ATM, la entidad usuaria puede especificar los límites permisibles en aquellos parámetros que afectan su desempeño, principalmente retardo y porcentaje de pérdida de celdas. Con los valores de estos parámetros, la red puede estimar confiablemente la cantidad de recursos requeridos para realizar el servicio y aceptar la conexión si cuenta con ellos. Así, la fuente adquiere el compromiso de ajustarse al servicio originalmente solicitado y la red lo debe proporcionar con la calidad requerida. En esta forma se crea tácitamente un *contrato de tráfico*.

Durante la duración de la conexión la red no controla la fuente (aproximación considerada como de *malla abierta*), aunque monitorea su comportamiento para verificar que cumpla con el contrato establecido. Esta técnica de control es adecuada para aplicaciones de tiempo real (voz y vídeo), cuyos requerimientos se pueden estimar fácilmente.

Por otro lado, los requerimientos de las aplicaciones que no son de tiempo real son muy difíciles de estimar. Tradicionalmente la transmisión de este tipo de datos se ha efectuado con la política de *realizar el mejor esfuerzo*. Esto es, no se establece ninguna garantía en la calidad del servicio de transferencia de datos y éstos pueden llegar a su destino aleatoriamente retardados o descartarse en algún punto intermedio de su trayectoria. Esta política de atención es adecuada para aplicaciones que pueden añadir procedimientos en los extremos para garantizar la integridad de la información, tales como el correo electrónico y las transferencias de archivos.

Para las aplicaciones anteriormente descritas, se encuentra en proceso de definición el servicio *available bit rate* o ABR, basado en un control de *malla cerrada*. La naturaleza del servicio difiere de VBR y CBR en varios aspectos. Las conexiones ABR comparten el ancho de banda que resta instantáneamente en el canal después de enviar aquellos datos correspondientes a las conexiones de CBR y VBR (ver figura 2.1). Así, al tráfico ABR se le asigna el ancho de banda que de otra forma quedaría sin usar, incrementándose la utilización del enlace sin afectar la calidad del servicio de las conexiones con prioridad.

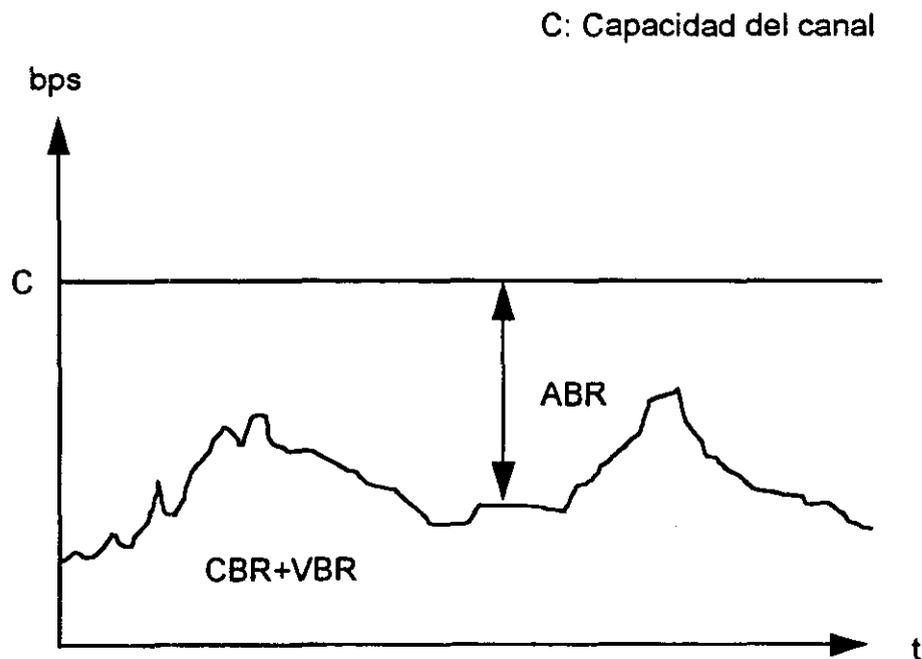


Figura 2.1 Servicio ABR

El ancho de banda disponible para cada conexión ABR es dinámico debido a la aleatoriedad del tráfico CBR/VBR y puede disminuir hasta una tasa mínima previamente especificada.

Ocasionalmente ocurren algunos períodos transitorios de ajuste en la tasa de transmisión de las fuentes debido al retardo en la obtención de la realimentación de la red. En consecuencia, es posible que se generen colas considerables en los *buffers* de los conmutadores durante aquellos momentos en los que el tráfico ABR excede el ancho de banda disponible.

Debido a estos retardos por encolamiento, la red no puede garantizar valores precisos en los retardos de transferencia extremo a extremo o en la varianza del retardo de las celdas. En consecuencia, el servicio ABR es apropiado sólo para aquellas aplicaciones que pueden adaptar sus tasas de transmisión a un ancho de banda variable con el tiempo y tolerar retardos impredecibles.

El servicio en la red dependerá de qué tan bien las fuentes pueden ajustar sus tasas al ancho de banda disponible. Como una motivación para que los usuarios

controlen sus tasas apropiadamente, a aquéllos que puedan adaptarse, se les garantiza un pequeño porcentaje de pérdida de celdas. Como se puede ver, ABR no es exactamente un servicio con la política de *realizar el mejor esfuerzo*. La red y el usuario tienen un acuerdo de cooperación implícito para controlar la dinámica del servicio.

## 2.2 La función de la red

Para determinar la tasa apropiada de transmisión, la fuente debe obtener información acerca de los recursos disponibles en la red. Ésta a su vez puede reportar su estado utilizando alguno de los siguientes métodos:

- No proporcionar información. En este caso los usuarios tienen que inferir indirectamente el estado de la red. Por ejemplo, una celda perdida podría interpretarse como la presencia de congestión.
- Notificación binaria de congestión. Con este método la red comunica la presencia o ausencia de congestión empleando una variable binaria. Ésta se define en los estándares con el nombre de *explicit forward congestion indication* (EFCI) y consiste en un *bit* del campo *payload type* (PT) de la cabecera de una celda ATM.

En esta aproximación, la fuente genera las celdas con la variable EFCI inicializada en 0 y cualquier nodo congestionado en la trayectoria la puede cambiar a 1. El extremo receptor monitorea el estado de esta indicación binaria de congestión en las celdas recibidas y regresa un mensaje de control a la fuente.

- Notificación detallada de congestión. Para implantar esta técnica los nodos de la red deben tener la capacidad de registrar y proporcionar información acerca del estado local de congestión. Con estos datos se puede lograr una descripción detallada del estado de la red. El reporte se puede enviar en el campo de carga útil de una celda de administración de recursos o RM (del inglés *resource management*).
- Ancho de banda explícito. En este caso la información acerca del ancho de banda disponible para cada conexión viaja en una celda de RM originada en la fuente. Este valor se ajusta sucesivamente por algunos nodos de la trayectoria dependiendo de las condiciones locales de congestión. Una vez que esta celda llega al destino, se envía de regreso hacia la fuente con el valor finalmente determinado.

### **2.3 La función de los usuarios**

Con la información del estado de la red, obtenida por alguno de los métodos anteriormente descritos, los usuarios deben ajustar su tasa de transmisión.

En un escenario típico, el destinatario recibe las celdas, realiza algún procesamiento en ellas y regresa un mensaje de realimentación a la fuente. La acción de control originada por este proceso puede clasificarse en una de los siguientes tipos:

- Realimentación negativa. La fuente incrementa su tasa por defecto (en la ausencia de mensajes de realimentación) y la reduce solamente cuando el destino le indica la presencia de congestión.
- Realimentación positiva. La fuente decrementa su tasa por defecto y la incrementa sólo cuando el destino anuncia la ausencia de congestión.
- Realimentación explícita. La fuente mantiene su tasa por defecto hasta que se le indica cambiarla a otro valor específico.

Con realimentación negativa o positiva las fuentes convergen gradualmente a la tasa apropiada, esto en algunos casos puede originar comportamientos oscilatorios. En contraste, con realimentación explícita la fuente se puede ajustar rápidamente.

### **2.4 Técnicas de control**

El proceso de definición de la técnica de control apropiada para el servicio ABR evolucionó inicialmente en dos direcciones opuestas: control basado en tasa efectuado en los puntos extremos de la conexión y control basado en créditos entre cada par de nodos de la red. A continuación se describen ambas técnicas.

#### **2.4.1 Control basado en créditos**

En esta aproximación, un nodo no envía celdas de datos al siguiente elemento de la ruta hasta que éste le anuncia, con mensajes de control, la disponibilidad de espacio para almacenarlas.

Hay dos fases en este proceso [Kung *et al.* 1995]. En la primera, al circuito virtual se le asigna un tamaño de *buffer* en el receptor. En la segunda, el transmisor mantiene un registro constantemente actualizado de la cantidad de memoria disponible en el receptor para asegurarse que no suceda ningún desbordamiento. De esta forma, no ocurre pérdida de celdas, lo cual ha constituido la motivación más importante para la inversión en el desarrollo de esta técnica.

El mecanismo de control puede ser estático o adaptable [Kung *et al.* 1995]. En el primero se utiliza un tamaño fijo del *buffer*, asignado al circuito virtual, durante el tiempo de vida de la conexión. El control adaptable se propone con la finalidad de lograr un desempeño óptimo al ajustar el tamaño del *buffer* tomando en consideración aquellos factores que cambian dinámicamente o que no se conocen *a priori*.

Las ventajas obtenidas se realizan con el costo de tener colas separadas para cada conexión y una sobrecarga asociada al tráfico de control. El control por conexión requiere un incremento en costo y complejidad de los conmutadores.

## **2.4.2 Control basado en tasa**

Ya que cada celda ATM contiene el mismo número de *bits*, el control del ancho de banda de una conexión, se logra directamente regulando la tasa de transmisión de celdas de la fuente [Bonomi *et al.* 1995]. Esta aproximación no garantiza la ausencia de pérdida de celdas aunque resulta ser sencilla y flexible. A continuación se describen las variantes más importantes que el mecanismo de control ha tenido en su proceso evolutivo.

### **2.4.2.1 EFCI**

El estado de la red se comunica, entre la fuente y el destinatario, a través del método de notificación binaria de congestión. Dependiendo de las condiciones locales de la carga de tráfico, la bandera binaria EFCI de las celdas puede ser modificada en algún punto del recorrido desde la fuente hacia el destinatario.

Cuando en el extremo receptor se detectan celdas con la indicación de congestión (EFCI=1), se envía periódicamente un mensaje de realimentación hacia la fuente (celdas de RM).

Ante la recepción de celdas de RM, la fuente decrementa su tasa en una cantidad proporcional al valor instantáneo de su ventana de control. Si por el contrario no se reciben celdas de realimentación, incrementa su ventana por una cantidad fija.

En la práctica, este mecanismo tiene algunas desventajas. Las celdas de RM se envían hacia la fuente sólo cuando hay congestión; es decir, cuando los retardos son largos y es más probable la ocurrencia de pérdidas. En consecuencia, es muy probable que las fuentes continúen incrementando sus tasas y agravando la situación. Para reducir el tiempo de respuesta se ha propuesto que en condiciones severas de congestión los conmutadores generen celdas de RM y las envíen a la fuente (*backward explicit congestion notification* o BECN).

Otro problema es la equidad con los usuarios. Ésta se garantiza sólo si todas las conexiones tienen los mismos valores en sus parámetros de operación [Chen et al. 1996].

Este esquema también incluyó una opción para cerrar la malla de realimentación en algún punto intermedio y de esta forma crear dominios más cortos.

#### 2.4.2.2 PRCA

El algoritmo *proportional rate control algorithm* (PRCA) parte de las ideas del EFCI; pero introduce algunos cambios fundamentales. A diferencia del EFCI (en el que todas las celdas se creaban con la indicación de congestión inicializada en 0), en el algoritmo PRCA la fuente transmite una de cada  $N_m$  celdas con la bandera EFCI en 0 y las restantes en 1.

Cada vez que el destinatario recibe una celda con EFCI en 0, envía una celda de RM hacia la fuente como se muestra en la figura 2.2. De esta forma, las celdas de RM se generan en forma aproximadamente proporcional a la tasa de transmisión de la fuente y el tráfico ocasionado por ellas se mantiene limitado a una fracción fija del total.

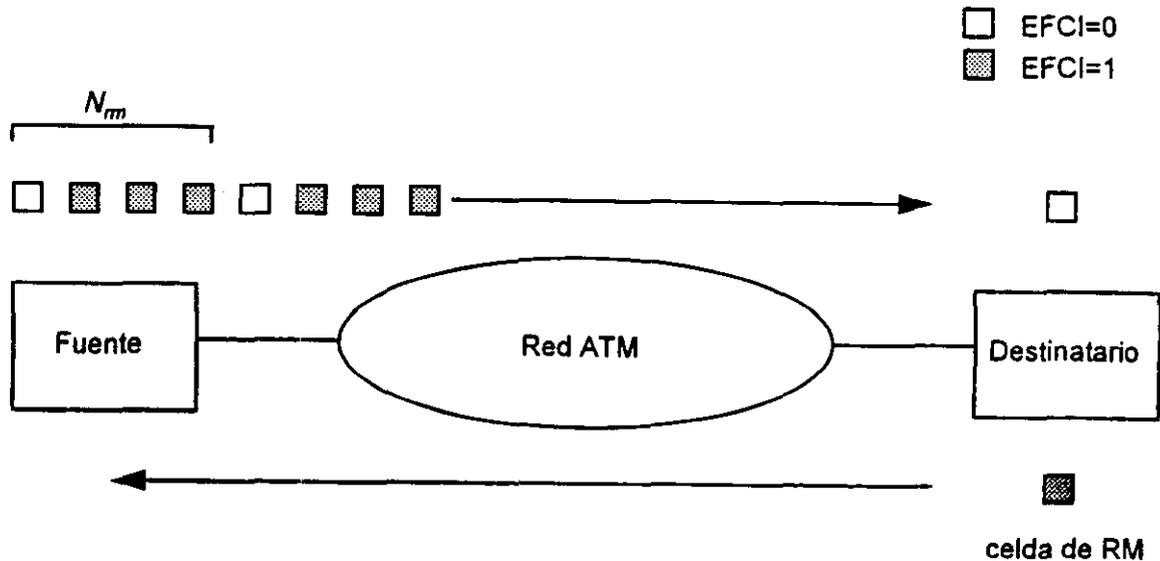


Figura 2.2 Flujo de celdas en el algoritmo PRCA (adaptado de [Chen *et al.* 1996])

La fuente a su vez, decrementa automáticamente su tasa de transmisión y la incrementa sólo cuando recibe una celda de RM. Así, la fuente podría incrementar su tasa cada  $N_m$  celdas a menos que el estado de la bandera EFCI sea modificado en algún nodo congestionado de la red o bien la celda de RM se pierda en su trayectoria hacia la fuente. El valor del incremento compensa el continuo decremento de la tasa la fuente y añade un incremento adicional fijo.

El esquema presenta algunos inconvenientes. El primero de ellos es que no es consistente con el algoritmo EFCI en donde todas las celdas se generan con la bandera EFCI inicializada en 0. Además, el mecanismo de control no es equitativo en el trato a usuarios de características diferentes [Chen *et al.* 1996].

#### 2.4.2.3 Realimentación con tasa explícita

En esta técnica las celdas de RM se envían desde la fuente hacia el extremo receptor intercaladas en el flujo de datos. Cada una de ellas contiene el campo *explicit rate* que se inicializa con la velocidad de transmisión de la fuente en ese momento.

Cada nodo conmutación de la red periódicamente determina la velocidad a la que puede enviar datos para las conexiones que pasan a través de él. Ante la

recepción de una celda de RM, el conmutador examina el campo *explicit rate* y lo disminuye si éste es mayor al valor al que puede operar. Así, al llegar al receptor, la celda de RM contiene la tasa de transmisión adecuada, la cual es establecida por el cuello de botella de su conexión.

La función del destinatario en este esquema de control consiste en enviar de regreso la celda de RM hacia la fuente y finalmente ésta se ajusta al valor de la tasa contenido en el campo *explicit rate*.

Esta técnica presenta varias cualidades atractivas. Los ajustes en la tasa de transmisión ocurren rápidamente eliminando las oscilaciones causadas por los cambios graduales. La operación del sistema es robusta ante la pérdida de celdas de RM y es menos sensible al valor inicial de la tasa de transmisión. Adicionalmente, logra tasas equitativas entre conexiones contendientes por los recursos de la red [Chen *et al.* 1996].

#### **2.4.2.4 EPRCA**

Las ideas del PRCA y de la realimentación con tasa explícita se combinan en un esquema llamado *enhanced proportional rate control algorithm* (EPRCA). De esta forma, se obtiene compatibilidad con redes que únicamente soportan EFCI y se logran incorporar, al mismo tiempo, las ventajas de la realimentación con tasa explícita.

En esta técnica se retoma la convención original para el uso de la bandera EFCI y el método de control de la fuente por la realimentación con tasa explícita. Sin embargo, la utilización específica de las celdas de RM depende de si se emplea la variante básica o la mejorada.

El mecanismo básico asume que la red sólo soporta la técnica de control EFCI y utiliza los siguientes dos campos de la celda de RM:

- DIR indica si la celda viaja en la dirección fuente-destino o viceversa.
- CI indica la presencia (CI=1) o ausencia de congestión (CI=0).

Las celdas se generan con CI=0 y este campo no es modificado durante su trayecto a través de la red. El sistema en el extremo receptor regresa las celdas

de RM con  $CI=1$  si la celda de datos precedente tenía la bandera  $EFCI=1$  (vea la figura 2.3). La fuente utiliza realimentación positiva de modo que incrementa su tasa sólo cuando las celdas de RM regresan con  $CI=0$ .

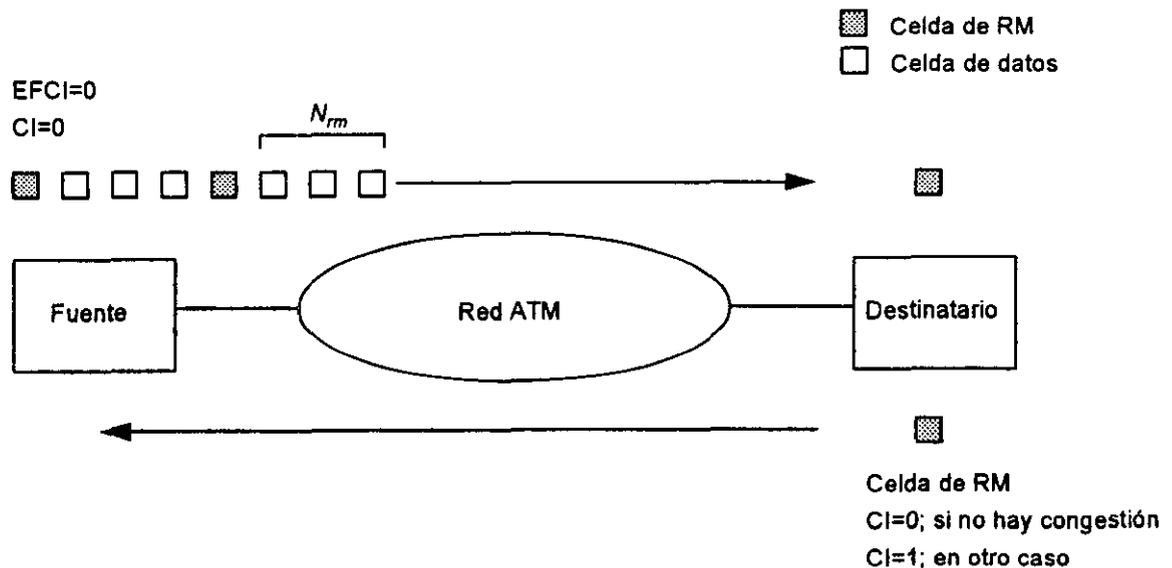


Figura 2.3 Flujo de celdas en el algoritmo EPRCA (adaptado de [Chen *et al.* 1996])

La versión mejorada asume más responsabilidades de la red. Utiliza adicionalmente los siguientes dos campos:

- ER contiene la tasa explícita inicializada a la tasa máxima de transmisión.
- ACR es el límite de transmisión permitido a la fuente al tiempo de creación de la celda de RM. No es modificado durante el recorrido de la celda.

El campo de ER se utiliza en la forma descrita anteriormente en el algoritmo de realimentación con tasa explícita. Cuando la fuente recibe una celda de RM examina los campos de ER y ACR. Si el ER es menor que el ACR, la fuente decrementa su ACR al valor de ER.

### 2.4.3 Especificación actual

En esta sección se describen las especificaciones actualmente válidas para el servicio ABR. En [ATM Forum 1996] se especifica en detalle el comportamiento estándar de la fuente, del destinatario y del conmutador.

El elemento básico en la definición del estándar es la estructura de la celda de RM que se describe a continuación. La cabecera de la celda (5 bytes) contiene el campo *payload type* (PT) que se fija en 110 (binario) para indicar que se trata de una celda de control. Los restantes 48 bytes tienen el formato mostrado en la siguiente tabla.

Tabla 2.1 Últimos 48 bytes de una celda de RM para servicio ABR

| Campo  | Nombre                                | Longitud [bits] |
|--------|---------------------------------------|-----------------|
| ID     | <i>Protocol identifier</i>            | 8               |
| DIR    | <i>Direction</i>                      | 1               |
| BN     | <i>Backward notification</i>          | 1               |
| CI     | <i>Congestion indication</i>          | 1               |
| NI     | <i>No increase</i>                    | 1               |
| RA     | <i>Request/acknowledge (no usado)</i> | 1               |
| ER     | <i>Elastic/rigid (no usado)</i>       | 1               |
| RES    | <i>Reserved (no usado)</i>            | 2               |
| ER     | <i>Explicit cell rate</i>             | 16              |
| CCR    | <i>Current cell rate</i>              | 16              |
| MCR    | <i>Minimum cell rate</i>              | 16              |
| QL     | <i>Queue length (no usado)</i>        | 32              |
| SN     | <i>Sequence number (no usado)</i>     | 32              |
| RES    | <i>Reserved</i>                       | 246             |
| CRC-10 | <i>Cyclic redundancy check</i>        | 10              |

El campo identificador de protocolo (ID) se fija en 1 para indicar que la celda pertenece a una conexión ABR. El bit de dirección (DIR) permite señalar si la celda de RM viaja en la dirección fuente-destino (0) o viceversa (1). La notificación hacia la fuente (BN) se fija únicamente en celdas BECN generadas por los conmutadores. El campo CCR lo utiliza la fuente para almacenar su tasa actual. El indicador de congestión (CI), el bit no incrementar (NI) y el campo tasa explícita (ER) se utilizan por la red para realimentar a los usuarios. En particular el campo ER indica la máxima tasa a la que se le permite transmitir a la fuente.

La especificación actual de la técnica de control se basa fundamentalmente en las ideas del EPRCA, aunque con algunas modificaciones menores [Jain *et al.* 1996]. La diferencia más significativa es que las fuentes mantienen su tasa entre cada recepción de una celda de RM en lugar de decrementarla. También se ha agregado un mecanismo que obliga a las fuentes a disminuir su tasa por un factor CDF (*cut decrease factor*) cuando la presencia de congestión origina la pérdida de una celda de RM.

Otra característica adicional es que si el tiempo entre celdas RM generadas por la fuente es excesivo, su tasa se decrementa a un valor inicial ICR (*initial cell rate*). Con esto, se recupera el ancho de banda de aquellas fuentes que no lo están utilizando [Fendick 1996].

## Capítulo 3

# Metodología de evaluación de las técnicas de control de tráfico

En este capítulo se describe el proceso de evaluación del desempeño de los algoritmos de control de tráfico para el servicio ABR. Es importante mencionar que no hay una metodología completamente general, por lo que en este trabajo se hace énfasis en lo que podría considerarse como la parte sistemática del método.

### 3.1 Criterios de comparación

El *ATM Forum* ha establecido un conjunto de criterios bajo los cuales se evalúa cualquier esquema de control de tráfico [ATM Forum 1996]. En éstos se indica que el algoritmo debe:

- ser escalable con la velocidad, la distancia, el número de usuarios y el número de conmutadores,
- dar un trato equitativo a diferentes fuentes ABR,
- ser robusto a perturbaciones menores y a fuentes que no se ajusten al contrato de servicio establecido,
- converger a estado estable,

- ser fácil de implantar en un ambiente heterogéneo sin requerir un diseño específico de los conmutadores y
- utilizar eficientemente los recursos, con requerimientos mínimos en el tamaño de los *buffers* de los conmutadores.

Otras características deseables son:

- Rápida adaptabilidad a los recursos disponibles
- Número reducido de parámetros
- Integración satisfactoria con los mecanismos de control de TCP
- Facilidades para la tarificación del servicio

### **3.2 Selección de las métricas de desempeño**

Los criterios anteriormente listados se deben traducir en conceptos mensurables a fin de poder comparar el desempeño de los algoritmos de control. El primer paso consiste en comprender claramente el tipo de servicios que una red de comunicaciones puede proporcionar. Desde algún nivel de abstracción, la red se puede considerar como el componente en un sistema de comunicaciones que, utilizando el medio físico que une sus nodos, proporciona un servicio de transferencia de datos entre los equipos terminales que la integran. A continuación se detalla cómo este servicio puede cuantificarse.

#### **3.2.1 Rendimiento, retardo, recursos consumidos y pérdidas**

Los datos se pueden agrupar en archivos, segmentos o celdas; dependiendo de si se observan al nivel de la capa de aplicación, de la de transporte o de la de red respectivamente.

El servicio de transferencia, de estas unidades de agrupamiento de datos, puede realizarse incorrectamente debido a errores de transmisión o bien por la indisponibilidad del sistema. Ambos factores se excluyen de este estudio ya que carecen de importancia en la evaluación del desempeño de algoritmos de control de tráfico.

Cuando el servicio se realiza correctamente, el desempeño se puede cuantificar básicamente por a) el rendimiento ( $\gamma$ ) es decir, la cantidad de bits transferidos por unidad de tiempo; b) el tiempo de respuesta ( $T$ ) o que tan rápido el sistema satisface las demandas de servicio; c) los recursos consumidos (principalmente utilización de los enlaces y de los *buffers* en los conmutadores) y d) las pérdidas de datos en la red. El objetivo de cualquier sistema de control es maximizar la primera métrica y minimizar las tres últimas. La relación entre algunos de estos conceptos<sup>1</sup> se muestra en la figura 3.1 en donde  $\lambda$  es la carga de tráfico ofrecida a la red, por lo que  $\lambda - \gamma$  simboliza las pérdidas.

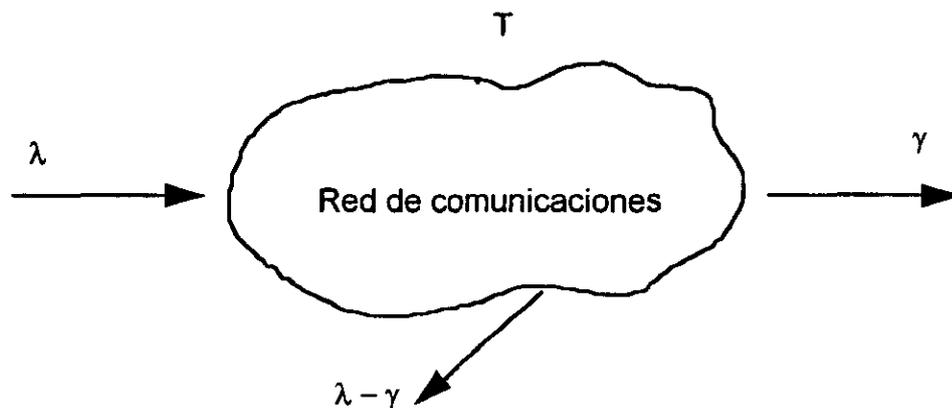


Figura 3.1 Rendimiento, tiempo de respuesta y pérdidas (adaptado de [Kleinrock 1993])

Como se mencionó anteriormente, el servicio ABR es adecuado sólo para aquellas aplicaciones que pueden operar con un ancho de banda variante en el tiempo y tolerar retardos impredecibles, por lo que las métricas de rendimiento y tiempo de respuesta carecen de importancia y no se utilizarán.

### 3.2.2 Equidad

La red es un sistema con múltiples usuarios y es deseable que todos se traten de una manera equitativa. Ningún conjunto de circuitos debe ser arbitrariamente discriminado o favorecido por la localización geográfica de sus fuentes o destinos, número de conmutadores en su trayectoria o el instante en el que se

<sup>1</sup> Rendimiento y tiempo de respuesta están relacionados. Ambos han sido combinados en un concepto llamado *potencia* [Kleinrock 1979] el cual se define como la razón del rendimiento al tiempo de respuesta. También se han propuesto otras definiciones más generales [Kleinrock 1993].

estableció la conexión. Por tanto, un índice de equidad debe agregarse como otra métrica.

Una primera aproximación a la cuantificación de la equidad puede obtenerse por la estimación de la variabilidad del rendimiento logrado por  $n$  fuentes diferentes en condiciones idénticas. La siguiente función [Jain 1991] se puede utilizar para asignar un índice de equidad al conjunto de rendimientos promedio  $\{\gamma_1, \gamma_2, \dots, \gamma_n\}$ :

$$f(\gamma_1, \gamma_2, \dots, \gamma_n) = \frac{\left( \sum_{i=1}^n \gamma_i \right)^2}{n \sum_{i=1}^n \gamma_i^2} \quad (3.1)$$

Siendo los valores de  $\gamma_i$  no negativos, la expresión anterior produce un valor entre 0 y 1. La misma ecuación puede expresarse en términos de celdas ATM o segmentos TCP. Esta es una métrica del tipo *más alto es mejor*, si todos los usuarios obtienen un rendimiento similar, el índice de equidad tiende a 1. Sin embargo, cuando las conexiones no son estadísticamente equivalentes, la expresión anterior no es aplicable.

El *ATM Forum* ha establecido un criterio más general, el principio de *max-min fairness*. Este criterio intenta asignar equitativamente el ancho de banda disponible entre las fuentes que lo pueden utilizar y maximiza el rendimiento global de la red [Bonomi *et al.* 1995]. Las ideas fundamentales se expresan a continuación y bajo la suposición de que el contrato de tráfico entre las fuentes y la red no establece ninguna garantía de ancho de banda (el MCR, o tasa mínima de transmisión de celdas para las fuentes, es cero).

En un enlace cualquiera de una red, las conexiones que compiten entre sí por ancho de banda se pueden clasificar en dos tipos:

- **Conexiones restringidas.** Son aquellas que no pueden alcanzar a transmitir a su ancho de banda asignado, debido a restricciones impuestas por su tasa máxima de transmisión de celdas (PCR), o muy probablemente, debido a que vienen limitadas por un cuello de botella en etapas previas de su ruta.

- Conexiones no restringidas. Su acceso a una mayor velocidad de transmisión es limitado únicamente por el ancho de banda disponible en el enlace considerado (éste se convierte en su cuello de botella).

El ancho de banda equitativo  $\gamma_{\text{óptimo}}$  para cada conexión no restringida que pasa a través de ese enlace es entonces:

$$\gamma_{\text{óptimo}} = \frac{C - \sum_{i \in A} r_i}{N - M} \quad (3.2)$$

donde:

$C$  : capacidad del enlace

$A$  : {conjunto de conexiones restringidas en el enlace}

$r_i$  : tasa de transmisión de la conexión  $i$

$N$  : número total de conexiones que utilizan el enlace

$M$  : número de conexiones restringidas en el enlace

La equidad de un algoritmo puede entonces estimarse comparando el rendimiento logrado, contra los valores óptimos del criterio *max-min fairness*. En múltiples trabajos de investigación se han propuesto diversos métodos para realizar esta comparación (vea por ejemplo [Fulton et al. 1997]). Nosotros utilizaremos el criterio establecido en [ATM Forum 1996] que define un índice de equidad dado por la ecuación:

$$f(x_1, x_2, \dots, x_n) = \frac{\left( \sum_{i=1}^n x_i \right)^2}{n \sum_{i=1}^n x_i^2} \quad (3.3)$$

donde  $x_i = \frac{\gamma_{i\text{-logrado}}}{\gamma_{i\text{-óptimo}}}$  y  $n$  es el número de fuentes. Los valores del rendimiento

logrado ( $\gamma_{i\text{-logrado}}$ ) se obtienen como resultado de aplicar un método de evaluación del desempeño. Los valores óptimos del rendimiento ( $\gamma_{i\text{-óptimo}}$ ) pueden quedar establecidos por limitaciones propias de las conexiones o por la fórmula 3.2 para conexiones no restringidas.

La idea fundamental expresada en la función 3.3 consiste en emplear la ecuación 3.1 utilizando el conjunto logrado de rendimientos promedio; pero escalados con respecto a los valores óptimos.

### 3.2.3 Eficiencia en la transferencia de tráfico ABR<sup>1</sup>

A fin de conocer el efecto en desempeño de la política de administración de recursos que implanta el algoritmo de control, introducimos el concepto de eficiencia. Ésta se define al nivel de la capa de aplicación como la razón entre la cantidad de datos recibidos correctamente y el límite máximo.

Esta idea de eficiencia aplicada a la transferencia de tráfico ABR, sobre un enlace, se expresa en la siguiente ecuación<sup>2</sup>:

$$eficiencia = \frac{\sum_{i=1}^n D_i}{C \cdot U \cdot T \cdot \frac{S}{53 \cdot \left\lceil \frac{S}{48} \right\rceil}} \quad (3.4)$$

en donde:

$D_i$ : cantidad de bits ABR recibidos correctamente en el destino de la conexión  $i$ . Aquellos bits correspondientes a segmentos con errores o retransmitidos no se toman en cuenta.

$n$ : número de conexiones compartiendo el enlace

$C$ : capacidad del enlace

$U$ : utilización lograda en el enlace debida únicamente a tráfico ABR

$T$ : intervalo de observación

$S$ : longitud del segmento en la capa de transporte

<sup>1</sup> En las publicaciones sobre la materia este concepto se conoce en forma genérica como *goodput*. La métrica de eficiencia aquí presentada es equivalente al *goodput*, pero calculado en forma relativa a la utilización lograda en el enlace por tráfico ABR.

<sup>2</sup> En la ecuación 3.4  $\lceil x \rceil$  es la función que da el entero más próximo pero no menor a  $x$ .

Se puede observar que la métrica calculada con la ecuación 3.4 refleja las pérdidas de celdas en los conmutadores, la ocurrencia de retransmisiones y la sobrecarga de tráfico de control introducida por el algoritmo.

En el caso de configuraciones de red con varios enlaces, la eficiencia global se puede obtener promediando las correspondientes a los enlaces individuales.

### **3.3 Método de evaluación**

Las tres técnicas comúnmente utilizadas para la evaluación del desempeño de algoritmos de control de tráfico son: mediciones sobre un sistema real, modelado analítico y simulación.

El método de efectuar mediciones no se consideró una solución viable para probar algoritmos nuevos ya que requiere de la construcción de un prototipo cuyo tiempo de diseño y fabricación, costo y poca flexibilidad no son acordes con las expectativas del proyecto. Por otro lado, el modelado analítico resulta la opción más barata y rápida, con la ventaja adicional de que permite observar las relaciones entre los diferentes parámetros del sistema. Sin embargo, el modelado de un sistema tan complejo como una red de comunicaciones requiere tantas simplificaciones y suposiciones que la confiabilidad de los resultados puede ser bastante dudosa. En contraste, la simulación en computadora puede incorporar más detalles y requerir menos suposiciones por lo que frecuentemente proporciona resultados más cercanos a la realidad. Es así que decidimos utilizar este método de evaluación y validaremos el modelo de simulación con la ayuda de un experto a través de varias estadísticas de desempeño (vea el apéndice A).

El modelo ya depurado se tendrá que someter a diversos casos de estudio para observar el comportamiento del algoritmo de control. Las simulaciones se ejecutarán por unos cuantos segundos antes de empezar a coleccionar mediciones con el objetivo de no tomar en cuenta el período transitorio inicial (causado por el arranque del programa), y posteriormente se dejarán correr por decenas de segundos para generar millones de celdas.

A lo largo de la simulación se colectarán las mediciones necesarias con las que al final de la ejecución del programa se consolidarán las estadísticas implantadas en el simulador.

De los resultados de las simulaciones, los valores obtenidos para las métricas listadas en la sección 3.2 proporcionarán un estimado del comportamiento de la técnica de control de tráfico. Finalmente, el desempeño de los diferentes algoritmos evaluados se comparará de acuerdo a los criterios mencionados anteriormente en el punto 3.1.

# Capítulo 4

## Modelo de simulación propuesto

El desempeño de cada algoritmo de control se estimará por la simulación, en computadora, de su comportamiento ante la carga de trabajo originada por un generador de tráfico. Esto da lugar a un modelo de simulación cuyas componentes se describen en este capítulo.

### 4.1 Arquitectura en capas del modelo

Con la finalidad de obtener resultados realistas, y por lo tanto confiables, se creó un modelo de simulación que incorpora todos los detalles de las interacciones entre los protocolos de las diferentes capas de un sistema real. La arquitectura propuesta para el modelado *de cada conexión* se muestra en la figura 4.1.

En la dirección fuente-destino el simulador incluye el modelado de protocolos de alto nivel (capas de aplicación y transporte) así como protocolos de red (capas de adaptación, ATM y física). A continuación se comentan brevemente las interacciones entre estos elementos.

En el lado de la fuente:

- Fuente de tráfico. Genera el tiempo de espera entre el fin de la transmisión de un archivo (indicado por la capa inferior) y la generación del siguiente. También define el tamaño de cada archivo.
- Capa TCP. Segmenta el archivo que recibe de la fuente y envía los fragmentos hacia la capa ATM de acuerdo con los mecanismos de control del protocolo TCP. La versión que utilizamos es *4.3 BSD Reno* con las características de incremento aditivo, decremento multiplicativo, recuperación rápida, retransmisión rápida, inicio lento y retransmisión adaptiva a través del algoritmo de Van Jacobson [Peterson *et al.* 1996].
- Capa de adaptación ATM. Fragmenta segmentos TCP en celdas ATM. El envío de éstas hacia la red, es regulado por el algoritmo de control de tráfico. En el modelo de simulación se asume que el adaptador ATM reside en el servidor TCP; de esta forma, el análisis de los mecanismos de control puede hacerse sin la intervención de otros elementos tales como ruteadores, enlaces *Ethernet*, etc.

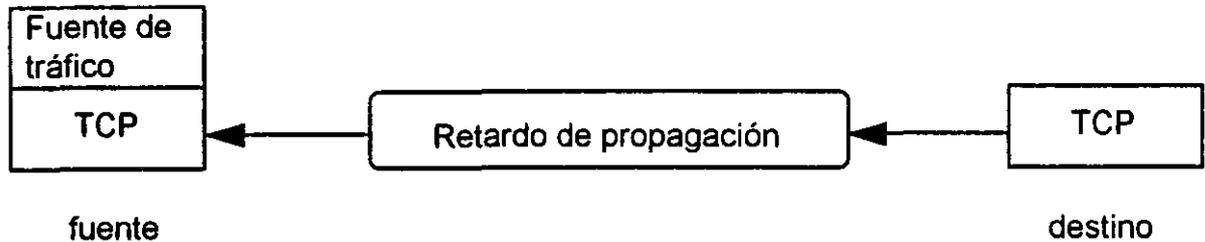
En el destino:

- Capa de adaptación ATM. Ensambla las celdas ATM en segmentos TCP y los envía a la capa de transporte.
- Capa TCP. Envía los reconocimientos (ACK) correspondientes a los segmentos TCP recibidos. Esta entidad también actúa como sumidero de datos.



Figura 4.1 Arquitectura de capas del modelo de simulación

En la trayectoria de retorno para los paquetes ACK del protocolo TCP, se consideró que sólo era significativo considerar el retardo de propagación. En nuestro modelo de simulación, este envío de mensajes se efectúa directamente entre las capas TCP -de los extremos fuente y destino- sin la intervención de algún otro protocolo de más bajo nivel. Esta situación se muestra en la figura 4.2.



**Figura 4.2** Modelo de la trayectoria de regreso para los paquetes ACK del TCP

## 4.2 Topologías de red para la evaluación del desempeño

La configuración de la red de prueba es un componente muy importante en la creación de un ambiente realista para la evaluación del desempeño de algoritmos de control de tráfico. Para este fin, numerosas publicaciones sobre la materia reportan la utilización de diversas topologías, aunque la mayoría de ellas se puede clasificar dentro de alguna de las dos descritas a continuación.

### 4.2.1 Configuración 2-nodos

En forma genérica se puede considerar que un nodo de conmutación en una red de comunicaciones se encuentra expuesto al tráfico generado por etapas previas, y que él a su vez, transmite datos hacia uno o más nodos subsecuentes. Los enlaces que unen los nodos pueden tener diversas características de retardo y velocidad de transmisión. Sin pérdida de generalidad, esta situación se puede representar como se muestra en la figura 4.3.

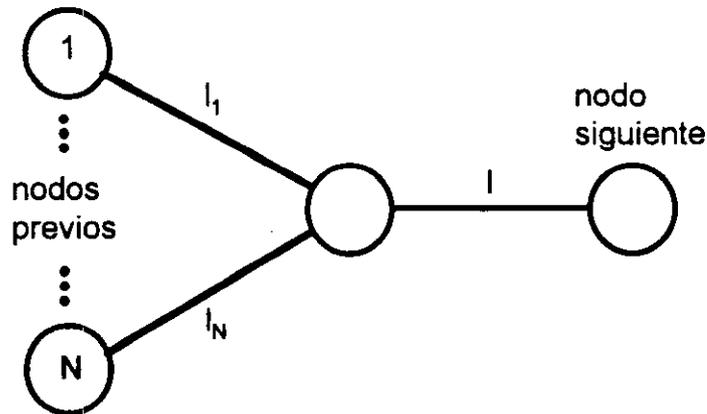


Figura 4.3 Un nodo genérico de la red

Una topología que puede potencialmente capturar esta situación consiste en dos nodos de conmutación interconectados entre sí, en donde el primero recibe y conmuta el tráfico proveniente de varias fuentes y al segundo se encuentran conectados los destinos, tal como se muestra en la figura 4.4. El comportamiento de las fuentes se determina por los modelos de tráfico y algoritmos de control. Opcionalmente y para representar cuellos de botella en segmentos previos de la red, las fuentes pueden limitarse a sí mismas (utilizando sólo una fracción del ancho de banda asignado a ellas). Los canales de comunicación  $L_{fc}$  pueden tener diferentes retardos de propagación para modelar de esta forma nodos previos localizados a diferentes distancias.

Debido a su sencillez y realismo, esta configuración 2-nodos ha sido extensamente utilizada en trabajos de investigación.

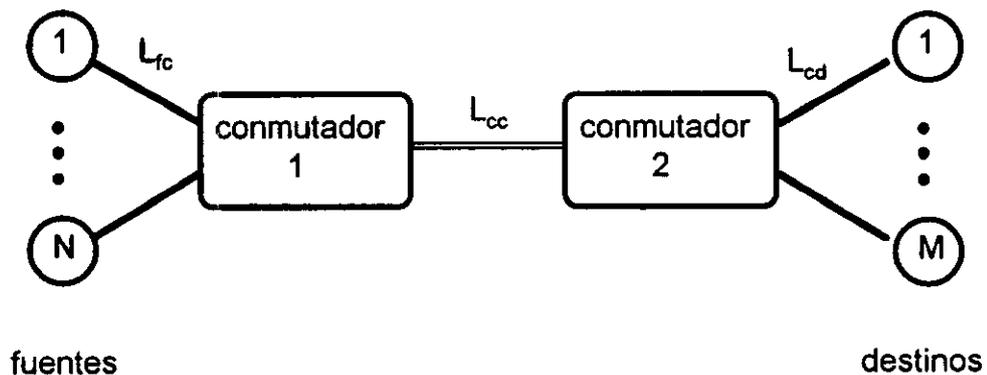


Figura 4.4 Configuración 2-nodos

#### 4.2.2 Configuración Max-Min Fairness

La configuración 2-nodos con una cuidadosa selección de parámetros permite simular prácticamente cualquier situación. Sin embargo, preferimos utilizar una topología más compleja para investigar el comportamiento de los algoritmos de control en un ambiente en el que coexisten múltiples conmutadores, así como diferentes velocidades de propagación, retardos y cuellos de botella.

La segunda configuración seleccionada se puede catalogar dentro de las topologías del tipo *Max-Min Fairness*. Éstas han sido ampliamente utilizadas para investigar la habilidad de las técnicas de control de tráfico para distribuir equitativamente el ancho de banda disponible entre todas las fuentes contendientes por los recursos de una red [Arulambalam *et al.* 1996].

La versión de la configuración *Max-Min Fairness* que usamos consta de cuatro conmutadores y doce pares fuente-destino. Estos elementos se conectan entre sí tal y como se muestra en la figura 4.5, en donde  $f_0, f_1, \dots, f_{11}$  representan las fuentes y  $d_0, d_1, \dots, d_{11}$  los destinos. Las doce conexiones resultantes quedan agrupadas, de acuerdo a sus trayectorias, en cinco clases diferentes y cada una de éstas tiene su propio cuello de botella.

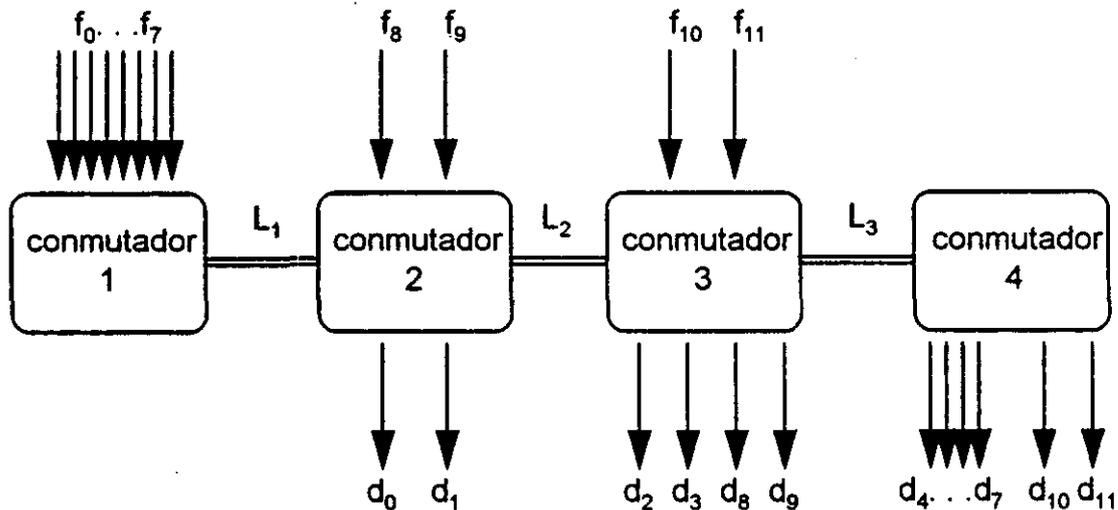


Figura 4.5 Configuración *Max-Min Fairness*

Como un ejemplo de las situaciones que crea este arreglo, consideremos las 8 conexiones que circulan a través del enlace  $L_2$  ( $L_1 \leq L_2$ ) entre los conmutadores 2 y 3. Bajo un esquema de control justo, aquéllas conexiones que vienen desde las fuentes  $f_2 \dots f_7$  deberían adquirir una velocidad de transmisión limitada únicamente por su cuello de botella en  $L_1$ , en tanto que las que inician en  $f_8$  y  $f_9$  se deberían repartir equitativamente el ancho de banda restante.

### 4.3 Modelado de la fuente

Un modelo de tráfico, aplicado a una red de computadoras, intenta caracterizar el tiempo entre demandas sucesivas de servicio y la naturaleza de los recursos requeridos. Estos dos elementos se pueden simular por generadores de tráfico teóricos o empíricos.

La realidad que el modelo de tráfico intenta capturar se encuentra en cambio constante ya que los patrones de utilización de la red reflejan el comportamiento de la gente, los servicios al usuario y las tecnologías de red [Frost *et al.* 1994]. Por esta razón, es deseable que un modelo pueda ajustarse a diferentes servicios variando sólo un pequeño conjunto de valores en los parámetros que lo definen. Otras características deseables [Stamoulis *et al.* 1994] son sencillez, precisión confiable en su análisis teórico y facilidad tanto de implantación como de validación.

#### 4.3.1 El tipo de tráfico a modelar

Entre las aplicaciones que ABR transportará se incluyen servicios proporcionados al público por sistemas de cómputo (tal como los tradicionalmente transportados por enlaces de X.25), transferencia de archivos, emulación de terminales, etcétera. Estos programas son usualmente muy sensibles a pérdidas de información (demandan una probabilidad muy pequeña de error), mientras que sus requerimientos de retardo no son muy estrictos. Actualmente TCP se considera como el mecanismo más factible para asegurar la integridad de los datos en los extremos y regular su transferencia [Stevens 1990]. En consecuencia, el flujo de información resultante puede contener pequeños grupos

aislados de caracteres (sesiones remotas) o grandes agrupaciones de *bytes* (transferencias de objetos).

#### 4.3.2 Modelos propuestos para fuentes de datos

Los modelos de los procesos de *Bernoulli* y de *Poisson* [Papoulis 1991], usados muy frecuentemente en estudios analíticos y de simulación, son inapropiados para el modelado del tráfico de datos ya que asumen que no hay correlación entre llegadas. En [Gusella 1990] se muestra que el proceso de arribo tiene un comportamiento *intermitente por ráfagas* que se conserva incluso a diferentes escalas de observación y que la suposición de independencia entre llegadas no se justifica. Existe una gran posibilidad de que la recepción de un dato sea seguida por una segunda dentro de un tiempo determinístico. Tres factores explican esta alta probabilidad: los objetos transmitidos por los protocolos de red son mucho más grandes que el tamaño máximo del paquete de las capas inferiores; los protocolos de *requerimiento-respuesta*, como TCP, generan tiempos entre llegadas cuyas distribuciones tienen modas muy marcadas; y por último, las fuentes de tráfico producen ráfagas. Éstas se pueden caracterizar por períodos alternantes de actividad e inactividad, en donde además, el tiempo durante el cual se transmiten datos, es mucho más corto que el de inactividad [Schwartz 1996].

Una gran variedad de modelos se han propuesto para las fuentes de ráfagas, uno muy comúnmente utilizado es el de *Poisson por lotes* en el cual, un cierto número de celdas  $r$  se generan instantáneamente en cada llegada de un proceso de Poisson [Kleinrock 1975]. Una gran ventaja de este modelo es la facilidad de su análisis, ya que cuando el lote de celdas se considera como un solo arribo, la interacción entre el proceso resultante y un servidor, se puede analizar como un sistema *M/M/1* con llegadas por lotes. El tamaño del lote puede ser determinístico o aleatorio; pero cuando se le restringe a una distribución geométrica, las expresiones analíticas se pueden obtener más fácilmente. Desafortunadamente, y en base a mediciones experimentales, se ha mostrado que el tráfico en una red real dista mucho del comportamiento predicho por un proceso de Poisson por lotes (véanse por ejemplo [Jain et al. 1986] y [Fowler et al. 1991]).

Otro modelo muy popular es el de la *fente ON-OFF*. En esta aproximación, sólo se generan datos -a una tasa fija-, cuando el proceso se encuentra en el estado "ON". Generalmente los períodos de actividad e inactividad se distribuyen exponencialmente, con lo que la fuente se puede considerar como un proceso de Markov de dos estados [Çinlar 1975] y su análisis se simplifica. El sistema se puede caracterizar completamente por su tasa promedio  $m$ , tasa máxima  $M$  y la media del número de celdas generadas durante el período de actividad. Para el tráfico de ráfagas (por ejemplo la interconexión entre redes LAN), la tasa máxima es alta y la tasa promedio de la fuente es baja ( $m/M$  probablemente sea menor a 0.1 [Yin *et al.* 1993]).

Los dos modelos anteriormente descritos están íntimamente relacionados. Consideremos un caso en el que la tasa máxima de una fuente ON-OFF es alta. Consecuentemente, todas las celdas generadas durante el período de actividad, aparecerán en un tiempo tan corto que se pueden considerar creadas *instantáneamente* y el período de inactividad, exponencialmente distribuido, se convierte en el tiempo entre lotes de celdas. De esta forma y conforme la tasa máxima de la fuente ON-OFF se incrementa, su comportamiento tiende al proceso de Poisson por lotes.

Debido a esta relación entre ambas aproximaciones, el modelo de Poisson por lotes puede ser más apropiado que la fuente ON-OFF para representar casos de condiciones extremas durante el análisis del desempeño de un sistema de colas. Sin embargo, cuando el tiempo entre generación de celdas no es despreciable, la utilización de este modelo puede ser cuestionable.

Una tercera aproximación que se extiende naturalmente de las dos anteriores es el *modelo de trenes* [Jain *et al.* 1986], en el que una fuente se considera que puede estar activa o inactiva. La celdas se producen en lotes (trenes) sólo durante el período activo. Sin embargo, y en contraposición al proceso de Poisson por lotes, sólo se genera una celda a la vez. Para describir la actividad de la fuente se utilizan tres parámetros especificados por funciones de probabilidad de varianza finita:

- tiempo entre llegadas de trenes subsecuentes
- tiempo entre llegadas de celdas pertenecientes al mismo tren
- número de celdas pertenecientes al mismo tren

Este último modelo de tráfico proporciona poca variabilidad a escalas de tiempo más largas que la duración de los trenes y supone implícitamente que variaciones de baja frecuencia son de pequeña amplitud o altamente predecibles. Sin embargo, en [Fowler *et al.* 1991] se proporciona evidencia experimental de que la variabilidad, en el tráfico de datos, a estas escalas de tiempo es enorme y de que las anteriores suposiciones no son generalmente válidas.

Las observaciones reportadas por Fowler y Leland en [Fowler *et al.* 1991] describen que la forma del tráfico en redes LAN consiste en la formación de "picos" sobre "ondas" de mayor escala; las cuales a su vez, se crean sobre "abultamientos" más largos. Estudios estadísticos basados en cientos de millones de mediciones colectadas sobre un período de cuatro años, en redes *Ethernet*, confirman este comportamiento *autosimilar* [Leland *et al.* 1994], con diferentes grados de autosimilitud dependiendo de la carga en la red.

El subestimar la presencia de estas *ráfagas intermitentes a diferentes escalas de tiempo* puede conllevar a consideraciones erróneas de dimensionamiento en los componentes de un sistema de comunicaciones. Los modelos de tráfico anteriormente descritos no son capaces de capturar estas correlaciones sobre un amplio rango de escalas de tiempo; por lo que, en los últimos años un gran esfuerzo se ha invertido en el desarrollo de métodos para generar tráfico sintético autosimilar. Los modelos más populares están basados en procesos estocásticos autosimilares (vea [Adas 1997] y referencias ahí contenidas), aunque otras técnicas para reproducir la naturaleza fractal del tráfico autosimilar también se han desarrollado [Forys *et al.* 1995].

Investigaciones recientes presentadas en [Willinger *et al.* 1997] explican formalmente<sup>1</sup> la causa de este comportamiento autosimilar en términos de la naturaleza de las interacciones fuente-destino. El artículo se apoya en estudios realizados por otros investigadores que indican: a) que las relaciones hombre-computadora ocurren en un amplio rango de escalas de tiempo; y b) una gran varianza encontrada tanto en la medida de los objetos residentes en servidores, como en el tiempo de CPU consumido por un proceso UNIX. Apoyados en estas

<sup>1</sup> Se ha confirmado la presencia de autosimilitud en el tráfico de redes LAN y WAN; pero la causa del fenómeno ha sido satisfactoriamente explicada sólo para las primeras.

observaciones, Willinger *et al.* simulan cada fuente con el tradicional modelo ON-OFF; pero utilizando distribuciones de varianza infinita para representar los períodos de actividad e inactividad y demuestran que al sobreponerlas, el tráfico resultante es autosimilar con correlaciones de larga escala.

Estos hallazgos indican que es posible utilizar modelos tradicionales con distribuciones de varianza infinita y así lograr reproducir patrones complejos de tráfico en forma simple y sin requerir modelos altamente parametrizados.

### 4.3.3 Modelo implantado

El estado actual de las teorías de teletráfico, sugiere que las mejores representaciones del tráfico de datos se obtienen con modelos que incorporan la propiedad de autosimilitud. Las aproximaciones más populares para generar patrones sintéticos autosimilares simulan la transmisión de datos multicanalizados en un enlace. Sin embargo, sólo el modelado de fuentes individuales y la implantación directa de los algoritmos de control de congestión de TCP permitirán una simulación realista. Esto es debido a que los algoritmos de control de congestión y de flujo de TCP introducen oscilaciones y periodicidades en el patrón de tráfico [Paxson *et al.* 1995].

Como se comentó anteriormente, el flujo de datos TCP consiste de tráfico interactivo (pequeños grupos de caracteres) y de grandes agrupaciones de *bytes*. La gran diferencia en volumen de información transportado por estas dos componentes ocasiona que la dinámica del proceso de transmisión, en enlaces compartidos, esté dominada por unas cuantas ráfagas grandes debidas a transferencias de archivos. En este tipo de tráfico el proceso de llegada de paquetes está determinado por factores dependientes de la red tales como el ancho de banda disponible, la presencia de congestión y detalles de los algoritmos -de control de tráfico- del protocolo de transporte [Paxson *et al.* 1995]. El patrón del flujo de datos interactivos refleja principalmente la dinámica del comportamiento de las personas y además tiene poca importancia en situaciones de congestión. Por estas razones decidimos modelar sólo el tráfico generado por transferencias de archivos al nivel de la aplicación.

Al nivel de la fuente consideramos que el flujo de datos se podía dividir en sesiones y éstas a su vez, en períodos de actividad e inactividad. Así, el modelo quedó formado por las siguientes partes:

- Inicio de nuevas sesiones FTP. Debido a que generalmente cada sesión involucra un usuario diferente, este proceso de llegadas carece de memoria y puede ser correctamente modelado por un proceso de Poisson [Paxson *et al.* 1995]. El valor de la tasa media de arribo se estima de la figura 10 en [Danzig *et al.* 1992].
- Número de archivos transmitidos por sesión. Se selecciona aleatoriamente usando el modelo empírico presentado en [Danzig *et al.* 1991] y [Danzig *et al.* 1992].
- Tiempo de inactividad entre cada transferencia de archivos. De acuerdo con [Paxson *et al.* 1995] se simula con una variable aleatoria *log-normal* y parámetros tomados de la figura 8 del mismo artículo.
- El número de *bytes* en cada archivo. Se modela como una variable aleatoria *Pareto* y *parámetro de forma* entre 0.9 y 1.4 [Paxson *et al.* 1995]. El *parámetro de localización* [Evans *et al.* 1993] se interpreta como el tamaño mínimo de los archivos generados. Pudimos haber tomado este otro dato directamente de [Danzig *et al.* 1992]; pero decidimos fijar su valor a uno mayor ( $2 \times 10^6$ ) de modo que se reflejara la tendencia a la utilización de archivos cada vez más grandes [Jain *et al.* 1986].

Observe que los períodos de actividad e inactividad se modelan con distribuciones de "cola pesada"<sup>1</sup> (varianza infinita) por lo que el tráfico multicanalizado resultante exhibe comportamiento autosimilar.

#### 4.4 Tráfico de fondo

En un escenario realista pueden coexistir, compartiendo los mismos recursos, múltiples conexiones con diferentes requerimientos de calidad en el servicio. A fin de investigar la integración de ABR con otros servicios que proporciona la red

<sup>1</sup> Una variable aleatoria  $U$  se denomina de "cola pesada" si  $P[U \leq u] \approx u^{-a}$  (cuando  $u \rightarrow \infty$ ,  $a > 0$ ) [Erramilli *et al.* 1994].

ATM, decidimos ejecutar algunas simulaciones utilizando tráfico de fondo como representante de los recursos consumidos por otros servicios.

En la selección del tráfico de fondo consideramos que servicios clase A (tales como señales de voz) siendo de tasa de transmisión constante, resultaban de poca utilidad en el estudio. En contraste, el tráfico ocasionado por servicios clase B (tales como transmisiones de vídeo en forma comprimida) presenta múltiples cambios de caudal y puede provocar situaciones más demandantes para los mecanismos de control de ABR.

Para generar tráfico de servicio clase B podemos utilizar cualquier formato de vídeo comprimido; sin embargo, utilizar un algoritmo de compresión estandarizado se ha convertido en una prioridad ya que esto puede reducir costos de producción y resolver el problema de interoperabilidad entre el equipo de diferentes fabricantes. El esfuerzo más importante en este proceso se conoce con el nombre del grupo que lo empezó: MPEG, acrónimo de *Moving Picture Experts Group*. La premisa de MPEG es que una señal de vídeo con su audio asociado pueden comprimirse a una tasa de aproximadamente 1.5 Mbits/s manteniendo una calidad aceptable.

El algoritmo de compresión de vídeo MPEG [Gall 1991] se basa en dos técnicas: compensación del movimiento para la reducción de la redundancia temporal y compresión por medio de la transformada coseno discreta para la reducción de la redundancia espacial. La información de movimiento se comprime usando códigos de longitud variable y se transmite junto con la información espacial.

El estándar de codificación MPEG logra altas tasas de compresión. Sin embargo, ya que el algoritmo fue originalmente desarrollado para almacenamiento y recuperación de vídeo comprimido, la técnica de codificación no toma en cuenta las características de las redes de telecomunicaciones. El patrón de tráfico resultante exhibe picos cuasi-periódicos con requerimientos relativamente altos de velocidad de transmisión, aunque la tasa promedio es mucho menor.

La figura 4.6 muestra un ejemplo de una bien conocida secuencia MPEG de un segundo. Las diversas implantaciones del algoritmo de compresión producen características ligeramente diferentes; pero en general todas ellas tienen esta forma.

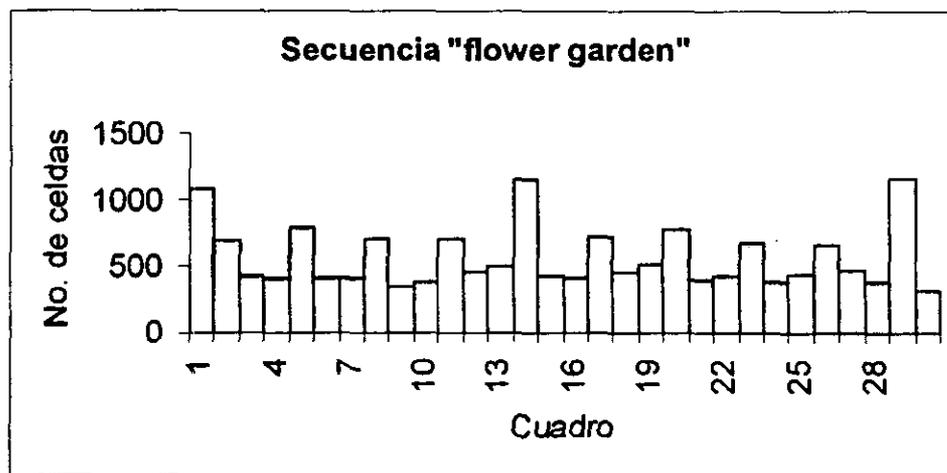


Figura 4.6 Ejemplo de una secuencia MPEG

En nuestro caso utilizaremos para algunas simulaciones el tráfico resultante de la multicanalización de tres secuencias de vídeo en formato MPEG. Cada una de ellas consta de 708 cuadros y deben transmitirse a una tasa de 30 imágenes por segundo. Con el patrón de tráfico VBR resultante de 23.6s se pueden obtener secuencias más largas empalmándolo sucesivamente. La figura 4.7 muestra el tráfico de 30 segundos utilizado en las simulaciones.

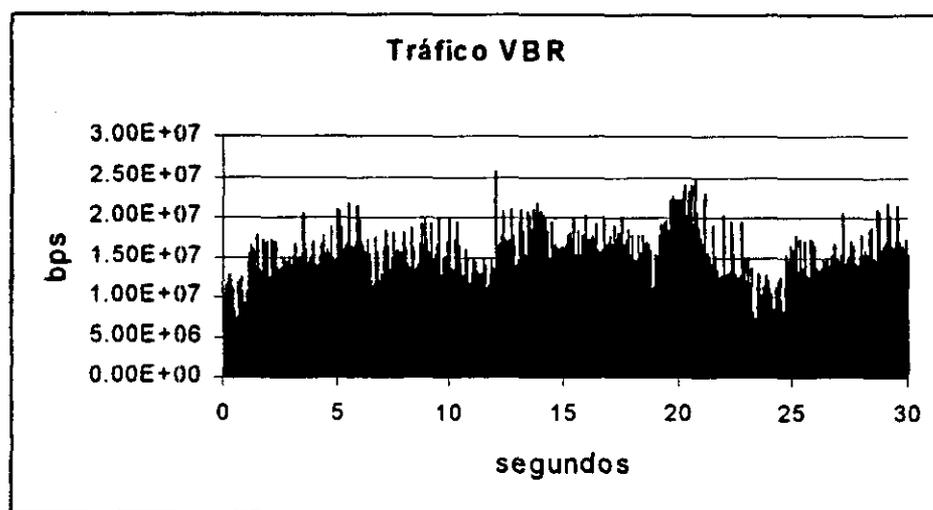


Figura 4.7 Tráfico resultante de la multicanalización de 3 secuencias MPEG

La presencia de cambios abruptos en el patrón de tráfico VBR ocasionará situaciones demandantes para los algoritmos de control. Éstos deberán adaptarse rápidamente al ancho de banda cambiante para utilizar óptimamente los recursos y evitar la acumulación de celdas en el *buffer* de los conmutadores.

# Capítulo 5

## Evaluación del algoritmo UT

En este capítulo se presenta y evalúa el desempeño de un algoritmo de control de tráfico basado en tasa explícita. A fin de realizar una evaluación completa y confiable, se proponen diversos casos de estudio bajo los cuales se observa el funcionamiento de la técnica de control.

### 5.1 Algoritmo *Uniform Tracking*

En [Fulton *et al.* 1997] sus autores proponen un nuevo esquema de control de tráfico llamado *Uniform Tracking* (UT). Entre las principales características del algoritmo destaca su simplicidad. UT logra las tasas de transmisión del criterio *max-min fairness* sin requerir un conocimiento explícito del número de conexiones activas, el número de las fuentes limitadas<sup>1</sup> o la velocidad de transmisión que éstas pueden alcanzar.

---

<sup>1</sup> Una fuente limitada se define como aquella incapaz de alcanzar una tasa de transmisión equitativa con respecto a las demás fuentes conectadas al mismo conmutador, debido a limitaciones propias (valor de PCR) o por su localización geográfica dentro de la red (cuellos de botella).

A continuación se describe brevemente la operación de UT. En cada intervalo de control, denotado por  $t_c$ , los conmutadores UT calculan una tasa equitativa de transmisión,  $FS_{eff}$ . Esta cantidad se almacena en el campo ER (de las celdas RM que viajan en la dirección destino-fuente); pero sólo si resulta ser mas pequeño que el valor previamente almacenado ahí. El  $FS_{eff}$  se determina a través de la expresión:

$$FS_{eff} = \frac{\rho_T C - r_B(n)}{N_{eff}} \quad (5.1)$$

en donde:

- $\rho_T$  es la utilización límite del enlace de salida
- $C$  es la capacidad del enlace
- $r_B(n)$  es el ancho de banda ocupado por el tráfico de fondo correspondiente al intervalo  $n$  (las componentes de alta frecuencia se eliminan con un filtro)

$\rho_T$  es un parámetro configurable con valores entre 0 y 1. El número efectivo de fuentes  $N_{eff}$  se estima por:

$$N_{eff} = \max \left\{ \frac{r_A(n)}{FS_{eff}}, 1 \right\} \quad (5.2)$$

en donde:

- $r_A(n)$  es la tasa del tráfico ABR medida en el intervalo  $n$
- $\frac{r_A(n)}{FS_{eff}}$  es una media, sobre una ventana deslizante de valores previos de  $FS_{eff}$

De la explicación anterior, es claro que el intervalo de control  $t_c$  es un parámetro importante para UT, en el sentido de que debe escogerse suficientemente grande para obtener estimados precisos del caudal (de modo que cambios tan pequeños como una celda no deben interpretarse como un incremento significativo en la tasa) y permitir una implantación práctica. En contraste, la selección de un intervalo muy largo puede ocasionar una respuesta torpe. Los autores en [Fulton *et al.* 1997] han propuesto y validado la siguiente ecuación para seleccionar el intervalo de control:

$$t_c \approx \max \left\{ \frac{1}{0.005 \rho_T C}, 10ms \right\} \quad (5.3)$$

en donde la capacidad del enlace C se expresa en unidades de celdas/ms.

### 5.2 Valores de los parámetros de las simulaciones

Para todos los casos, la capacidad máxima del *buffer* en el adaptador ATM fue de 1,130 celdas. Para el modelo de TCP la resolución del temporizador se fijó en 0.5s, el tamaño del segmento a 1,536 bytes y la longitud de la ventana se estableció igual al producto del retardo por ancho de banda para cada caso. El conjunto de parámetros utilizados por el algoritmo UT se muestra en la tabla 5.1.

Tabla 5.1 Parámetros UT

| Símbolo         | Descripción   | Valor                     |
|-----------------|---|---------------------------|
| T <sub>c</sub>  | Intervalo de control  | 10 ms                     |
| MCR             | Tasa de transmisión mínima  | 2 x 10 <sup>4</sup> bps   |
| ICR             | Tasa de transmisión inicial   | 1/t <sub>c</sub> celdas/s |
| PCR             | Tasa de transmisión máxima  | Velocidad del enlace*     |
| ρ <sub>T</sub>  | Utilización límite del enlace   | 0.90**                    |
| RDF             | Factor de decremento de la tasa de transmisión                          | 1/32768 *                 |
| RIF             | Factor de incremento de la tasa de transmisión                          | 1*                        |
| TBE             | Exposición a transitorios del <i>buffer</i>                             | 16x10 <sup>6</sup> **     |
| CDF             | Factor de decremento en congestión                                      | 1/16*                     |
| ADTF            | Factor de tiempo para decrementos en ACR                                | 0.5s*                     |
| M <sub>rm</sub> | Parámetro que controla la capacidad asignada a las celdas RM y de datos | 2*                        |
| T <sub>rm</sub> | Límite superior en el tiempo entre celdas RM (dirección hacia adelante) | 0.1s*                     |
| N <sub>rm</sub> | Número de celdas de datos por cada RM (dirección hacia adelante)        | 32*                       |

\* Valor por defecto definido en [ATM Forum 1996]

\*\* Configurable

El *retardo fijo de vuelta completa* (FRTT) se seleccionó de acuerdo al tiempo total de propagación para cada caso. El tamaño del *buffer* en los conmutadores se estableció a un valor suficientemente grande de modo que se pudieran observar los valores que alcanzaba el tamaño de la cola. Consecuentemente, no se registró pérdida de celdas.

### 5.3 Simulaciones con la configuración 2-nodos

Decidimos utilizar la configuración 2-nodos, mostrada en la figura 5.1, para investigar el comportamiento del algoritmo de control de tráfico en ambientes de diversas características. En cada simulación utilizamos la misma velocidad de transmisión en todos los enlaces. Consideramos fuentes ABR compartiendo un destino común y capaces de transmitir con todo el ancho de banda asignado a ellas. El retardo de propagación entre conmutadores ( $L_{cc}$ ) se asumió como el doble del retardo entre un conmutador y puntos terminales ( $L_{fc}$  o  $L_{cd}$ ).

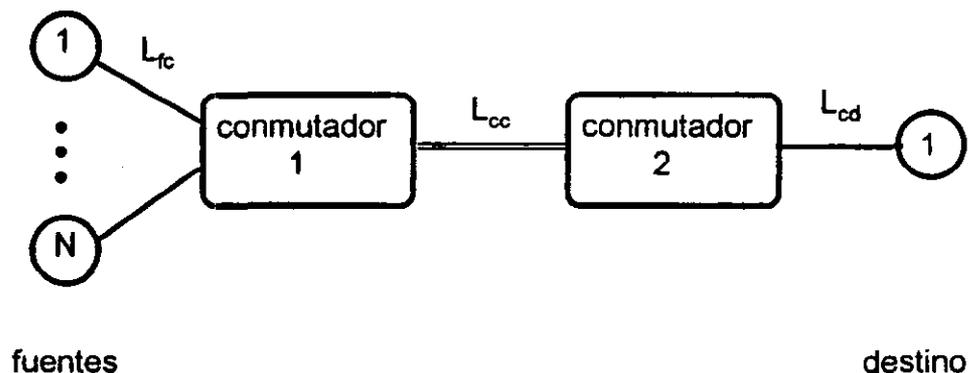


Figura 5.1 Configuración 2-nodos

A fin de facilitar la presentación de resultados, decidimos dividir el conjunto de simulaciones en dos grupos. El primero de ellos, mostrado en la tabla 5.2, tuvo el propósito de investigar el comportamiento del algoritmo bajo diferentes velocidades, retardos de propagación y número de fuentes. En todos los casos el período de inicialización fue de 5 s.

Tabla 5.2 Primer grupo de casos de estudio para la configuración 2-nodos

| Caso | Número y tipo de las fuentes de datos | Velocidad de transmisión en todos los enlaces* | Retardo de propagación de vuelta completa [ms] | Intervalo de simulación [s] |
|------|---------------------------------------|--|--|-----------------------------|
| 1    | 10 FTP                                | DS3  | 1  | 50                          |
| 2    | 100 FTP                               | OC3  | 1  | 50                          |
| 3    | 10 FTP                                | DS3  | 25   | 100                         |
| 4    | 100 FTP                               | OC3  | 25   | 100                         |
| 5    | 10 FTP                                | DS3  | 50   | 150                         |
| 6    | 100 FTP                               | OC3  | 50   | 150                         |

\* DS3=40.704Mbps, OC3=149.76Mbps<sup>1</sup>

Con el segundo grupo de simulaciones, mostrado en la tabla 5.3, intentamos investigar la integración de ABR con otros servicios. Las simulaciones propuestas consisten básicamente en los anteriores casos de estudio 1, 3 y 5; pero esta vez utilizando como tráfico de fondo el flujo de datos generado por la transmisión multicanalizada de tres secuencias de vídeo en formato MPEG. En todos los casos el tiempo de inicialización fue de 30 s.

Tabla 5.3 Segundo grupo de casos de estudio para la configuración 2-nodos

| Caso | Número y tipo de las fuentes de datos | Velocidad de transmisión en todos los enlaces* | Retardo de propagación de vuelta completa [ms] | Intervalo de simulación [s] |
|------|---------------------------------------|--|--|-----------------------------|
| 7    | 10 FTP                                | DS3  | 1  | 30                          |
| 8    | 10 FTP                                | DS3  | 25   | 30                          |
| 9    | 10 FTP                                | DS3  | 50   | 30                          |

\* DS3=40.704Mbps

<sup>1</sup> Estos valores son las tasas de transferencia que quedan disponibles a ATM después de considerar los bits de control en la capa de física.

### 5.3.1 Resultados para la configuración 2-nodos sin tráfico de fondo

La tabla 5.4 muestra el tamaño de la cola observado en el conmutador 1 para el primer grupo de simulaciones. Como podemos ver, aquél fue de 750 celdas en la peor situación (caso 5). Esto muestra que el esquema es escalable ante diferentes retardos de propagación y número de fuentes.

**Tabla 5.4** Algoritmo UT: tamaño de la cola en el conmutador 1

| Caso | Media<br>[celdas] | Máximo<br>[celdas] |
|------|-------------------|--------------------|
| 1    | 1.6               | 156                |
| 2    | 3.2               | 24                 |
| 3    | 1.6               | 259                |
| 4    | 3.5               | 31                 |
| 5    | 3.0               | 750                |
| 6    | 3.5               | 82                 |

El tamaño de la cola indica la rapidez de respuesta del algoritmo de control ante situaciones cambiantes, ya que la acumulación de celdas ocurre únicamente durante períodos transitorios, cuando una nueva fuente empieza a transmitir, y antes de que el mecanismo de control pueda ajustar correctamente las tasas de transmisión en las fuentes. Por ello, es de esperarse que la longitud de la cola se incremente con valores crecientes del retardo de propagación (observe los grupos de simulaciones 1-3-5 y 2-4-6).

Es interesante observar que el máximo fue más grande en los casos de 10 que en los de 100 fuentes, mientras que con la media ocurrió lo contrario. La longitud media se relaciona directamente con el número de períodos transitorios, el cual es más grande en el caso de 100 fuentes.

Para explicar la diferencia en las longitudes máximas de las colas, consideremos primero los escenarios con 100 fuentes (casos 2, 4 y 6). Cuando un gran número de ellas se encuentran transmitiendo, el hecho de que otra fuente se active, no produce una perturbación significativa. En contraste, para las simulaciones de 10

fuentes (casos 1, 3 y 5), alguna de ellas que comience a transmitir, involucra un incremento de al menos 11% en el tráfico ofrecido a la red. Por ello, en la configuración con 100 fuentes, el conmutador se encuentra expuesto a ráfagas transitorias de menor intensidad que en el caso de 10 fuentes.

El tamaño máximo de la cola para el conmutador 2 (no mostrado en la tabla anterior) fue siempre de una celda. Aquí no hay razón para la acumulación de celdas ya que las velocidades de transmisión a la entrada y a la salida de este nodo son iguales.

En todos los casos, la utilización promedio del enlace entre los conmutadores fue igual al 100% del valor límite (seleccionada como el 0.90 de la capacidad nominal del enlace).

Los valores del índice de equidad y eficiencia obtenidos para este conjunto de simulaciones se proporcionan en la tabla 5.5. Como se puede observar, el comportamiento de UT, en general, fue muy cercano al óptimo en ambos aspectos. Los valores obtenidos de eficiencia reflejan la sobrecarga introducida por el tráfico de control, ya que el algoritmo envía una celda de RM por cada 32 de datos.

Tabla 5.5 Algoritmo UT: índice de equidad y eficiencia

| Caso | Índice de equidad | Eficiencia [%] |
|------|-------------------|----------------|
| 1    | 0.9752            | 96.97          |
| 2    | 0.9818            | 96.97          |
| 3    | 0.9819            | 96.97          |
| 4    | 0.9829            | 96.97          |
| 5    | 0.9803            | 96.97          |
| 6    | 0.9915            | 96.97          |

### 5.3.2 Resultados para la configuración 2-nodos con tráfico de fondo

Para este conjunto de simulaciones, el comportamiento observado en la cola del conmutador 1 se muestra en la tabla 5.6. El tamaño máximo de la cola para el conmutador 2 (no mostrado en la tabla) fue de una celda en todos los casos.

Tabla 5.6 Algoritmo UT: tamaño de la cola en el conmutador 1

| Caso | Media<br>[celdas] | Máximo<br>[celdas] |
|------|-------------------|--------------------|
| 7    | 6.5               | 275                |
| 8    | 7.1               | 293                |
| 9    | 8.1               | 351                |

Como se puede observar, el tamaño de la cola se incrementó como una función del retardo de propagación. Sin embargo, el máximo alcanzado resultó menor a su contraparte sin tráfico de fondo (caso 5 de la tabla 5.4). Esto es debido a que ancho de banda consumido por el tráfico VBR ocasionó que las fuentes ABR se activaran con valores generalmente menores de ACR. En consecuencia, las rafagas transitorias fueron de menor intensidad.

En contraste, aunque menores en intensidad, ocurrieron más períodos transitorios debido a la alta variabilidad en requerimientos de ancho de banda del tráfico de video. Esto ocasionó que los valores medios resultaron más grandes en este grupo de simulaciones que en los casos sin tráfico de fondo.

En todos los casos, la utilización promedio del enlace entre los conmutadores fue igual al 100% del valor límite (0.90 de la capacidad nominal del enlace).

El índice de equidad y la eficiencia obtenidos para este conjunto de simulaciones se proporcionan en la tabla 5.7. Como se puede observar, el comportamiento del algoritmo UT en general fue muy cercano al óptimo. De nuevo, la eficiencia se afectó únicamente por la sobrecarga introducida por el tráfico de control.

Tabla 5.7 Algoritmo UT: Índice de equidad y eficiencia

| Caso | Índice de equidad | Eficiencia [%] |
|------|-------------------|----------------|
| 7    | 0.9984            | 96.97          |
| 8    | 0.9986            | 96.97          |
| 9    | 0.9987            | 96.97          |

Los resultados con esta primera configuración muestran la efectividad del esquema UT para mantener bajos niveles de ocupación en los buffers de los conmutadores, limitando, por lo tanto, el retardo de transferencia de las celdas a través de la red. Adicionalmente, el algoritmo logra altos porcentajes de utilización y eficiencia y converge a valores equitativos en las tasas de transmisión, independientemente del instante en el que las fuentes se activen.

#### 5.4 Simulaciones con la configuración *Max-Min Fairness*

El análisis de los resultados para la configuración mostrada en la figura 5.2 nos permitirán caracterizar la forma en la que el esquema UT distribuye los recursos disponibles entre fuentes expuestas a diferentes retardos de propagación.

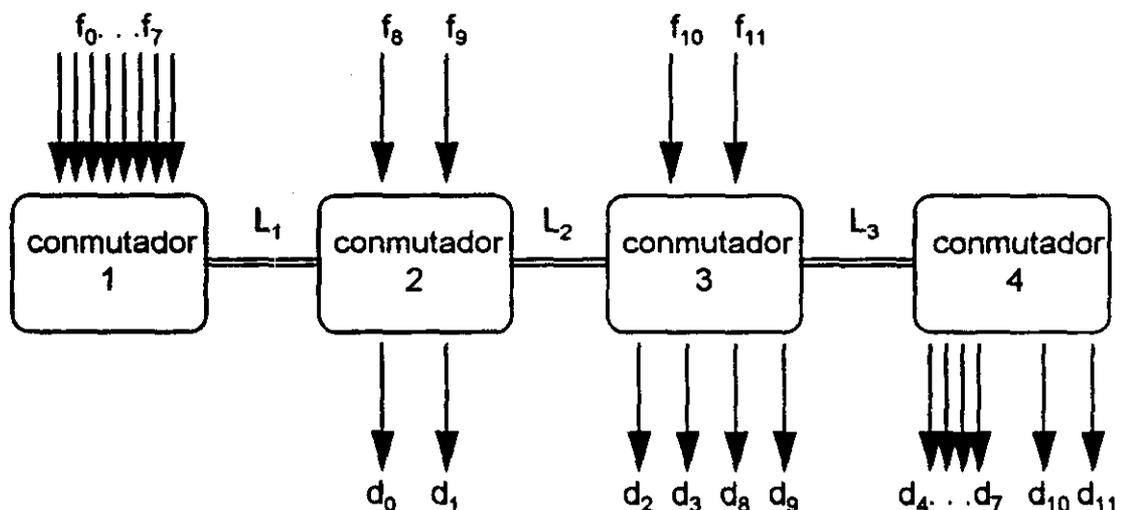


Figura 5.2 Configuración *Max-Min Fairness*

En nuestro caso decidimos fijar en 5ms los retardos de propagación de los enlaces entre los conmutadores (L1, L2 y L3). La conexión entre un punto extremo (fuente o destino) con el conmutador al cual se conecta se asumió con un retardo de 100µs. Los casos de estudio considerados se muestran en la siguiente tabla:

**Tabla 5.8** Casos de estudio para la configuración *Max-Min Fairness*

| Caso | Velocidad de transmisión* |     |        |
|------|---------------------------|-----|--------|
|      | L1                        | L2  | L3     |
| 1    | DS3                       | DS3 | DS3    |
| 2    | DS3                       | DS3 | 25Mbps |
| 3    | 25Mbps                    | DS3 | DS3    |

\*DS3=40.704Mbps

#### 5.4.1 Resultados para la configuración *Max-Min Fairness*

La tabla 5.9 muestra las longitudes máxima y media de la cola obtenidas para los conmutadores 1 a 3. Para el conmutador 4 la longitud máxima de la cola fue de una celda.

**Tabla 5.9** Algoritmo UT: longitudes de las colas

| Caso | Conmutador 1   |                 | Conmutador 2   |                 | Conmutador 3   |                 |
|------|----------------|-----------------|----------------|-----------------|----------------|-----------------|
|      | Media [celdas] | Máxima [celdas] | Media [celdas] | Máxima [celdas] | Media [celdas] | Máxima [celdas] |
| 1    | 40.2           | 2,177           | 6.2            | 762             | 21.9           | 1,283           |
| 2    | 19.7           | 1,850           | 12.9           | 1,553           | 21.0           | 2,039           |
| 3    | 2.4            | 573             | 6.8            | 780             | 4.5            | 565             |

La tabla 5.10 proporciona la utilización del enlace de salida lograda en los tres casos de estudio. Este valor se calculó relativo a la utilización límite seleccionada (90% de la capacidad nominal). Se puede observar que, para el caso 1, el enlace que interconecta el conmutador 2 con el 3, se convirtió en el cuello de botella de la red ya que presentó una utilización claramente superior a los restantes. En los casos 2 y 3, los enlaces de 25 Mbps presentaron la utilización más alta como se esperaba.

Tabla 5.10 Algoritmo UT: utilización del enlace de salida

| Caso | Utilización del enlace a la salida del conmutador |       |       |
|------|---|-------|-------|
|      | 1   | 2     | 3     |
| 1    | 0.893   | 0.992 | 0.761 |
| 2    | 0.926   | 0.969 | 0.974 |
| 3    | 0.992   | 0.768 | 0.598 |

La tabla siguiente muestra los resultados en las métricas de índice de equidad y eficiencia, como se puede observar, en todos los casos se alcanzaron los valores óptimos. El valor obtenido de eficiencia indica que ésta se ve únicamente afectada por la sobrecarga del tráfico de control introducida por el algoritmo.

Tabla 5.11 Algoritmo UT: eficiencia

| Caso | Índice de equidad | Eficiencia [%] |
|------|-------------------|----------------|
| 1    | 1.0000            | 96.97          |
| 2    | 1.0000            | 96.97          |
| 3    | 1.0000            | 96.97          |

Los resultados con esta segunda configuración muestran la efectividad del algoritmo UT para soportar simultáneamente trayectorias cortas y largas; manteniendo un bajo nivel de ocupación en los buffers de los conmutadores, logrando una alta utilización de los enlaces y distribuyendo equitativamente los recursos independientemente del número de conmutadores en la ruta y de la localización física de los pares fuente-destino.

# Evaluación del algoritmo CDT y comparación con UT

En este capítulo se presenta y evalúa una aproximación diferente al control de tráfico para servicio ABR. Esta otra técnica utiliza el control basado en créditos para regular la transferencia de datos. El desempeño del algoritmo se compara contra el observado en el capítulo anterior para el control de tasa explícita UT y se enfatizan las diferencias más significativas.

### 6.1 Algoritmo *Controlled Dynamic Transfer*

En [QFC 1997] se describe el protocolo *Controlled Dynamic Transfer* (CDT) que se caracteriza por regular la transferencia de celdas a través del intercambio de mensajes que contienen el estado de los *buffers* en elementos contiguos de la red. El algoritmo logra incrementos agresivos en las tasas de transmisión y evita la pérdida de celdas por congestión. Estas características se mantienen a lo largo de la trayectoria de la conexión.

La idea básica del esquema es que el transmisor envía celdas pendientes por transmitirse, cada vez que el receptor le informa que hay espacio para recibirlas.

Usando el mismo mecanismo de control, el receptor transferirá las celdas al siguiente nodo en la trayectoria, liberando de esta forma espacio en su *buffer*. Conforme estas celdas se envían, el receptor le informa al transmisor, con un mensaje especial, la cantidad de espacio que ha quedado disponible. Dicho mensaje se conoce como *actualización del estado del buffer* o BSU (*buffer state update*).

El estado del receptor se controla por las siguientes variables de estado, en donde el índice  $i$  indica la conexión  $i$ -ésima:

- $Rx\_Counter[i]$ , cuenta la cantidad de celdas que se han recibido.
- $Fwd\_Counter[i]$ , indica el número de celdas que se han transferido al siguiente nodo en la trayectoria. El valor actualizado de esta variable se incluye en cada mensaje BSU.
- $Policy\_Limit[i]$ , especifica el máximo número de celdas que pueden almacenarse en el *buffer*.
- $N2\_Counter[i]$ , almacena la cantidad de celdas que se han transferido, hacia el siguiente nodo en la ruta, desde el instante en el que el último mensaje de actualización se envió de regreso al transmisor.

Asimismo, el estado del transmisor se especifica con las siguientes variables de estado:

- $Tx\_Counter[i]$ , indica la cantidad de celdas que se han transmitido.
- $BSU\_Fwd\_Count[i]$ , almacena el valor de la variable  $Fwd\_Counter$  recibido en el último mensaje BSU.
- $Policy\_Limit[i]$ , indica el tamaño del *buffer* asignado en el receptor para la conexión.

Un transmisor puede continuar enviando celdas siempre que se cumpla la siguiente desigualdad:

$$Tx\_Counter[i] - BSU\_Fwd\_Count[i] \leq Policy\_Limit[i] \quad (6.1)$$

Los contadores se deben inicializar a cero cuando se crea una nueva conexión. En esta etapa inicial, las variables  $Policy\_Limit[i]$  del transmisor y receptor pueden negociarse o bien se puede utilizar un valor por defecto previamente

almacenado. Asimismo, estos valores se pueden renegociar durante el tiempo de vida de la conexión.

A continuación se listan los mensajes de control utilizados por CDT. Todos ellos se envían en celdas RM.

- *Actualización del estado del buffer o BSU (buffer state updates)*. Enviados por el receptor al transmisor después de que aquél envía  $N_2$  celdas. Esta operación actualiza el valor de  $BSU\_Fwd\_Count[i]$  en el transmisor. El valor de  $N_2$  es también negociable al momento de establecer de la conexión.
- *Verificación del estado del buffer o BSC (buffer state check)*. Estos mensajes se envían periódicamente del transmisor al receptor incluyendo una copia del valor de  $Tx\_Counter[i]$  a fin de verificar el valor de  $Rx\_Counter[i]$  y para compensar celdas o mensajes de BSU perdidos por errores de transmisión. Cuando el receptor obtiene un mensaje BSC, comparará el valor de su variable  $Rx\_Counter[i]$  con el valor recibido de  $Tx\_Counter[i]$ . Si los valores difieren, sus variables se actualizarán a través de las siguientes dos ecuaciones:

$$Fwd\_Counter[i] = Fwd\_Counter[i] + (Tx\_Counter[i] - Rx\_Counter[i]) \quad (6.2)$$

$$Rx\_Counter[i] = Tx\_Counter[i] \quad (6.3)$$

El receptor responderá a cada mensaje BSC con un BSU conteniendo un valor actualizado de  $Fwd\_Counter[i]$ , aún si el receptor ha enviado menos que  $N_2$  celdas desde la última actualización.

- *Estado del buffer y retardo de propagación de vuelta completa o BSR (Buffer State RTT)*, se utilizan para medir los retardos de propagación en las direcciones fuente-destino y destino-fuente por medio de la inserción de sellos de tiempo. Cuando un elemento en la red recibe una celda como esta, lo único que tiene que hacer es enviarla de regreso al transmisor.

El control de flujo en VP y VC al mismo tiempo (control extendido) se propone como una opción. En tal caso, todos los VCs de un VP específico comparten un conjunto de créditos. El transmisor continuará enviando celdas sólo si la

desigualdad (6.1) se satisface para las variables correspondientes a ambos, el VP y el VC de la conexión. Nuestro estudio utiliza control por VP; pero los resultados se pueden igualmente aplicar a control extendido.

## 6.2 Valores de los parámetros de las simulaciones

Al igual que para las simulaciones del algoritmo UT, en todos los casos la capacidad máxima del *buffer* en el adaptador ATM fue de 1,130 celdas. Para el modelo de TCP la resolución del temporizador se fijó en 0.5 s, el tamaño del segmento a 1,536 *bytes* y la longitud de la ventana se estableció igual al producto del retardo por ancho de banda en cada caso.

En el modelo de simulación de CDT, el valor de  $N_2$  se fijó a 32 celdas y el tiempo entre mensajes BSC se seleccionó como 1s.

A fin de lograr una comparación justa, cada caso de estudio se probó bajo dos diferentes criterios en la selección del tamaño de los *buffers* en los conmutadores. Los dos conjuntos de *buffers* se calcularon de modo que:

- La memoria total requerida sea similar a la máxima utilizada por UT. Cuando el número de celdas por conexión resultaba ser un valor fraccionario, se asignaron al menos dos.
- Se logre una transmisión ininterrumpida. Esto se garantizó tomando en consideración el producto retardo por ancho de banda (BDP). El ancho de banda considerado fue la tasa de transmisión del enlace que une la fuente con el conmutador más cercano. El retardo considerado fue el tiempo que transcurre entre la transmisión de la primera celda hasta la recepción del primer mensaje BSU (enviado por el receptor después de transferir  $N_2$  mensajes al siguiente elemento en la ruta).

## 6.3 Simulaciones con la configuración 2-nodos

Con el objetivo de poder realizar una comparación directa, se utilizó la misma configuración de red y los mismos casos de estudio que para UT. Por claridad, éstos se reproducen a continuación en la figura 6.1 y tablas 6.1 y 6.2.

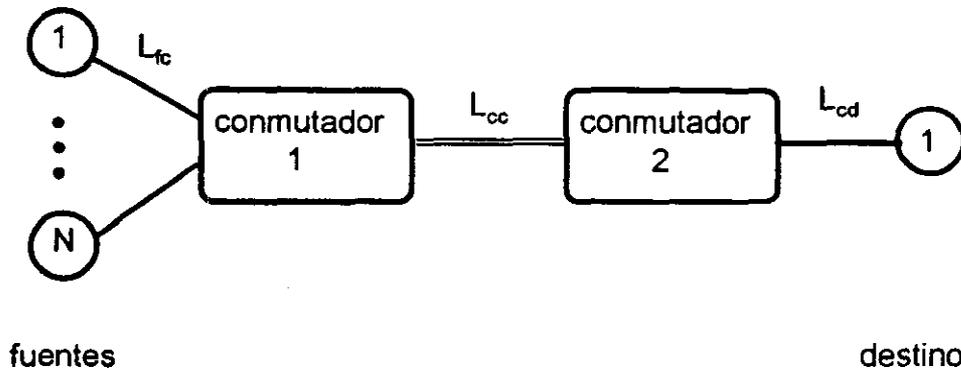


Figura 6.1 Configuración 2-nodos

Al igual que en el caso UT todos los enlaces utilizan la misma velocidad de transmisión, las fuentes ABR comparten un destino común y son capaces de transmitir con todo el ancho de banda asignado a ellas. El retardo de propagación entre conmutadores ( $L_{cc}$ ) se asumió como el doble del retardo entre un conmutador y puntos terminales ( $L_{fc}$  o  $L_{cd}$ ).

Como en las simulaciones de UT, el primer grupo de simulaciones listado en la tabla 6.1 no utiliza tráfico de fondo, mientras que en el mostrado en la tabla 6.2, se utilizó como tráfico VBR, la multicanalización de tres secuencias de video diferentes en formato MPEG. El tiempo de inicialización fue de 5s para el primer grupo y de 30s para el segundo.

Tabla 6.1 Primer grupo de casos de estudio para la configuración 2-nodos

| Caso | Número y tipo de las fuentes de datos | Velocidad de transmisión en los enlaces* | Retardo de propagación de vuelta completa [ms] | Intervalo de simulación [s] |
|------|---------------------------------------|--|--|-----------------------------|
| 1    | 10 FTP                                | DS3                                      | 1  | 50                          |
| 2    | 100 FTP                               | OC3                                      | 1  | 50                          |
| 3    | 10 FTP                                | DS3                                      | 25   | 100                         |
| 4    | 100 FTP                               | OC3                                      | 25   | 100                         |
| 5    | 10 FTP                                | DS3                                      | 50   | 150                         |
| 6    | 100 FTP                               | OC3                                      | 50   | 150                         |

DS3=40.704Mbps, OC3=149.76Mbps

**Tabla 6.2 Segundo grupo de casos de estudio para la configuración 2-nodos**

| Caso | Número y tipo de las fuentes de datos | Velocidad de transmisión en los enlaces* | Retardo de propagación de vuelta completa [ms] | Intervalo de simulación [s] |
|------|---------------------------------------|--|--|-----------------------------|
| 7    | 10 FTP                                | DS3                                      | 1  | 30                          |
| 8    | 10 FTP                                | DS3                                      | 25   | 30                          |
| 9    | 10 FTP                                | DS3                                      | 50   | 30                          |

\* DS3=40.704Mbps

### 6.3.1 Resultados para la configuración 2-nodos sin tráfico de fondo y buffers establecidos con el criterio "similar a UT"

El tamaño total del *buffer* se fijó al mismo valor en ambos conmutadores. Éste se muestra en la siguiente tabla, la cual además presenta las estadísticas de ocupación obtenidas para el conmutador 1, que es el expuesto a situaciones más demandantes debido a que recibe directamente al tráfico proveniente de las fuentes.

**Tabla 6.3 Algoritmo CDT: ocupación de los buffers establecidos con el criterio "similar a UT"**

| Caso | Buffer en el conmutador 1 [celdas] |                 |                  |
|------|------------------------------------|-----------------|------------------|
|      | Total                              | Ocupación media | Ocupación máxima |
| 1    | 156                                | 33.8            | 145              |
| 2    | 200                                | 80.8            | 200              |
| 3    | 259                                | 90.8            | 247              |
| 4    | 200                                | 99.6            | 200              |
| 5    | 750                                | 293.2           | 657              |
| 6    | 200                                | 97.5            | 200              |

En todos los casos el tamaño promedio de la cola resultó mas grande que la correspondiente para UT (vea la tabla 5.4). La diferencia se debe a que el algoritmo CDT tiende a retener tantos datos como se puedan almacenar mientras

espera la llegada de créditos. Esto es un efecto de la combinación del control basado en créditos y un *buffer* pequeño.

La diferencia más notable en desempeño se obtuvo en la utilización del enlace entre los conmutadores, mostrado en la siguiente tabla. Este valor resultó en general anormalmente bajo, excepto para la simulación de 10 fuentes y un retardo de propagación pequeño (caso 1). La figura 6.2 muestra una comparación gráfica de esta métrica entre ambos esquemas de control.

Tabla 6.4 Algoritmo CDT: Utilización del enlace entre los conmutadores, criterio "similar a UT"

| Caso | Utilización del enlace de salida |
|------|----------------------------------|
| 1    | 0.914                            |
| 2    | 0.839                            |
| 3    | 0.164                            |
| 4    | 0.044                            |
| 5    | 0.125                            |
| 6    | 0.010                            |

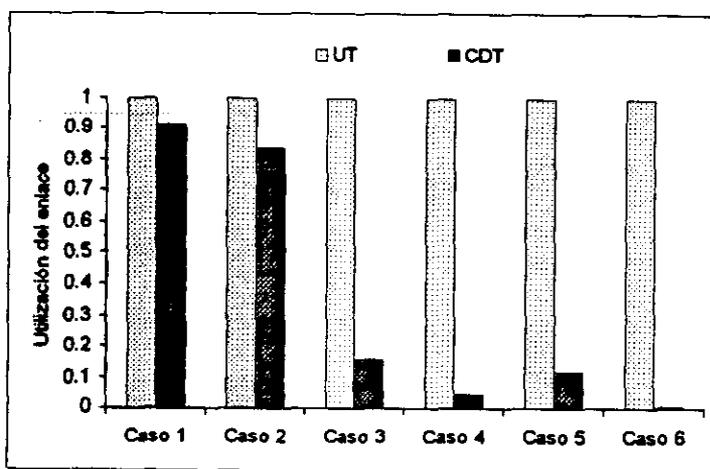


Figura 6.2 Comparación de utilización entre UT y CDT (criterio "similar a UT")

Excepto por aquellos casos con pequeño retardo de propagación, es claro que los *buffers* no fueron suficientemente grandes para que CDT lograra una utilización satisfactoria.

En los escenarios con muchas fuentes y largos retardos de propagación (casos 4 y 6), la baja utilización originó una gran cantidad de retransmisiones de segmentos TCP debido a tiempos de expiración: 2,864 en el caso 4 y 4,181 en el 6.

La tabla 6.5, mostrada a continuación, despliega los valores del índice de equidad y eficiencia obtenidos. En la mayoría de los casos, el índice de equidad resultó comparable al obtenido en las simulaciones de UT (vea la tabla 5.5). En algunos casos, los valores de eficiencia fueron muy cercanos a los óptimos. Sin embargo, observe que el valor de esta métrica se ve reducido en los casos 4 y 6 debido a las retransmisiones.

**Tabla 6.5** Algoritmo CDT: índice de equidad y eficiencia, criterio "similar a UT"

| Caso | Índice de equidad | Eficiencia [%] |
|------|-------------------|----------------|
| 1    | 0.9899            | 99.99          |
| 2    | 0.9979            | 99.97          |
| 3    | 0.9949            | 99.90          |
| 4    | 0.9980            | 93.29          |
| 5    | 0.9999            | 99.91          |
| 6    | 0.9990            | 76.28          |

En base a estos resultados, se puede concluir que, con un *buffer* pequeño como el utilizado por UT, el esquema CDT presenta una baja utilización de los enlaces, ocasionando retransmisiones y una disminución en eficiencia. Este comportamiento se hace evidente en escenarios con largos retardos de propagación y gran número de fuentes.

### 6.3.2 Resultados para la configuración 2-nodos con tráfico de fondo y buffers establecidos con el criterio "similar a UT"

El tamaño total del *buffer* se fijó al mismo valor en ambos conmutadores. Éste se muestra en la siguiente tabla, en donde además se presentan los resultados del tamaño de la cola en el *buffer* para el conmutador 1.

Tabla 6.6 Algoritmo CDT: ocupación de los *buffers* establecidos con el criterio "similar a UT"

| Caso | Buffer en el conmutador 1 [celdas] |                 |                  |
|------|------------------------------------|-----------------|------------------|
|      | Total                              | Ocupación media | Ocupación máxima |
| 7    | 275                                | 102.5           | 260              |
| 8    | 294                                | 100.9           | 280              |
| 9    | 351                                | 159.0           | 320              |

Al comparar el anterior grupo de valores con los mostrados en la tabla 5.6 para UT, se puede observar que la ocupación media del *buffer* para CDT fue mayor por al menos un orden de magnitud en todos los casos.

De nueva cuenta, la diferencia más notable en desempeño se puede observar en la utilización del enlace entre los conmutadores. Estos resultados se muestran en la siguiente tabla, observe la baja utilización en las simulaciones con largos retardos de propagación. La figura 6.3 muestra una comparación gráfica de esta métrica entre ambos esquemas de control.

Tabla 6.7 Algoritmo CDT: Utilización del enlace entre los conmutadores, criterio "similar a UT"

| Caso | Utilización del enlace de salida |
|------|----------------------------------|
| 7    | 1.000                            |
| 8    | 0.519                            |
| 9    | 0.411                            |

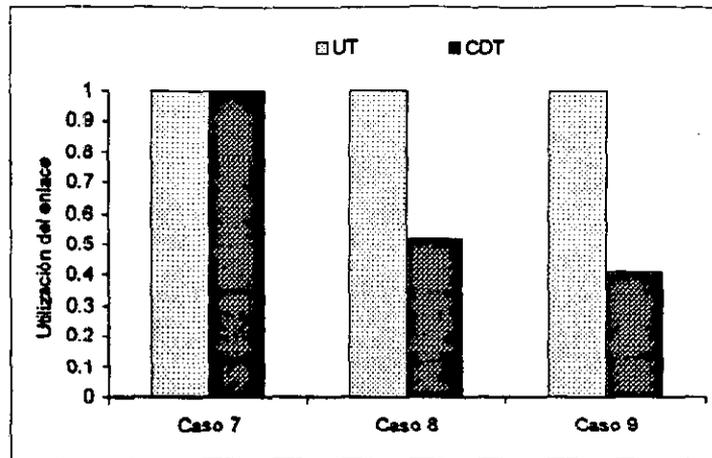


Figura 6.3 Comparación de utilización entre UT y CDT (criterio "similar a UT")

La baja utilización, en este grupo de simulaciones, no ocasionó retransmisiones de segmentos por tiempos de expiración en la capa TCP. Sin embargo, excepto por el caso 1 con un pequeño retardo de propagación, los *buffers* no fueron suficientemente grandes para que CDT lograra una utilización satisfactoria.

La tabla 6.8, mostrada a continuación, despliega los valores del índice de equidad y eficiencia obtenidos. Ambas métricas resultaron comparables a las obtenidas en las simulaciones de UT (vea la tabla 5.7).

Tabla 6.8 Algoritmo CDT: índice de equidad y eficiencia, criterio "similar a UT"

| Caso | Índice de equidad | Eficiencia [%] |
|------|-------------------|----------------|
| 7    | 0.9906            | 99.99          |
| 8    | 1.0000            | 99.93          |
| 9    | 1.0000            | 99.83          |

En base a estos resultados, se confirma que, con *buffers* pequeños, el esquema CDT presenta una baja utilización de los enlaces y que este comportamiento se hace evidente en escenarios con largos retardos de propagación.

### 6.3.3 Resultados para la configuración 2-nodos sin tráfico de fondo y buffers calculados de acuerdo al criterio BDP

A ambos conmutadores se les asignó el mismo tamaño total de *buffer*, cuyo valor se muestra en la siguiente tabla. Ésta muestra además el tamaño de la cola observado en el conmutador 1. La ocupación máxima del buffer en el conmutador 2 fue de dos celdas.

Tabla 6.9 Algoritmo CDT: ocupación de los *buffers* calculados con el criterio BDP

| Caso | Buffer en el conmutador 1 [celdas] |                 |                  |
|------|------------------------------------|-----------------|------------------|
|      | Total                              | Ocupación media | Ocupación máxima |
| 1    | 3,340                              | 1,949.7         | 3,286            |
| 2    | 324,500                            | 102,925.9       | 130,896          |
| 3    | 6,220                              | 3,042.0         | 5,591            |
| 4    | 430,500                            | 351,937.4       | 428,266          |
| 5    | 15,210                             | 7,681.8         | 12,790           |
| 6    | 761,600                            | 333,292.2       | 751,997          |

Al comparar los anteriores resultados contra los obtenidos para el algoritmo UT (tabla 5.4), se puede observar que CDT utilizó mucha más memoria que UT; la figura 6.4 despliega gráficamente esta gran diferencia (observe que la escala vertical es logarítmica). A causa de los grandes retardos de encolamiento, en el caso 6 ocurrieron un total de 82,817 retransmisiones de segmentos por tiempos de expiración en la capa TCP.

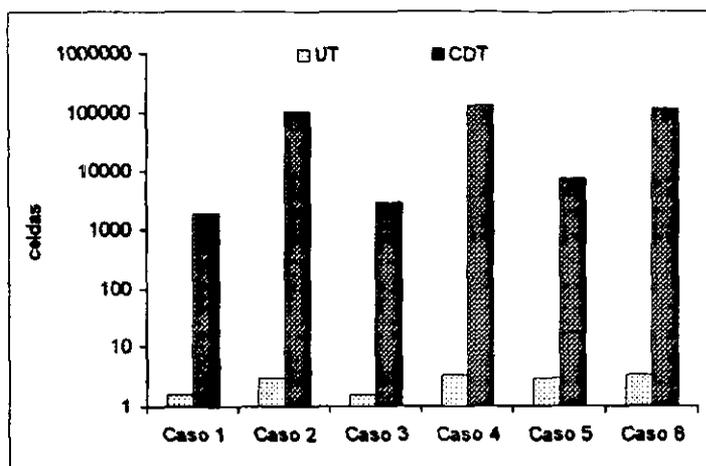


Figura 6.4 Comparación de la longitud media de la cola entre UT y CDT (criterio BDP)

La utilización del enlace fue prácticamente el 100% en todos los casos. Esto muestra que con un tamaño suficientemente grande en los *buffers* de los conmutadores, CDT es capaz de lograr altas tasas de utilización.

La tabla 6.10 muestra los valores obtenidos del índice de equidad y eficiencia. Se puede ver que en estos aspectos el desempeño de CDT fue óptimo. Sin embargo, al comparar contra los resultados obtenidos anteriormente para UT (tabla 5.5), se ve que la diferencia realmente no es muy significativa. Observe que hay una disminución en eficiencia debido a las retransmisiones en el caso 6.

Tabla 6.10 Algoritmo CDT: índice de equidad y eficiencia, criterio BDP

| Caso | Índice de equidad | Eficiencia [%] |
|------|-------------------|----------------|
| 1    | 0.9703            | 99.99          |
| 2    | 0.9623            | 99.97          |
| 3    | 0.9698            | 99.99          |
| 4    | 0.9769            | 99.97          |
| 5    | 0.9953            | 99.99          |
| 6    | 0.9833            | 95.01          |

Se volvieron a ejecutar los casos 4 y 6 con el tamaño de los *buffers* optimizados a fin de evitar retardos excesivos de encolamiento y por lo tanto retransmisiones. El criterio utilizado fue limitar el tiempo que transcurre desde la transmisión de un segmento hasta la recepción de su reconocimiento (ACK) a menos de 0.5 s. Sin embargo, aunque no ocurrieron retransmisiones, otras métricas como el índice de equidad y los valores de eficiencia fueron idénticos a los obtenidos con el criterio BDP.

### 6.3.4 Resultados para la configuración 2-nodos con tráfico de fondo y *buffers* calculados de acuerdo al criterio BDP

A ambos conmutadores se les asignó el mismo tamaño total de *buffer*, cuyo valor se muestra en la siguiente tabla. Ésta muestra además el tamaño de la cola observado en el conmutador 1. La ocupación máxima del *buffer* en el conmutador 2 fue de dos celdas.

Tabla 6.11 Algoritmo CDT: ocupación de los *buffers* calculados con el criterio BDP

| Caso | Buffer en el conmutador 1 [celdas] |                 |                  |
|------|------------------------------------|-----------------|------------------|
|      | Total                              | Ocupación media | Ocupación máxima |
| 7    | 4,000                              | 2,442.8         | 3,062            |
| 8    | 6,220                              | 4,283.4         | 5,420            |
| 9    | 15,210                             | 8,368.0         | 11,740           |

Al comparar los anteriores resultados contra los obtenidos para el algoritmo UT (tabla 5.6), se puede apreciar un incremento de miles de veces en la memoria que CDT utilizó; la figura 6.5 despliega gráficamente esta gran diferencia (observe que la escala vertical es logarítmica).

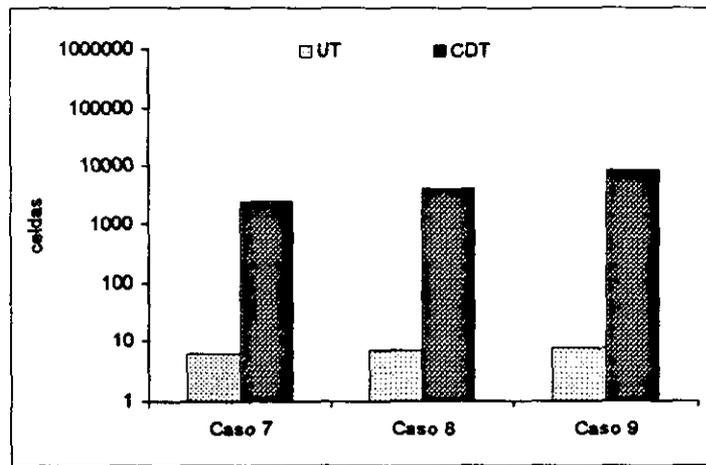


Figura 6.5 Comparación de la longitud media de la cola entre UT y CDT (criterio BDP)

La utilización del enlace fue prácticamente del 100% en todos los casos. Esto muestra que el tamaño de los *buffers* fueron suficientes para que CDT lograra una transmisión continua y en consecuencia altas tasas de utilización.

La tabla 6.12 muestra los valores obtenidos del índice de equidad y eficiencia. En estos aspectos el desempeño de CDT fue óptimo, aunque sólo marginalmente mejor que UT (vea la tabla 5.7).

Tabla 6.12 Algoritmo CDT: índice de equidad y eficiencia, criterio BDP

| Caso | Índice de equidad | Eficiencia [%] |
|------|-------------------|----------------|
| 7    | 0.9933            | 99.98          |
| 8    | 0.9930            | 99.98          |
| 9    | 0.9914            | 99.98          |

En resumen, cuando le asignamos a CDT un tamaño en los *buffers* comparable a el utilizado por UT, su rendimiento fue muy pobre en términos de utilización del enlace y en algunos casos ocurrieron una gran cantidad de retransmisiones de segmentos debido a tiempos de expiración en la capa TCP. Cuando se utilizaron *buffers* dimensionados de acuerdo al criterio del producto retardo por ancho de banda, el CDT presentó una utilización óptima del enlace y se comportó marginalmente mejor que UT en términos de eficiencia. Sin embargo, sus

requerimientos de memoria se incrementaron en algunos casos hasta decenas de miles de veces. Esto originó un gran retardo en la transferencia de celdas y retransmisiones de segmentos debido a tiempos de expiración en la capa TCP.

### 6.4 Simulaciones con la configuración *Max-Min Fairness*

La topología de red utilizada (figura 6.6) y los casos de estudio (tabla 6.13) fueron los mismos que los anteriormente utilizados para el algoritmo UT. Por claridad, ambos se reproducen en esta sección.

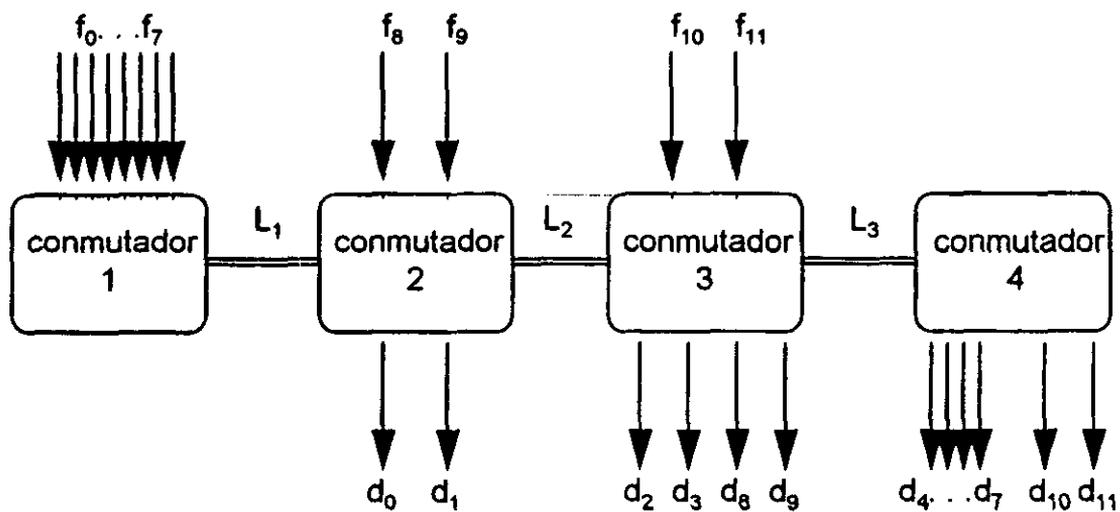


Figura 6.6 Configuración *Max-Min Fairness*

Los retardos de propagación de los enlaces entre los conmutadores ( $L_1, L_2$  y  $L_3$ ) fueron de 5ms. La conexión entre un punto extremo (fuente o destino) con el conmutador al cual se conecta se asumió con un retardo de 100 $\mu$ s. Los casos de estudio considerados se muestran a continuación.

Tabla 6.13 Casos de estudio para la configuración *Max-Min Fairness*

| Caso | Velocidad de transmisión* |     |        |
|------|---------------------------|-----|--------|
|      | L1                        | L2  | L3     |
| 1    | DS3                       | DS3 | DS3    |
| 2    | DS3                       | DS3 | 25Mbps |
| 3    | 25Mbps                    | DS3 | DS3    |

\*DS3=40.704Mbps

### 6.4.1 Resultados para la configuración *Max-Min Fairness* y *buffers* establecidos con el criterio "similar a UT"

Para este conjunto de simulaciones, el tamaño de los *buffers* en los conmutadores quedó establecido como se muestra a continuación.

**Tabla 6.14** Algoritmo CDT: tamaño de los *buffers* en los conmutadores, criterio "similar a UT"

| Caso | Buffer en cada conmutador [celdas] |       |       |     |
|------|------------------------------------|-------|-------|-----|
|      | 1                                  | 2     | 3     | 4   |
| 1    | 2,176                              | 760   | 1,278 | 192 |
| 2    | 1,848                              | 1,552 | 2,034 | 192 |
| 3    | 568                                | 776   | 564   | 192 |

La cantidad de memoria utilizada efectivamente por los conmutadores 1 al 3 se muestra en la tabla 6.15. Para el conmutador 4 el tamaño medio de la cola fue menor a uno.

**Tabla 6.15** Algoritmo CDT: cantidad total de memoria ocupada, criterio "similar a UT"

| Caso | Conmutador 1   |                 | Conmutador 2   |                 | Conmutador 3   |                 |
|------|----------------|-----------------|----------------|-----------------|----------------|-----------------|
|      | Media [Celdas] | Máximo [Celdas] | Media [Celdas] | Máximo [Celdas] | Media [Celdas] | Máximo [Celdas] |
| 1    | 1,729.9        | 2,176           | 194.3          | 760             | 340.6          | 1278            |
| 2    | 1,387.1        | 1,848           | 590.1          | 1,544           | 661.5          | 2034            |
| 3    | 437.6          | 560             | 196.6          | 704             | 161.2          | 564             |

Puede observarse que, aunque el tamaño total del *buffer* se estableció utilizando como referencia el tamaño máximo de la cola en las simulaciones del algoritmo UT, la ocupación media del *buffer* para CDT se mantuvo a un mayor nivel (vea la tabla 5.9). El tamaño insuficiente en el *buffer* de los conmutadores y el algoritmo basado en créditos originó esta acumulación de celdas.

La tabla 6.16 muestra la utilización del enlace de salida en los conmutadores para los tres casos de estudio. Estos valores resultaron en general más bajos que para las correspondientes simulaciones de UT, la figura 6.7 presenta una comparación gráfica entre los resultados obtenidos para ambos esquemas de control. Es claro de nuevo que el desempeño de CDT dista mucho de ser satisfactorio.

Tabla 6.16 Algoritmo CDT: utilización del enlace de salida, criterio "similar a UT"

| Caso | Utilización del enlace a la salida del conmutador |       |       |
|------|---|-------|-------|
|      | 1   | 2     | 3     |
| 1    | 0.330   | 0.353 | 0.170 |
| 2    | 0.638   | 0.533 | 0.257 |
| 3    | 0.531   | 0.358 | 0.165 |

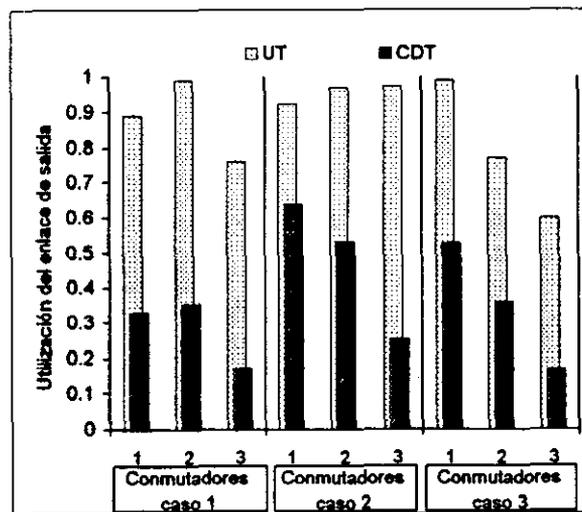


Figura 6.7 Comparación de la utilización enlace de salida entre UT y CDT (criterio "similar a UT")

La siguiente tabla muestra los valores obtenidos del índice de equidad y eficiencia.

**Tabla 6.17** Algoritmo CDT: eficiencia, criterio "similar a UT"

| Caso | Índice de equidad | Eficiencia [%] |
|------|-------------------|----------------|
| 1    | 0.8360            | 99.98          |
| 2    | 0.7297            | 99.98          |
| 3    | 0.7474            | 99.97          |

Los valores de eficiencia resultaron óptimos; pero los del índice de equidad hacen evidente una desventaja del algoritmo basado en créditos.

Cuando se utilizan *buffers* pequeños, como en este caso, la fuente transmite intermitentemente, ya que después de enviar todos los datos que puede alojar el receptor, tiene que detenerse a esperar por una notificación indicándole que hay espacio disponible. Conforme aumenta la lejanía de la fuente con respecto de su destino, se incrementa el tiempo que transcurre para la recepción de este mensaje. El efecto inmediato de esta situación es una dependencia de la utilización del enlace con respecto al retardo de propagación (esta observación se confirma por los resultados mostrados en la tabla 6.4). En consecuencia, los grupos de conexiones con fuentes cercanas a sus destinos se ven favorecidos con respecto a aquéllos con trayectorias de grandes retardos de propagación. Esto permite concluir que el algoritmo CDT con *buffers* pequeños resulta ser injusto.

#### **6.4.2 Resultados para la configuración *Max-Min Fairness* y *buffers* calculados de acuerdo al criterio BDP**

Para este conjunto de simulaciones, el tamaño utilizado en los *buffers* de los conmutadores se muestra a continuación.

**Tabla 6.18** Algoritmo CDT: tamaño de los buffers calculados con el criterio BDP

| Caso | Buffer en cada conmutador [celdas] |       |       |       |
|------|------------------------------------|-------|-------|-------|
|      | 1                                  | 2     | 3     | 4     |
| 1    | 2,216                              | 2,216 | 1,278 | 1,200 |
| 2    | 2,216                              | 2,216 | 2,004 | 1,200 |
| 3    | 3,504                              | 2,216 | 1,278 | 1,200 |

La siguiente tabla da las longitudes de la cola máxima y promedio alcanzadas en estas simulaciones para los conmutadores 1 a 3. En el conmutador 4 el tamaño medio de la cola fue menor a uno.

**Tabla 6.19** Algoritmo CDT: cantidad total de memoria ocupada, criterio BDP

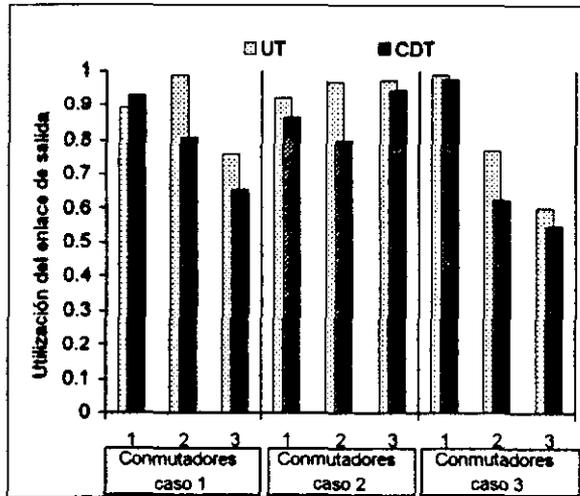
| Caso | Conmutador 1   |                 | Conmutador 2   |                 | Conmutador 3   |                 |
|------|----------------|-----------------|----------------|-----------------|----------------|-----------------|
|      | Media [Celdas] | Máximo [Celdas] | Media [Celdas] | Máximo [Celdas] | Media [Celdas] | Máximo [Celdas] |
| 1    | 1,323.3        | 2,208           | 311.0          | 2,216           | 132.5          | 1,242           |
| 2    | 1,427.2        | 2,216           | 595.4          | 2,200           | 512.9          | 1,980           |
| 3    | 2,376.3        | 3,504           | 194.0          | 2,216           | 144.1          | 1,278           |

Cuando estos resultados se comparan contra los de UT (ver tabla 5.9) se puede apreciar un enorme incremento hasta de cientos de veces en el tamaño medio del *buffer*.

Con esta configuración de *buffers* optimizados, los valores obtenidos en la utilización del enlace de salida de los conmutadores se muestran en la tabla 6.20. Comparando contra los valores de la tabla 5.10 se puede ver que la métrica de utilización es mejor para UT en todos los casos y para todos los conmutadores, excepto en el caso 1 y en el conmutador 1. La figura 6.8 muestra gráficamente esta diferencia.

**Tabla 6.20** Algoritmo CDT: utilización del enlace de salida, criterio BDP

| Caso | Utilización del enlace a la salida del conmutador |       |       |
|------|---|-------|-------|
|      | 1   | 2     | 3     |
| 1    | 0.934   | 0.808 | 0.656 |
| 2    | 0.865   | 0.801 | 0.946 |
| 3    | 0.979   | 0.626 | 0.550 |



**Figura 6.8** Comparación de la utilización del enlace de salida entre UT y CDT (criterio BDP)

La siguiente tabla muestra los valores obtenidos del índice de equidad y eficiencia.

**Tabla 6.21** Algoritmo CDT: eficiencia, criterio BDP

| Caso | Índice de equidad | Eficiencia [%] |
|------|-------------------|----------------|
| 1    | 0.9912            | 99.99          |
| 2    | 0.9536            | 99.99          |
| 3    | 0.9612            | 99.99          |

Como se puede ver la eficiencia resultó óptima. Los valores del índice de equidad resultaron ser menores en todos los casos a los correspondientes obtenidos en las simulaciones de UT (vea la tabla 5.11).

# Conclusiones y comentarios

La evaluación del desempeño de algoritmos de control de tráfico carece de un método completamente general para realizarse. Es así que los primeros capítulos de este trabajo reflejan la preocupación por definir una metodología, adecuada al caso particular, que fuera objetiva, racional y sistemática. Considero que en este proceso las siguientes observaciones fueron importantes:

- El conjunto más adecuado de métricas para cuantificar el desempeño de un sistema intenta describirlo completamente, forma un grupo de elementos no redundantes y de baja variabilidad.
- Generalmente los modelos de tráfico actualmente conocidos no pueden satisfacer simultáneamente todos los criterios de selección. En consecuencia, el modelo apropiado depende del nivel de detalle deseado, se puede seleccionar un modelo sencillo; pero éste debe estar acorde con la realidad que pretenda representar.
- Para simular un ambiente de múltiples nodos en diferentes localizaciones físicas se puede emplear la configuración 2-nodos con diferentes retardos de propagación y fuentes limitadas. Sin embargo, para evitar omisiones y una muy meticulosa selección de parámetros, se puede utilizar una topología más compleja (como la *Max-Min Fairness*) que permita exponer naturalmente los algoritmos de control a condiciones heterogéneas.

La metodología de evaluación consistió básicamente en la simulación por computadora utilizando modelos de tráfico realistas y el posterior análisis de los resultados obtenidos. De esta forma, investigué el desempeño de dos técnicas diferentes para el control del tráfico ocasionado por el servicio ABR en redes ATM. Se seleccionaron los algoritmos *Uniform Tracking (UT)* y *Controlled Dynamic Transfer (CDT)* como representantes de las aproximaciones de control basadas en tasa explícita y en créditos respectivamente.

El análisis de las estadísticas obtenidas muestran que UT transfiere eficientemente el tráfico de aplicaciones con requerimientos no estrictos de calidad de servicio. Utiliza poca memoria en los *buffers* de los conmutadores y logra pequeños retardos de transferencia de celdas y altas tasas de utilización. También se confirmó que UT escala bien con diferentes retardos de propagación y número de conexiones. Como trabajo futuro, queda por investigar la mejora en desempeño bajo diversas técnicas de descarte selectivo de celdas (como *Early Packet Discard* [Romanow 1994]).

Por otro lado, el desempeño del algoritmo CDT resulta muy sensible al tamaño de la memoria disponible en el conmutador. Una apropiada selección de ésta depende sobre todo de un conocimiento preciso de los retardos de propagación y del número de usuarios que se pretende atender.

De acuerdo con los resultados de este estudio, una cantidad pequeña de memoria, como la requerida por UT, no es suficiente para que CDT logre un desempeño satisfactorio. Cuando CDT utiliza *buffers* optimizados, su desempeño es marginalmente mejor al logrado por UT en términos de rendimiento, eficiencia y utilización de los enlaces en los conmutadores; pero el retardo medio de transferencia de las celdas aumenta considerablemente ya que cada elemento en la red tiende a retener tantos datos como pueda almacenar.

De las anteriores observaciones se puede concluir que a menos de que se requiera garantizar un porcentaje nulo de pérdida de celdas (lo cual no es comúnmente el caso en las aplicaciones de transferencia de datos), no es recomendable utilizar aproximaciones de control basadas en créditos para proporcionar el servicio ABR.

Con los resultados de esta investigación se escribió la contribución canadiense a la ITU-T [CAN 1997] y el artículo [López *et al.* 1998]. Algunos de los resultados obtenidos se publicarán también en [Gallardo *et al.* 1998].

## Apéndice A

### Estadísticas en el simulador

En esta sección se listan las estadísticas que se implantaron en el simulador. Aún cuando no todas ellas son estrictamente necesarias para realizar el análisis del desempeño de los algoritmos de control, fue necesario tomarlas en consideración para verificar el correcto funcionamiento del código del programa y el adecuado comportamiento de los componentes del sistema. Asimismo, en base a sus valores en resultados preliminares, fue necesario reajustar los parámetros para las simulaciones finales.

El modelo de simulación se implantó en lenguaje C++ utilizando el compilador g++ versión 2.7.2.2<sup>1</sup>. Las facilidades de simulación se proporcionan al programa a través de la librería SimS 2.0.0<sup>2</sup>.

En todos los casos las mediciones se obtienen para ambos algoritmos de control a menos de que explícitamente se especifique lo contrario.

Nivel de la aplicación (mediciones en cada fuente):

- Número de archivos generados
- Tamaño de los archivos producidos (valor mínimo, medio, máximo y varianza)
- Tiempo de espera entre el fin de la transmisión de un archivo y la generación del siguiente (valor mínimo, medio, máximo y varianza)
- Tasa de transmisión lograda (valor mínimo, medio, máximo y varianza)
- Tiempo de transferencia de los archivos (valor mínimo, medio, máximo y varianza)

---

<sup>1</sup> El compilador g++ es *software* sin costo producido por la compañía *Free Software Foundation*.

<sup>2</sup> SimS es una librería orientada a la *simulación por procesos* para sistemas en tiempo continuo, discretos o híbridos [Jade]. Es producida por *Jade Simulations International Corporation*.

Capa TCP (mediciones en cada fuente a menos de que se indique lo contrario):

- Número de segmentos enviados
- Número de segmentos retransmitidos
- Número de segmentos retransmitidos debido a tiempos de expiración
- Número de segmentos retransmitidos debido a reconocimientos repetidos (global)
- Histograma de los valores de la ventana de congestión de TCP
- Tiempo de vuelta completa para un segmento TCP (valor mínimo, medio, máximo y varianza)

Adaptador ABR (mediciones en cada fuente a menos de que se indique lo contrario)

- Número total de celdas enviadas (datos y control)
- Número de celdas de control enviadas
- Número de paquetes TCP descartados
- Número de incrementos en ACR (sólo para el algoritmo de tasa explícita)
- Número de decrementos en ACR (sólo para el algoritmo de tasa explícita)
- Número de decrementos en ACR debidos al comportamiento 6 de la fuente (sólo para el algoritmo de tasa explícita)
- Número de decrementos en ACR debidos al comportamiento 5 de la fuente (global y sólo para el algoritmo de tasa explícita)
- ACR asignado a la fuente (valor mínimo, medio, máximo y varianza; sólo para el algoritmo de tasa explícita)
- Tiempo de vuelta completa para las celdas RM (valor mínimo, medio, máximo y varianza; sólo para el algoritmo de tasa explícita)
- Nivel de ocupación del *buffer* en el adaptador (valor mínimo, medio, máximo y varianza)
- Tiempo que le toma a la fuente incrementar su tasa de transmisión hasta alcanzar su valor final (valor mínimo, medio, máximo y varianza; sólo para el algoritmo de tasa explícita)
- Utilización del enlace de salida de la fuente (valor mínimo, medio, máximo y varianza)
- Traza de los valores ACR (sólo para el algoritmo de tasa explícita)

**Conmutador (mediciones en cada conmutador):**

- Histograma del nivel de ocupación de los *buffers* del conmutador (para cada clase de servicio y en las direcciones hacia adelante y hacia atrás)
- Tamaño máximo y medio de la cola (para cada clase de servicio y en las direcciones hacia adelante y hacia atrás)
- Número de celdas conmutadas (en las direcciones hacia adelante y hacia atrás)
- Número de celdas perdidas
- Utilización del enlace de salida (en las direcciones hacia adelante y hacia atrás)
- Retardo de transferencia de la celda (incluye encolamiento y servicio, en las direcciones hacia adelante y hacia atrás)

**Receptor ABR (mediciones en cada receptor):**

- Número de bits recibidos
- Número de bits descartados

**Capa TCP en el receptor (mediciones en cada receptor):**

- Número de segmentos recibidos
- Número de segmentos descartados

**Métricas de rendimiento global de la red:**

- Eficiencia (sólo configuración 2-nodos)
- Índice de equidad (calculado al nivel de la capa TCP)

**Fuente MPEG**

- Número de celdas enviadas

**Receptor MPEG**

- Número de celdas recibidas

# Referencias

- [Adas 1997] Abdelnaser Adas, "Traffic Models in Broadband Networks", IEEE Communications Magazine, pp. 82-89, julio de 1997.
- [Arulambalam et al. 1996] Ambalavanar Arulambalam y Xiaoqiang Chen, "Allocating Fair Rates for Available Bit Rate Service in ATM Networks", IEEE Communications Magazine, pp. 92-100, noviembre de 1996.
- [ATM Forum 1996] ATM Forum, "Traffic Management Specification Version 4.0", mayo de 1996.
- [Bonomi et al. 1995] Flavio Bonomi y Kerry W. Fendick, "The Rate Based Flow Control for the Available Bit Rate ATM Service", IEEE Network, pp. 25-39, marzo-abril de 1995.
- [Campbell 1994] Joe Campbell, *C Programmer's Guide to Serial Communications*, segunda edición, SAMS Publishing, USA 1994.
- [CAN 1997] "Support of Bursty Data Applications with ABR", contribución CAN COM 13-31 para la ITU-T, Toronto, septiembre de 1997.
- [Chen et al. 1996] Thomas M. Chen, Steve S. Liu y Vijay K. Samalam, "The Available Bit Rate Service for Data in ATM Networks", IEEE Communications Magazine, pp. 56-71, mayo de 1996.
- [Çinlar 1975] Erhan Çinlar, *Introduction to Stochastic Processes*, Prentice Hall, USA 1975.
- [Danzig et al. 1991] Peter B. Danzig y Sugih Jamin, "tcplib: A Library of TCP Internetwork Traffic Characteristics", Reporte técnico USC-CS-91-495, University of Southern California, Los Angeles, septiembre de 1991.

- [Danzig *et al.* 1992] Peter B. Danzig, Sugih Jamin, Ramón Cáceres, Danny J. Mitzel y Deborah Estrin, "An Empirical Model for Driving Wide-Area TCP/IP Network Simulations", *Internetworking: Res., Experience*, pp. 1-26, marzo de 1992.
- [Erramilli *et al.* 1994] Ashok Erramilli, James Gordon y Walter Willinger, "Applications of Fractals in Engineering for Realistic Traffic Processes", *The Fundamental Role of Teletraffic in the Evolution of Telecommunications Networks*, ITC 14, pp. 35-43, Antibes Juan-les-Pins, Francia, 6 al 10 de junio de 1994, editores: J. Labetoulle y J. W. Roberts, Elsevier Science B. V.
- [Evans *et al.* 1993] Merran Evans, Nicholas Hastings y Brian Peacock, *Statistical Distributions*, John Wiley & Sons, USA 1993.
- [Fendick 1996] Kerry W. Fendick, "Evolution of Controls for the Available Bit Rate Service", *IEEE Communications Magazine*, pp. 35-39, noviembre de 1996.
- [Forys *et al.* 1995] Lenord J. Forys, Ashok Erramilli y Jonathan L. Wang, "New Traffic Analysis and Engineering Methods for Emerging Technologies", *IEEE GLOBECOM '95*, pp. 848-854, Singapur, 1995.
- [Fowler *et al.* 1991] Henry J. Fowler y Will E. Leland, "Local Area Network Traffic Characteristics, with Implications for Broadband Network Congestion Management", *IEEE Journal on Selected Areas in Communications*, vol. 9, núm. 7, pp. 1139-1149, septiembre de 1991.
- [Frost *et al.* 1994] Victor S. Frost y Benjamin Melamed, "Traffic Modeling for Telecommunications Networks", *IEEE Communications Magazine*, pp. 70-81, marzo de 1994.
- [Fulton *et al.* 1997] Cathy Fulton, San-Qi Li y Chang Soo Lim, "UT: ABR Feedback Control with Tracking", *Infocom'97*, Kobe Japón, pp. 806-815, abril 7 al 11 de 1997.
- [Gall 1991] Didier Le Gall, "MPEG: A Video Compression Standar for Multimedia Applications", *Communications of the ACM*, vol. 34, núm. 4, pp. 46-58, abril de 1991.

- [Gallardo *et al.* 1998] José R. Gallardo, Dimitris Makrakis, Miguel López Guerrero, Luis Orozco Barbosa y Steve Rosenberg, "Performance Comparison of UT and CT under Realistic Traffic Models", a presentarse en ICT'98, Grecia, junio de 1998.
- [Gusella 1990] Riccardo Gusella, "A Measurement Study of Diskless Workstation Traffic on an Ethernet", *IEEE Transactions on Communications*, vol. 38, núm 9, pp. 1557-1568, septiembre de 1990.
- [Jade] Jade Simulations International Corporation, "SimS, A C++ Library for Process-Oriented Simulation", Interface Specification, Release 2.0.0, [s. a.].
- [Jain *et al.* 1986] Raj Jain y Shawn A. Routhier, "Packet Trains, Measurements and a New Model for Computer Network Traffic", *IEEE Journal in Selected Areas in Communications*, vol. SAC-4, núm. 6, pp. 986-995, septiembre de 1996.
- [Jain 1991] Raj Jain, *The Art of Computer Systems Performance Analysis*, John Wiley & Sons, USA 1991.
- [Jain *et al.* 1996] Raj Jain, Shivkumar Kalyanaraman, Sonia Fahmy, Rohit Goyal y Seong-Cheol Kim, "Source Behaviour for ATM ABR Traffic Management: An Explanation", contribución ATM Forum/96-1270, octubre de 1996.
- [Kleinrock 1975] Leonard Kleinrock, *Queueing Systems, Vol. I: Theory*, John Wiley & Sons, USA 1975.
- [Kleinrock 1979] Leonard Kleinrock, "Power and Deterministic Rules of Thumb for Probabilistic Problems in Computer Communications", ICC'79, pp. 43.1.1-43.1.10., Boston MA, junio 10 al 14 de 1979.
- [Kleinrock 1993] Leonard Kleinrock, "On the Modeling and Analysis of Computer Networks", *Proceedings of the IEEE*, vol. 81, núm. 8, pp. 1179-1189, agosto de 1993.
- [Kung *et al.* 1995] H. T. Kung y Robert Morris, "Credit-Based Flow Control for ATM Networks", *IEEE Network*, pp. 40-48, marzo-abril de 1995.

- [Leland *et al.* 1994] Will E. Leland, Murad Taqqu, Walter Willinger y Daniel V. Wilson, "On the Self Similar Nature of Ethernet Traffic (Extended Version)", *IEEE/ACM Transactions on Networking*, vol. 2, núm. 1, pp. 1-15, febrero de 1994.
- [López *et al.* 1998] Miguel López Guerrero, José R. Gallardo, Steve Rosenberg, Luis Orozco Barbosa y Dimitris Makrakis, "Study of a Rate-Based ABR Scheme Supporting Bursty Data Applications", a presentarse en *IEEE ISCC'98*, Grecia, julio de 1998.
- [Papoulis 1991] Athanasios Papoulis, *Probability, Random Variables and Stochastic Processes*, McGraw Hill, Singapur 1991.
- [Paxson *et al.* 1995] Vern Paxson y Sally Floyd, "Wide Area Traffic: The Failure of Poisson Modeling", *IEEE/ACM Transactions on Networking*, vol. 3, núm. 3, pp. 226-244, junio de 1995.
- [Peterson *et al.* 1996] Larry L. Peterson y Bruce S. Davie, *Computer Networks, a Systems Approach*, Morgan Kaufman Publishers Inc., California USA, 1996.
- [QFC 1997] Quantum Flow Control, General Release versión 2.0.5, marzo de 1997.
- [Romanow *et al.* 1994] Allyn Romanow y Sally Floyd, "Dynamics of TCP Traffic over ATM Networks", *ACM Sigcomm'94*, pp. 79-88, Londres UK.
- [Saito *et al.* 1996] Hiroshi Saito, Konosuke Kawashima, Hideo Kitazume, Arata Koike, Mika Ishizuka y Atsushi Abe, "Performance Issues in Public ABR Service", *IEEE Communications Magazine*, pp. 40-48, noviembre de 1996.
- [Schwartz 1994] Mischa Schwartz, *Redes de Telecomunicaciones*, Addison-Wesley Iberoamericana, USA 1994.
- [Schwartz 1996] Mischa Schwartz, *Broadband Integrated Networks*, Prentice Hall, USA 1996.
- [Stallings 1995] W. Stallings, *ISDN and Broadband ISDN with Frame Relay and ATM*, Prentice Hall, USA 1995.

- [Stamoulis *et al.* 1994] G. D. Stamoulis, M. E. Anagnostou y A. D. Georgantas, "Traffic Source Modeling for ATM Networks: a Survey", *Computer Communications*, vol. 17, núm. 6, pp. 428-438, junio de 1994.
- [Stevens 1990] W. Richard Stevens, *UNIX Network Programming*, Prentice Hall, USA 1990.
- [Stevenson 1993] Daniel Stevenson, "Electropolitical Correctness and High-Speed Networking or Why ATM is Like a Nose", *Asynchronous Transfer Mode Networks*, editado por Y. Viniotis y R. O. Onvural, Plenum Press, New York, 1993.
- [Willinger *et al.* 1997] Walter Willinger, Murad S. Taqqu, Robert Sherman y Daniel V. Wilson, "Self-Similarity Through High-Variability: Statistical Analysis of Ethernet LAN Traffic at the Source Level", *IEEE/ACM Transactions on Networking*, vol. 5, núm. 1, pp. 71-86, febrero de 1997.
- [Yin *et al.* 1993] Nanying Yin y Michael G. Hluchyj, "Simple Models for Statistically Multiplexed Data Traffic in Cell Relay Networks", *Globecom'93*, pp. 824-829, Houston TX, noviembre 28 a diciembre 3, 1993.

# Glosario

**ACK** Abreviatura de *acknowledgment* (acuse de recibo). Es la señal de un protocolo de comunicaciones que le anuncia al transmisor la recepción de un mensaje.

**AppleTalk** Serie de protocolos de comunicaciones desarrollados por la compañía *Apple Computer*.

**ATM Forum** Es una organización voluntaria e internacional compuesta de fabricantes, vendedores, proveedores de servicios, organismos de investigación y usuarios de ATM. Su propósito es acelerar el desarrollo de productos y servicios ATM a través de la rápida definición de las especificaciones de conectividad, la promoción de la cooperación industrial y algunas otras actividades.

**B-ISDN** *Broadband ISDN*: ISDN de banda ancha. Estándares de comunicaciones que se desarrollan para soportar servicios de banda ancha.

**canal D** Canal de la ISDN utilizado para señalización, es *full duplex* de 16 kbps (tasa básica) o de 64 kbps (tasa primaria).

**conmutación de circuitos** Tecnología de red que dedica un circuito a la vez para la transmisión de datos.

**conmutación de paquetes** Tecnología de red en la cual los nodos comparten el ancho de banda disponible porque envían paquetes en forma intermitente.

**CCITT** Comité Consultivo Internacional de Telefonía y Telegrafía (siglas en francés). Organización internacional para la definición de estándares de comunicaciones. Actualmente llamada ITU-TSS.

**frame relay** Una tecnología de red basada en circuitos virtuales que soporta la transmisión de tramas de longitud variable.

**FTP** *File Transfer Protocol*: Protocolo de transferencia de archivos.

**ISDN** *Integrated Services Digital Network*: Red digital de servicios integrados. Protocolos de comunicaciones propuestos por las compañías telefónicas para lograr que sus redes transmitan otros tipos de datos además de las convencionales conversaciones de voz.

**ISO** *International Standards Organization*. Un organismo de estandarización internacional y voluntario.

**ITU-TSS** *International Telecommunications Union - Telecommunications Standards Sector*: Es el nuevo nombre para la CCITT. Un organismo internacional de estandarización.

**LAN** *Local Area Network*: Red de área local. Red que cubre un área geográfica relativamente pequeña (usualmente no mayor que un grupo de edificios cercanos). Suele caracterizarse por velocidades de transferencia de datos relativamente altas y una baja incidencia de errores.

**NetWare** Desarrollado y distribuido por Novell, Inc., es un sistema de archivos muy popular. Ofrece acceso transparente a archivos remotos y muchos otros servicios distribuidos de redes.

**OSI Reference model** Modelo de referencia OSI. Modelo de arquitectura de redes desarrollado por ISO. Consiste en siete capas, cada una de las cuales especifica funciones particulares de la red, tales como direccionamiento, control de flujo, control de errores, encapsulamiento, transferencia confiable de mensajes y muchas otras. La capa más alta (*application layer*: capa de aplicación) es la más cercana al usuario. La capa más baja (*physical layer*: capa física) es la más cercana a la tecnología del medio físico.

**PCM** *Pulse Code Modulation*: Modulación por código de pulsos. Transmisión de información analógica en forma digital mediante muestreo y codificación con un número fijo de bits.

**SMDS** *Switched Multimegabit Data Service*: Servicio de datos conmutados multimegabit. Tecnología WAN basada en datagramas que emplea conmutación de paquetes a alta velocidad.

**TCP** *Transmission Control Protocol*: Protocolo de control de transmisiones. Protocolo correspondiente a la capa 4 del modelo de referencia OSI (capa de transporte); ofrece transmisión confiable de datos.

**X.25** Estándar del ITU-TSS para conmutación de paquetes de longitud variable en redes públicas de datos. Especifica los protocolos y funciones de la interfaz entre los usuarios y la red en las capas 1, 2 y 3 del modelo de referencia OSI.