UNIVERSITAT POLITÈCNICA DE CATALUNYA

DEPARTAMENT D'ARQUITECTURA DE COMPUTADORS

ANALISIS Y EVALUACION DE LOS SISTEMAS DE PROTECCION CONTRA LA CONGESTION EN LA RED DIGITAL DE SERVICIOS INTEGRADOS DE BANDA ANCHA

TESIS DOCTORAL

Presentada en la Universitat Politècnica de Catalunya para
la obtención del título de Doctor Ingeniero de Telecomunicación

Autor: Germán Santos Boada

Director: Jordi Domingo Pascual

BARCELONA, Noviembre de 1992

Conclusiones Parte E.

En esta parte se ha analizado la necesidad de la existencia de funciones de control de la congestión en redes ATM, estudiando las tendencias actuales en ese camino y viendo las dificultades surgidas en ello.

Se propone un modelo de red que permita realizar la gestión de la misma de una forma eficaz y estandarizada, incluyendo en dicho modelo las ubicacione correspondientes a las funciones de control de la congestión.

Se hace un estudio detallado de las funciones de control de la congestión, analizando y evaluando el estado del arte sobre la cuestión.

Prioridades, control de admisiones y funciones de policía son el objeto del estudio, incidiendo de una forma especial en estas últimas.

Se analiza el comportamiento de las funciones de policia de forma comparada a partir de trabajos concretos de simulación propuestos en la bibliografía.

Las conclusiones mas importantes que se obtienen de esta parte son las siguientes:

1. Las funciones de prioridades deben reducirse al mínimo para evitar grandes complejidades en su tratamiento (en la propuesta de distribución de bits en la celda ATM se reserva un sólo bit para prioridades).

2. Las funciones propuestas de control de admisiones son muy sencillas en general y poco eficientes.

3. Las funciones de policía propuestas tienen diferentes campos de aplicación, siendo el Leaky Bucket la que parece como mas firma candidata por su simplicidad de uso e implementación y su eficiencia en la mayoría de los casos.

4. Las funciones de policia tratadas y propuestas actúan por circuito virtual de forma independiente, sin tener en cuenta las condiciones generales de carga y de situación de la red, lo que unido a que no tienen en cuenta el efecto Jitter, las aleja de la realidad. Este último punto se tratará con mas detalle en la parte G.

Parte F. Nuevas tendencias en las funciones de policia.

Todas las funciones de policia que hemos estudiado hasta ahora tienen en común que son estáticas en el sentido de que no modifican los parámetros que las definen dependiendo de la situación de la red.

Por ello las nuevas tendencias en el desarrollo de funciones de policía van encaminadas a buscar soluciones dinámicas que tengan en cuenta la carga general de la red o bien que actuen sobre la fuente con realimentación en caso de detectar la necesidad de ello [DEC90].

En este caso se rompe con la tendencia habitual hasta el momento de separar las actuaciones de las funciones de control de admisiones y la función de policia. Los métodos basados en la carga relacionan ambas funciones. Esta es una propuesta realizada por [SANb91] y que está adelantada por [RACa90] como la alternativa mas viable en un futuro dentro del control de congestión.

Se identifican dos funciones distintas en el control del tráfico en estas condiciones: Acondicionamiento del ancho de banda y Policía de fuente [SAT90]. Exactamente igual que en las funciones estáticas de policia la función de Policia de Fuente debe regular el flujo de celdas en la red. La diferencia está en que la esta función actuará sobre la fuente en origen.

El problema que se introducirá con estos métodos es el de la señalización adicional que será necesario añadir para cumplir los objetivos de las funciones de control de congestión, que a su vez puede introducir mas congestión.

En el fondo estos métodos son ya sobradamente conocidos en otros entornos mas tradicionales, y supone una aplicación concreta de éstos a situaciones mas críticas en redes ATM.

F.1 Acondiiconamiento del ancho de banda

Las redes de banda ancha requerirán sistemas de control de la congestión tolerantes y flexibles, con algoritmos integrantes de los elementos de conmutación o nodos [LAZ91] [DIT91].

F1.1 Función de policía dependiente de la carga de red.

Método basado en la utilización dinámica del ancho de banda disponible, en el cual éste es concedido a las fuentes según sus necesidades. Para ello se utilizan autorizaciones "tickets" de

paso que se entregan a las fuentes conforme lo demanden, y la carga lo permita.

Cada autorización es negociada y concedida por la central local de la que depende esa fuente, y permite la introducción en la red de un cierto número de celdas. Asociado con el enlace está la máxima capacidad de autorizaciones que será determinada por la máxima capacidad del enlace. En la fase de conexión, la fuente recibe un mínimo de autorizaciones, que coincide con el mínimo necesario por la fuente para trabajar. Este mínimo debe estar garantizado durante toda la conexión. Si la fuente necesita enviar mas ritmo de datos debe solicitar mas autorizaciones, las cuales no están en ningún caso garantizadas.

Esto implica la necesidad de una nueva función en el contexto del control de la congestión que será la de control de autorizaciones según se indica en la figura.

Los parámetros necesarios para controlar esta función serán:

. Minimo número de autorizaciones por conexión.
. Número de autorizaciones en un periodo determinado.
. Duración de ese periodo.

En este procedimiento la tarificación será proporcional al numero de autorizaciones concedidas.

El gran problema que surge con esta solución es el conocimiento por parte de la red del número de autorizaciones que puede conceder sin que exista congestión. Sigue sin resolverse la cuestión.

F1.2 Función de policía de nivel de ráfaga.

Este método está basado en la utilización de un sistema tradicional en el control de la congestión como es el de la "Reserva de Recursos". Se fundamenta en el modelo de tres niveles de tráfico explicado en la parte C de esta tesis y mas concretamente en el estudio de la viabilidad de propagación sin congestión por la red de una ráfaga de celdas.

Ya anteriormente propuesto [HUI88], por el momento el único método desarrollado en detalle es el de FRP/DT propuesto por [BOY91] y que se explica resumido en este apartado.

El método no es de aplicación general sino que se basa en fuentes denominadas "stepwise variable bit rate". Esto quiere decir que son fuentes en las que el denominado nivel de ráfagas puede ser controlado, y en el momento que las rafagas se producen se puede esperar en su transmisión a que se les de la autorización para entrar en la red ATM.

Tráfico de este estilo es por ejemplo el de interconexión

LAN-LAN, que es en pricipio el primero que se espera soporte la IBCN. Este tráfico puede ser multiplexado estadisticamente sin una degradación significativa del GOS y del QOS ofrecido por la red por el hecho de introducirle un control de acceso a nivel de ráfaga.

Durante cada ráfaga, la fuente requiere una velocidad constante negociando con la red los recursos necesarios al principio de la transmisión. Si se le concede el ancho de banda solicitado este se mantendrá durante la transmisión de la ráfaga.

Se definen dos esquemas de multiplexación para el método propuesto:

FRP/DT : "Fast Reservation Protocol with Delayed Transmission"

FRP/IT : "Fast Reservation Protocol with Inmediate Transmission"

Ambos métodos utilizan el mismo Hardware básico en los elementos de conmutación y pueden coexistir en una red general. Como se puede observar en la definición, la diferencia entre ambos métodos consiste en la técnica de elección del momento de transmisión. En el FRP/DT no se transmite hasta que no se tiene los recursos reservados en toda la ruta de la conexión. En el FRP/IT esta reserva se hace nodo a nodo y se transmite cuando se tiene el recurso reservado con el siguiente nodo, desconociéndose la existencia de recursos disponibles en el recorrido hasta que se llega a ellos.

F1.2.1 Especificaciones del FRP/DT

De acuerdo con el protocolo, las fuentes a las que se puede aplicar el FRP/DT pueden esperar a disponer de los recursos necesarios el tiempo necesario para conseguirlos. En este caso las fuentes se denominarán "Negotiated Stewise VBR Sources (NSVBR)". Por razones de seguridad, los mensajes de reserva y de desocupación de recursos no serán procesados por los nodos sino por una unidad especial que deberá estar en el nodo de entrada que se denominará "FRP Control Unit (FRPCU)". Por lo tanto el protocolo FRP/DT se compondrá del procedimiento de usuario y del procedimiento interno de red. El proceso de señalización se hará dentro de banda y utilizando los recursos habituales en la transmisión de la información.

A) El procedimiento de usuario será como se indica a continuación, considerando una comunicación LAN to LAN:

1. La LAN1 evalua el tráfico CBR hacia la LAN2 y envia RRC (celda de solicitud de reserva) indicando el valor del ancho de banda que necesita (Vcbr). FRPCU contesta con RRCB (celda de reserva recibida) hacia la LAN1 intentando de reservar Vcbr a lo largo del Paso Virtual. Si lo consigue, FRPCU envia RAC (Celda de reserva aceptada) hacia la LAN1 adaptando los parámetros de la

función de policia correctamente al valor Vcbr aceptado. Si la reserva no se puede realizar FRPCU enviará RDC (celda de reserva denegada) hacia la LAN1.

2.Cada vez que la LAN1 quiera enviar unj fichero hacia la LAN2, la LAN1 deberá enviar RRC con el nuevo valor absoluto de bit rate Vcbr + Vráfaga. FRPCU enviará RRCB y tratará de conseguir los recursos necesarios para Vráfaga. Si se consigue FRPUC enviará RAC al LAN1 indicando los parámetros para la función de policía correspondientes a Vcbr + Vráfaga. En ese momento puede comenzar la transmisión de la ráfaga. Si la reserva no se puede realizar después de algunos intentos, FRPCU enviará RDC hacia la LAN1, y se mantendrá la transmisión en las condiciones anteriores.

3. Después de transmitir una ráfaga, se debe volver a la situación original de CBR mediante la celda RERC (liberación de recursos).

B) El procedimiento interno de red permite reservar los diferentes anchos de banda en todos los recursos de la red. Los parámetros que utilizará serán los siguientes:

Ωmax : Maximo bit rate requerido por la conexión
Ω : Bit rate reservado actual
Ωreq : Nuevo valor de bit rate requerido
T1 : Valor de Time-Out
BB : Dos Bits en el campo de control

Veamos como se produciria la actuación del protocolo en el caso de la conexión LAN to Lan indicada anteriormente:

Cuando la FRPCU recibe el RRC la actuación que se seguirá será:

1. Enviará a lo largo de todo el Paso virtual RRC indicando el Ωreq y el temporizador T1. Después de la recepción cada nodo seleccionará el ancho de banda requerido en el puerto de salida correspondiente. Si la reserva es posible se actualizará T1 y enviará RRC al siguiente nodo, sino será descartado. T1 informará al final del proceso del retardo esperado por reserva en la conexión.

2. Si el nodo de destino puede soportar la reserva enviará hacia atrás RAC siguiendo el camino inverso hasta el nodo de entrada.

3. Después de la recepción de RAC, la FRPCU enviará una celda de validación VC por todo el recorrido para confirmar la reserva, actualizando a su vez los parámetros de la función de policia de todo el recorrido al nuevo Ωreq.

4. El proceso dispondrá de un temporizador de validación el cual prodrá agotarse sin recibir la confirmación final del proceso, suponiendo ello que las condiciones d ela red en ese momento no

permitian la reserva, pudiéndose repetir el proceso.

El problema que se plantea en un método como este, es como siempre el determinar cuando en un Nodo es posible o no conceder la capacidad de recursos que se requiere, sin afectar al QOS de las otras conexiones que utilizan los mismos recursos. El método propuesto por [BOY91] es el denominado "Peak Allocation" o bit rate de pico, ya explicado en el apartado de Control de Admisiones.

F1.2.2 Especificaciones del FRP/IT

Este método se encuentra aún en fase de especificación [BOY91] , y sirve como complemento al anterior para aquellas fuentes que no pueden esperarse a recibir la autorización de reserva de recursos en todo el paso virtual. Tiene aplicación en fuentes con tráfico de ráfagas de pequeña duración comparado con el tiempo total de espera a la concesión de recursos en FRP/DT, y aplicaciones que no toleran una negativa por parte de la red en el acceso, como por ejemplo voz, correo rápido, codec en tiempo real etc.

Estas fuentes se denominarán "Negotiated Burstwise Variable Bit Rate Sources (NBVBR)".

De acuerdo con el protocolo, cuando una fuente tiene una ráfaga que enviar, se solicita reserva de recursos al nodo adiacente enviando RRC e indicando el bit rate necesario. Este le contestará afirmativamente o negativamente procediéndose o no a la transmisión. En caso de respuesta negativa se perderá la ráfaga. Este proceso se repetirá nodo a nodo.

Evaluación del FRP/DT

Desde un punto de vista funcional, el FRP/DT será una parte de la señalización de la IBCN [BOY91].

El principo de reserva puede estudiarse desde dos puntos de vista opuestos. En primer lugar introduce una predicción sobre la carga multiplexada lo cual permite un eficiente control del acceso de la ráfaga, y en segundo lugar el sistema es responsable de la ineficiencia producida debido al "Round trip delay" o retardo debido a la espera de la consulta de reserva de recursos. Este bloqueo que se produce durande el "Round trip delay" es el resumen del comportamiento del FRP/DT.

Si consideramos EA como la duración media del periodo activo de una fuente y ERT como el "Round trip delay" maáximo, y consideramos que una ráfaga no puede comenzr a transmistirse hasta que la reserva está hecha, la carga σ reservada por la red difiere de la carga ofrwecida σs en:

$$\sigma = \sigma s \left(1 + \frac{ERT}{EA} \right)$$

La ineficiencia de la transmisión quedará reflejada por la relación ERT/EA. Un valor de 10 % o 15 % será aceptado generalmente como máximo. Si consideramos por ejemplo la transmisión de una imagen de 1 Mbit y debe ser displayada con EA = 500 ms en la pantalla en España con ERT = 25 ms el valor de la ineficiencia será del orden del 5 %. En comunicaciones costa a costa como en USA este valor será del orden del 10 %.

Otro caso distinto sería por ejemplo emplear este sistema para transmitir una ráfaga de 1518 octetos de un sistema 802.3 que representa 1.2 ms a 10 Mbps. Está claro que ráfagas tan pequeñas no se acomodan bien al sistema FRP/DT.

La evaluación de la ganancia estadística que se produce en el FRP/DT puede estudiarse mediante la utilización de fuentes ON/OFF. Sea como antes EA el periodo medio de duración de ON, y ERT el "Round trip delay". El nodo amolda el ancho de banda necesario con una distribución exponencial con un valor medio de ERT + EA. El periodo OFF es distribuido exponencialmente con una media ER. Sea δ el bit rate medio, Ω el valor de pico del bit rate y b el rafagueo. Tenemos:

$$\Omega \, EA = \delta \, (ERT + EA + ER)$$

$$b = \frac{\Omega}{\delta} = 1 + \frac{ERT + ER}{EA}$$

Sea μ la capacidad de multiplex y σn la carga nominal de multiplex. Entonces sólo M0 periodos ON podrán ser simultaneamente soportados por el multiplex con

$$M0 = \frac{\mu \, \sigma n}{\Omega}$$

Suponiendo que la unidad del FRP acaba inmediatamente el proceso de reserva cuando ocurra time-out, la probabilidad de bloqueo estará siempre dada por la probabilidad de espera en un sistema de cola marcoviano con M0 servidores y tamaño finito.

F2. Funciones de Policía modificadoras del flujo.

El método propuesto por [WAM91] y [GUI92] de espaciamiento de celdas está basado en la actuación sobre la fuente como complemento de las funciones de policia. Las celdas son espaciadas con la ayuda de un buffer el cual está organizado como una cola FIFO y sirviendo la salida de la cola al ritmo indicado por la función de policia.

Utilizando por ejemplo una cola del tipo Leaky Bucket física para el control del espaciamiento de celdas para una conexión con un peak bit rate del 10 % de la capacidad del enlace, una celda podría entrar en la red cada 10 ciclos de celda. Generalmente una linea de acceso será utilizada por muchas conexiones virtuales simultaneamente. En este caso el problema del espaciamiento de las celdas es mucho mas complejo yaque todos los VC deben ser espaciaciados individualmente y al mismo tiempo. Desde un punto de vista de implementación no es práctico utilizar colas especificas para cada VC tal como se indica en la figura.

El método utilizado por [WAM91] no utiliza colas específicas por VC. En la figura F1 se indica el diagrama de bloques del sistema.

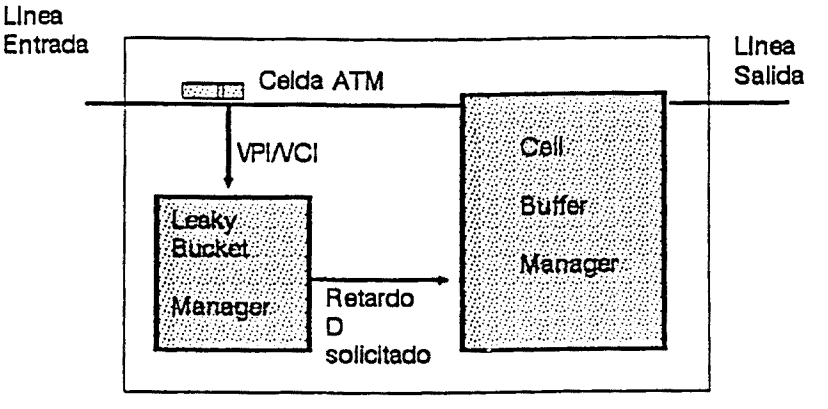

Figura F1. Diagrama de bloques Cell Spacing

Las lineas de entrada y salida que forman parte de un sistema ranurado ATM tienen la misma velocidad de transmisión. Se recibe una celda por cada ciclo. Simultaneamente otra celda abandona el modelo. La funcion de policia por espaciamiento de celdas consiste de dos dispositivos pricipales: El controlador del Leaky Bucket (CLB) y el controlador del Buffer de celdas (CBC). Una celda que entra se dirige hacia el CBC, el cual es informado por el CLB con la ayuda del retardo D de cuanto tiempo esta celda debe esperarse antes de su transmisión o cuando debe ésta ser descartada.

El funcionamiento del Leaky Bucket manager es como sigue: Cuando se recibe una celda se estudia su cabecera para analizar el circuito virtual al que pertenece y asignarle los parámetros de actuación del Leaky Bucket convencional. El valor del contador S se interpreta como el contenido actual de la cola ficticia Leaky Bucket, y podemos analizar los siguientes parámetros: F indica el contenido actual del contador utilizado, C es la capacidad máxima del contador y L es el ritmo de salida del Leaky Bucket ("leak"). El contador S es inicializado a 0. En cada uno

de los VC`s el contador se decrementa en iL donde i denota el tiempo desde la última celda llegada (en celdas).

F3. Funciones de Policía dependientes de la carga de nodo.

Los métodos estudiados hasta ahora no consideran en ningún caso la influencia que en el control de la congestión tiene la interrelación entre los distintos circuitos virtuales.

Se propone una función de policía, mas extensamente definida como función de control de congestión, ubicada en el nodo y que considera los eventos que suceden en el nodo de una forma global.

F3.1 Vigilante inteligente del tráfico.

Esta tesis propone un procedimiento de control de la congestión original denominado "Vigilante inteligente del tráfico (STW)" que considera en su actuación todos los circuitos virtuales que utilizan en un momento dado algún recurso de la red.

Se denomina de nodo porque su actuación se inscribe en el entorno de un nodo de conmutación, aspecto que como veremos mas adelante se puede generalizar.

Su actuación será como complemento de las funciones de policía tradicionales o bien como función de control de la congestión independiente.

Conclusiones parte F.

Las técnicas emergentes en la realización de la función de policía siguen caminos distintos de las tradicionales en el sentido de que tienen en cuenta la carga que debe soportar la red en un momento determinado.

Por un lado utilizan técnicas ya utilizadas en redes de conmutación de paquetes tradicionales, como la reserva de recursos o los "permits" de entrada, o bien utilizan técnicas de reducción del problema. Esto último quiere decir que con el fin de evitar el efecto perjudicial sobre la congestión que producen las ráfagas, se reducen éstas al máximo a pesar de introducir un retardo superior al flujo de datos. Esto se consigue con el espaciamiento de celdas.

Ambas técnicas están en fase de desarrollo e investigación, con resultados muy incipientes.

Se presenta en esta parte además un método de control de la congestión, basado en la policía de nodo, denominado "Vigilante inteligente del tráfico".

Parte G. La Función de Policía como mecanismo de control de la congestión.

El control de congestión en redes ATM no es un problema trivial, dada la diversidad del tráfico, el caracter asíncrono de su aparición y la limitación de recursos.

Esta tesis ha revisado las propuestas referentes a las Funciones de Policía planteadas hasta el momento , que adolecen de grandes problemas de efectividad y objetivos [HON91], y ha analizado los resultados obtenidos. Después de ello se hacen los siguientes planteamientos que hasta el momento no estan suficientemente clarificados:

a) ¿ Debe la Función de Policía prevenir la congestión o por el contrario debe controlar que un usuario introduzca en la red todo aquel tráfico que se le ha garantizado?.

Esta pregunta tiene relación con la garantia por parte de la red de la Calidad de Servicio (QOS) contratada. Bajo el concepto tradicional de Función de Policía, ésta se aplica sobre circuito virtual o sobre "virtual path". Esta actuación localizada no puede mantener el control de la congestión de una forma dinámica y global [GAL90].

b) ¿Debe la función de Policía detectar usuarios "infractores"?.

En primer lugar definimos un usuario "infractor", como aquel que aprovecha el efecto "Jitter" de los multiplexores estadísticos para enviar ráfagas mas intensas a la red de las que está autorizado, salvando el control de la Función de Policía. Si esto ocurre se aumenta de forma sustancial la probabilidad de congestión y supone un grave problema en la gestión de la red [HEI92]. Esto nos conduce a la conveniencia de detectar tales situaciones, pero no obstante deberemos contestar además a la siguiente pregunta:

c) ¿ Hay razones objetivas para ser infractor ?.

En un etorno puramente objetivo podemos llegar a la conclusión de que no hay razones claras para ser infractor, por lo que la segunda pregunta pierde sentido práctico, pero en todo caso se cuestiona el que la función de policia penalize a estos usuarios.

Hay autores como [HEI92] que indican ejemplos de usuarios infractores, pero no dan razones para ello.

d) ¿ Es dimensionable una función de Policía para controlar la congestión y a su vez permitir la QOS requerida por un usuario?.

Podremos comprobar la dificultad de cuantificar los parámetros de diseño de una Función de Policia del tipo Leaky Bucket que satisfagan simultáneamente ambas propuestas del enunciado. En este caso el efecto Jitter será un mal aliado en el diseño [DIR92].

e) ¿ Son las funciones de Policía el camino para prevenir la congestión ?.

Desde un punto de vista general y de una forma aislada, tal como han sido estudiadas y analizadas hasta ahora, las Funciones de Policía suponen una solución aceptable para prevenir la congestión [ECK91]. Pero la realidad de los tráficos presentes en la red están inclinando los estudios de investigación hacia dos tendencias coexistentes que contrastan con los planteamientos iniciales de la Función de Policía. Estas tendencias son la definición de clases de tráfico y la reducción de la presencia de ráfagas.

Todo ello nos conduce hacia una nueva cuestión:

f) ¿Es necesario clasificar la tipología del tráfico para que la red pueda actuar ante él de una forma predefinida y/o evitar en lo posible la producción de ráfagas?.

Estos son dos nuevos caminos, abiertos recientemente, para conseguir una prevención adecuada de la congestión, y nos conduce hacia nuevas técnicas , revisadas en la parte F que tienen en cuenta estas ideas. Por un lado y para determinados tráficos las investigaciones se dirigen hacia la reserva previa de recursos, protocolos del tipo FRP/IT y FRP/DT con mentalidad de "burst switching" [BOY91]. Por otro lado actuando sobre el tráfico por circuito virtual separando las celdas de aquellas ráfagas que presumiblemente producirán problemas bajo el concepto de "cell spacing" [WAM91].

G1. El dilema de la prevención de la congestión o mantenimiento de la calidad de servicio.

Tal como se ha indicado en las partes anteriores, entre las funciones que hemos asignado a la función de Policía nos encontramos con dos de ellas, las indicadas en el título de este apartado, que aparentemente no se pueden mantener simultaneamente controladas y entre las que existe un compromiso.

Ello es debido al efecto "Jitter" que se produce en la red debido a la existencia de multiplexores estadísticos y ser una red asíncrona [FIC91] [IVE92].

Además la arquitectura elegida para actuar la función de policía está basada en los circuitos virtuales o virtual path por lo que independiza las actuaciones sobre la congestión.

Los procedimientos de policía estudiados hasta ahora se podrían englobar en la denominación de "pick-up". Si se utiliza la monitorización de la velocidad de pico basandose exclusivamente en algoritmos del tipo pick-up, no se protege a la red de la congestión [BOY92], y por el efecto jitter tampoco se garantiza la calidad de servicio contratada. Parece entonces que estos mecanismos deben ser suplementados para poder garantizar la ausencia de congestión.

Entonces hemos de tomar la decisión de dimensionar la función de policía para controlar la congestión de una forma global, no dejando pasar celdas que violen la función de policía, o para controlar la calidad de servicio para cada circuito virtual a pesar del jitter.

Veamos como ejemplo como se pueden calcular los parámetros del un leaky bucket con el fin de que absorba el efecto jitter y mantenga la calidad de servicio con un tráfico CBR [DIR91]. Evidentemente los valores calculados serán superiores a los que utilizariamos si no considerasemos el jitter, con lo cual beneficiamos este efecto pero perjudicamos el control de la congestión.

Caracterizamos un Leaky Bucket por tres parámetros:

S : Valor Splash

L : Valor de Leaky

Blim : Valor del Bucket.

Cada vez que llega una celda se añade una unidad S al bucket. Si después de esto el valor del bucket Blev es mas pequeño que el valor máximo Blim, la celda se deja entrara en la red, en caso contrario se rechaza. El bucket se decrementa un valor L por slot. El bit-rate permitido entonces será:

$$\text{bit-rate permitido} = (L / S) \times \text{rate del enlace}$$

Vamos a ver como determinamos el límite inferior de los parámetros de la función de policía que aseguren que un flujo de tráfico CBR sometido a jitter puede ser controlado por la función de policía correctamente.

Consideramos un tráfico CBR como aquel que el intervalo entre celdas de origen es constante y de valor Tcia. Suponemos que cuando este tráfico llega a la función de policía va multiplexado con el producido en otras fuentes, que no estudiamos.

Esto puede suceder en la red de acceso del usario o en un nodo remoto de la red de acceso local. El multiplexor contiene una cola FIFO con un tamaño que denominaremos W.

En primer lugar estudiaremos el efecto jitter que se produce debido a la interferencia de flujos de celdas en el multiplexor que hace que el flujo de celdas observado a la salida del mismo sea distinto por circuito virtual del originado por la fuente.

Si asumimos que la fuente tranmsite celdas cada Tcia slots, el jitter δ se define como el número de slots de desviación de una celda sobre el slot esperado. Este slot esperado está Tcia slots después del slot previo de la misma conexión. En otras palabras el jitter se define como la diferencia en slots entre Tcia de la celda transmitida por la fuente y el Tcia de la celda observada por la función de policía.

Asumiendo un multiplexor de W plazas para celdas, el máximo jitter es W slots. En algunos caso se puede predecir el máximo número de slots de jitter dependiendo del número de fuentes. Aqui consideraremos el caso de W máximo para el jitter.

Podemos considerar dos casos, alto o bajo, dependiendo del bit-rate respecto al jitter:

. Una fuente transmite celdas con un bajo bit-rate con un Tcia grande respecto a W. Entonces Tcia - W > 1, y las celdas no pueden llegar a la función de policía en slots consecutivos.
. Una fuente transmite con un alto bit-rate con un Tcia mas pequeño que W. Entonces Tcia - W < 1, y las celdas pueden llegar a la función de policía en slots consecutivos.

A) Fuentes con un bajo bit-rate

Cuando una fuente transmite un flujo CBR con un Tcia mas grande que W, la mínima distancia entre dos celdas llegando a la función de policía será Tcia - W. El cociente leak-splash debe elegirse entonces de la siguiente forma:

$$\frac{1}{Tcia} \leq \frac{L}{S} \leq \frac{1}{Tcia - W}$$

El bucket debe ser lo suficientemente grande para aceptar dos splashes Tcia-W. Cada slot el bucket se decrementa en L, siendo entonces el límite:

$$Blim = 2S - (Tcia-W)xL$$

Cuando el ratio leak-splash es igual al término derecho de la inecuación anterior, el contenido del bucket estará siempre vacio antes de que la siguiente celda llegue y un límite de bucket de 1 será suficiente. En el caso de que sea igual al término izquierdo

el bit-rate acordado puede ser actuado por la función d epolicía exactamente. Sin embargo y según la expresión anterior de Blim el límite del bucket debe incrementarse hasta Blim = (1 + W/Tcia)xS para prevenir el descartar celdas de la fuente.

B) Fuentes con un alto bit-rate

En este caso en que que el tiempo de interllegadas de celdas es mas pequeño que la capacidad del buffer, dos o mas celdas pueden llegar en slots consecutivos a la función de policía.

Se pueden observar dos fases:

Una fase transitoria con α celdas que se observan en slots consecutivos, seguida de ß slots vacios, y una fase estacionaria que corresponde al bit-rate contratado de una celda cada Tcia slots.

La relación leak-splah quedará en este caso de la siguiente forma:

$$\frac{1}{Tcia} \leq \frac{L}{S} \leq \frac{\alpha}{\alpha+ß}$$

La relación L/S igual al término de la izquierda corresponde al valor medio en la fase estacionaria. El de la derecha corresponde a la fase transitoria.

Para determinar el tamaño mínimo del bucket una vez se ha elegido el ratio L/S, se puede observar el nuvel del bucket en dos momentos. El primer nivel del bucket considerado es cuando α celdas son transmitidas en slots consecutivos, y el segundo nivel de bucket considerado es cuando la primera celda de la fase esatcionaria llega. El máximo nivel de bucket de estos dos momentos corresponde al límite del bucket.

Después de α slots el nivel del bucket se incrementa en

$$Blev1 = \alpha \times S - (\alpha - 1) \times L$$

El segundo nivel de bucket en consideración está ß slots mas tarde cunado se añade un splash debido a la llegada de la primera celda del estado estacionario. Durante ese ß slots no llegan celdas.

$$Blev2 = (\alpha + 1) \times S - (\alpha + ß) \times L$$

Relacionando ambas fórmulas

$$Blev2 = Blev1 + S - (ß + 1) \times L$$

El límite del bucket deberá ser el máximo de Blev1 y Blev2

$$Blim = \alpha \times S - (\alpha - 1) \times L + \{ S - (ß + 1) \times L \}$$

El término encuadrado entre corchetes {} solo se considerará cuando sea positivo.

Naturalmente debemos expresar los valores de α y ß en función de Tcia y W que son los valores conocidos.

Una vez que hemos elegido la relacion L/S podemos calcular el valor limite del bucket. Para expresar α y ß en función de Tcia y W las celdas deben seguirse a través de los lugares que ocupan en el buffer del multiplexor.

En el buffer, la cabeza del flujo de celdas avanza Tcia posiciones del buffer cuando se saca la siguiente celda de la fuente. Por otro lado, la cola se incrementa en pasos de Tcia - 1 slots, porque cada vez la longitud del flujo de celdas en el buffer se incrementa en uno. Cuando la i-ésima celda llega al multiplexor, la cabez del flujo de celdas en el buffer está en la posición

$$Cabeza = Tcia \times (i - 1)$$

y la cola del flujo de celdas está en la posición

$$Cola = (Tcia -1) \times (i - 1)$$

Tanto tiempo como la cola esté en el buffer una nueva celda incrementa la longitud del flujo de celdas. Cuando la cola deja el buffer, la longitud de las celdas con slots consecutivos es fija y corresponde a α. Para obtener α, hemos de obtener el momento que la cola deja el buffer. Esto sucede para el primer i que satisface la siguiente desigualdad:

$$(Tcia-1) \times (i-1) > W$$

Podemos ver que α = i en el buffer y α = i-1 la primera vez que el flujo de celdas deja el buffer. Sustituyendo entonces α por i-1 en la desigualdad anterior

$$\alpha = \left\{ \frac{W}{Tcia-1} \right\} + 1$$

donde {} denota la parte entera de la expresión.

ß podrá obtenerse restando W de la posición de la celda de cola

$$ß = (Tcia-1) \times \alpha - W = (Tcia-1) \times \left(\left\{ \frac{W}{Tcia-1} \right\} + 1 \right) - W$$

finalmente sustituimos las dos ecuaciones anteriores de α y β en la expresión de Blim y obtenemos:

$$Blim = \left\{\frac{W}{Tcia-1}\right\} \times (S-L) + S + \left\langle\!\left\langle S-\left(\left\{\frac{W}{Tcia-1}\right\}\times(Tcia-1)+(Tcia-W)\right)\times L\right\rangle\!\right\rangle$$

considerando el término entre «» cuando es positivo.

Conclusion que obtenemos de este cálculo

A pesar de que se han utilizado hipótesis sencillas para hacer este cálculo,se podría extender este método para casos mas complejos y dimensionar el bucket lo mas adecuado posible al tráfico que ha sufrido un jitter en algún multiplexor estadístico de la red. Pero esto siempre comportará mayores tamaños de bucket que para el caso de que no haya jitter, y por lo tanto disminuiremos el efecto que sobre la congestión efectúa la función de policía, correctamente dimensionada para ello en conjunción con la función de control de admisiones.

G2. Como influye sobre la congestión el "tricky user".

En este apartado entramos en el concepto de usuario estafador o "tricky user" que se indica en la nomenclatura anglosajona [BOR91] [BOY92]. También salen mencionados como "clever user" [HEI92].

Las funciones de policia como sabemos pueden estar basadas en el monitoreo de la velocidad de transmisión de pico, considerando ésta como el mínimo intervalo entre dos celdas consecutivas. La longitud de la máxima ráfaga a velocidad de pico depende del sistema de policia empleado, del importe del jitter tolerado y de la velocidad de pico de la conexión. La debilidad inherente del jitter tolerado por el sistema de policia se aprovecha por el usuario estafador. Los usuarios "trickies" envian ráfagas cortas (por ejemplo paquetes X.25) y negocian una menor velocidad de pico que la que realmente utilizan, pero mantienen su velocidad media. Envian ráfagas que el sistema de policía absorve sin arrestar y se esperan a tener de nuevo el crédito para enviar esas ráfagas a velocidad superior a la contratada. Esto que en principio podría parecer que no tiene mayor importancia ya que mantienen uno de los parámetros contratados, como es la velocidad media, en realidad produce una extraordinaria congestión y una sustancial reducción de la calidad de servicio del resto de usuario que comparten los recursos con el "tricky" [HEI92].

Un "tricky user" envia periodicamente ráfagas a la máxima velocidad con longitud constante que el Leaky Bucket, suponemos este sistema de policia, absorve, con un mínimo periodo de

silencio constante que permite vaciar el contenido del contador bucket.

Comprobaremos como se puede afectar al grado de congestión de la red con un procedimiento de este estilo [HEI92].

Consideramos un primer multiplexor de entrada a la red, el cual es alimentado directamente por las unidades de policia. Suponemos que el multiplexor tiene 16 canales y está diseñado para aceptar tráfico de Bernouilli con una carga total de 0.8 y con una probabilidad de pérdida de celdas menor que 2.5 10E-11. Esto implica un buffer de longitud S=49.

El multiplexor recibe tráfico que ha pasado por las unidades de policia. Se supone que hay m = 1 o m = 2 "tricky users" que utilizan canales distintos del multiplexor. El tráfico se modela para 16-m fuentes de Bernouilli de igual carga.

Se supone que se utiliza un procedimiento de reserva de valor de pico en el control de aceptación y que la utilización del enlace de salida es del orden del 80 % (carga 0.8). Las unidades de policía actuan sobre el valor de pico de cada conexión.

Los "tricky users" envian periodicamente una ráfaga de longitud BL seguido por un periodo de silencio de longitud SL. La ráfaga de longitud BL es tan larga como la función de policia tolera. El periodo de silencio es justamente de la longitud suficiente para que el contador de la función de policia admita otra ráfaga de longitud BL. Los valores de BL y de SL dependen del valor del jitter tolerado y del valor de pico negociado Rc.

La carga total del usuario "tricky" es

$$P1 = m.BL/(BL*SL)$$

que debe de ser aproximadamente igual a m.Rc.

La carga total del tráfico de bernouilli se elige en:

$$P2 = 0.8\ m.Rc$$

dividida equitativamente por todas las conexiones.

Ambos usuarios "trickies" se comportan con las mismas condiciones y valores, aunque por distintos canales. El peor caso será el que las ráfagas de ambos "trickies" empiezan al unísono, que en las pruebas que se presentan se supone que ocurrirá un 5% de las veces.

Los valores que se usarán en el leaky bucket serán los siguientes:

F Valor de llenado por cada celda que llega

L Valor de salida (leak) por cada ciclo de celda

C Capacidad del bucket

Cada slot el contador se decrementa en L pero no por debajo de cero. Cada vez que llega una celda, ésta es descartada si después de incrementar el contador por F (después de un decremento de L) resulta un valor por encima de C. En otro caso la celda se acepta y el contador se incrementa en F.

La relación L/F es la velocidad de celdas permitida. Se elige el valor de $Rc \leq L/F \leq Rc.(1+0.01)$. Esto resulta un Leaky Bucket que permite un 1% como máximo por encima de la velocidad de transmisión de celdas contratada Rc. Entre las parejas de L y C que cumplen la desigualdad anterior se elegirán las mas pequeñas.

La capacidad del bucket C sin tolerancia al jitter debería ser igual a F. Para permitir este jitter se elige un valor muy superior de C como el que se indica:

$$C = F + Delta.L$$

El estudio del valor que puede tomar Delta se indica en [NIE90], y este valor vendrá especificado por la red. Un retardo variable en la red superior a este valor de Delta (en celdas) puede conducir a descartar celdas de fuentes que se esten comportando bien. El CPN (Customer premises network) debe asegurar que $(1-\alpha)$ del retardo variable de celdas, es como mucho Delta cuando α es la probabilidad de pérdida de celdas para la función de policía (por ejemplo α = 10E-10). Esto quiere decir que el CPN debe asegurar que la probabilidad del retardo de celdas variable que excede Delta es menor que α.

La máxima longitud de ráfaga permitida por el Leaky Bucket será

$$BL = \{ (C-L/(F-L) \}$$

donde {} denota la parte entera de la expresión contenida.

El máximo silencio necesario para vaciar un bucket lleno será

$$SL = \{ C/L \} + 1$$

Elegiremos un valor de Delta de 150. Se ha elegido este valor después de considerar un CPN simple con un multiplexor con carga de Bernouilli igual para todos los canales. Si las celdas han de pasar por mas de una cola se debería incrementar ese valor [HEI92].

Podemos indicar graficamente la probabilidad de pérdida de celdas (figura G1) para una distribución de Bernouilli para este caso, en donde se puede observar el efecto pernicioso de los

usuarios "trickies". No obstante este efecto se puede reducir incrementando el valor de los buffers.

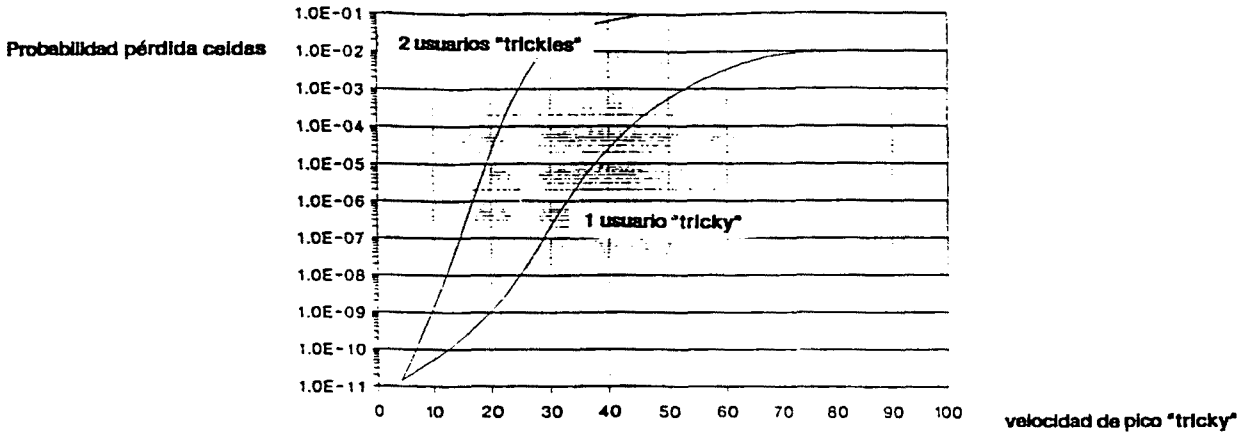

Figura G1. Efecto pernicioso usuarios maliciosos

Del estudio presentado deducimos lo siguiente:

1. Con las funciones de Policía tradicional un usuario "tricky" puede producir daño en la congestión y en el mantenimiento de la calidad de servicio del resto de usuarios. No obstante se puede afirmar que se puede contrarestar este efecto aumentando los valores de los bufferes. O sea es un problema de dimensionado correcto de los recursos.

2. El método que pueda ser utilizado para contrarestar el efecto del "tricky user" no debe perseguir a este tipo de usuarios, sino que simplemente debe actuar sobre ellos sólo cuando sea necesario para mantener la calidad de servicio del resto y evitar la congestión. Se piensa que en el punto de entrada a la red en donde se actúa con la función de policía, la red conoce el número y características de los circuitos virtuales que están simultaneamente funcionando. Por ello la red puede saber el proceso de multiplexación estadística que se está aplicando y deducir si una ráfaga de celdas que supera la velocidad de pico contratada es debida al efecto multiplexor estadístico o a un fraude. Por ejemplo si hay cinco fuentes con una velocidad de pico que corresponde a una celda cada 20 slots, si llegan dos celdas seguidas de una fuente ha de ser forzosamente un "trickie", o si antes de una ráfaga que viola la velocidad de pico hay slots vacíos indica que ha habido una actuación irregular. Ante una evidencia como ésta la red puede actuar sobre el usuario infractor como lo haría la función de policía.

3. No se adivinan razones suficientemente claras para que un

usuario obtenga beneficios de ser "tricky", ya que él sale también perjudicado en su calidad de servicio aunque engañe a la función de policía.

4. Estas técnicas como las presentadas que incrementan el valor del Bucket para prevenir el efecto Jitter no son las mas adecuadas como función de policía, ya que reducen el objetivo fundamental de las mismas que es el control del tráfico. Por lo tanto no es dimensionalble una función de policía para controlar la congestión y a su vez controlar la calidad de servicio contratada.

5. Todo ello nos conduce a la necesidad de dirigir las investigaciones hacia nuevos caminos que solventen los problemas planteados.

Aparte de los mencionados de espaciado de celdas [WAM91] [BOR91] [BOY92] en fase de investigación y sin resultados publicados, esta tesis estudia un método basado en la policía de nodo denominado "Smart Traffic Warden" que se analiza en la parte H siguiente.

Conclusiones parte G.

De los estudios presentados se deduce que no se puede mantener simultaneamente la calidad de servicio requerida y controlar a su vez la congestión, ya que para solventar el efecto jitter y conseguir la QOS contratada es necesario modificar los parámetros de la función de policía al alza, con lo cual se reduce el efecto del control de la congestión.

Por otro lado la existencia de "tricky users" puede perjudicar notablemente la congestión del resto de usuarios que comparten los recursos. No parece necesario detectarlos ni penalizarlos, ahora si parece que se debe de encontrar algún método que evite ese efecto de congestión inducido.

Parte H. Vigilante inteligente del tráfico (STW).

H1. La vigilancia inteligente.

La aportación principal de esta tesis es una línea de investigación abierta sobre el control de la Congestión denominada "Smart traffic Warden (STW)", en la cual se configura la Función de Policía de "Nodo". En este caso se realiza una acción conjunta entre las Funciones de Control de admisiones (CAC) y las tradicionales funciones de Policía, de tal forma que las consideraciones sobre la Congestión se abordan desde la perspectiva de nodo o enlace y no sobre circuito virtual o "virtual path" [SANc91].

Además el concepto de STW se extiende a toda la red, y no sólo a los puntos de entrada, aplicándose en todos los multiplexores de red, considerando un conmutador NxM como M multiplexores estadísticos de N entradas y una salida.

En estas condiciones se estudia el problema bajo dos puntos de vista. En primer lugar aplicando STW y funciones de Policia tradicionales como el Leaky Bucket, y en segundo lugar aplicando exclusivamente el concepto STW como método de control de la congestión.

Los resultados obtenidos por simulación permiten comprobar y comparar el efecto de control de congestión de métodos como el Leaky Bucket y el STW en sus dos variantes, utilizando como fuentes de tráfico secuencias que provienen de codificadores de vídeo reales proporcionados para este estudio por Telefónica I+D, y explicadas en la parte D.

Como conclusión se propone que la Función de Policía sea un dispositivo de control de flujo que "vigile" los excesos de las conexiones en curso, a fin de que:

1. Se garantize la calidad de servicio contratada para esas conexiones.

2. Se evite la congestión.

3. Se mejore todo lo posible la utilización de los recursos disponibles.

Para ello se podrá actuar de una forma:

a) tolerante con los excesos si no perjudican a nadie y no producen congestión (STW) con lo cual se cubren los tres objetivos planteados.

b) intolerante por defecto (funciones de policia tradicionales) con lo que se cubren parcialmente los objetivos planteados.

En esta propuesta se combinan algoritmos de control de admisiones basados en la carga con funciones de policía a fin de conseguir no solo una mejor utilización del ancho de banda del enlace disponible sino una menor acción punitiva de la función de policia.

La idea básica del algoritmo se complementa con las funciones conocidas de CAC y Policia, de tal forma que actúa sobre el flujo de celdas de entrada a la red en el sentido de que si las condiciones lo permiten, a pesar de que la función de policia impida el paso de una celda, el vigilante inteligente de la red la dejará pasar.

Durante el proceso de conexión se fijará un mínimo de ancho de banda y de recursos para una determinada conexión según indica el método de CAC basado en la carga. Durante la conexión la función de policia asigna el ancho de banda y los recursos para cada conexión dependiendo de la demanda global de todas las conexiones que comparten el enlace.

Cuando la función de policía actúa descartando celdas, STW estudiará la posibilidad de que estas celdas no se descarten y modificará en su caso los parámetros de la función de policia para conseguirlo. Si es un Leaky Bucket incrementará la velocidad de Leak, si es un mecanismo de ventana incrementará el número de celdas previsto en la ventana, y si es un mecanismo dinámico modificará el parámetro dinámico, pero en este caso en función de las otras conexiones y no en función de su misma conexión como habia actuado hasta ahora.

Este sistema nos conduce a una disminución de la severidad en el tratamiento de las ráfagas y una menor degradación de la calidad de servicio ofrecida, por supuesto sin degradación de la calidad de servicio del resto de conexiones.

Un símil de este procedimiento es suponer una red viaria de automóbiles totalmente establecida y diseñada en lo que se refiere a CAC y a PF, en la cual el control de admisiones y la función de policia se realiza a través de semáforos en cada cruce que equivale a cada nodo. En un momento determinado a un vehiculo (celda) se le pone el semáforo en rojo en el cruce de entrada a la red impidiendo su paso dado que sobrepasa el tráfico autorizado para esa conexión, aunque el resto de conexiones no tranmiten nada en un caso extremo. Es en ese momento cuando STW que es un guardia de tráfico del nodo (inteligencia distribuida

) el cual dispone de la información necesaria para saber la situación global del cruce, deja pasar la celda a pesar de tener el semáforo en rojo, porque sabe que el puerto de salida del nodo hacia donde se dirige ese vehiculo tiene suficiente capacidad libre para absorverlo, sin degradar la calidad de servicio del resto de conexiones.

Como podemos observar, este método se añade a los métodos conocidos y proporciona un segundo nivel de Función de Policia de celda. Es la función de Policia de Nodo.

En realidad el incremento en los parámetros de la Función de Policia requerido se puede considerar como la admisión de una nueva conexión con los requerimientos de recursos iguales a la diferencia entre los que se tenian y la nueva demanda.

Sea δ el bit rate concedido por la PF
Sea δreq el incremento solicitado

STW actuará obteniendo si es posible el nuevo bit rate δn tal que

$$\delta n = \delta + \delta req$$

En este caso CAC está intimamente relacionado con la función de policia y debe de aplicarse ambos criterios para la utilización de STW.

Este sistema propuesto, no es una modificación de una función de policia, sino que es aplicación adicional a cualquier función de policia. En realidad es aplicación del Virtual Path, y nos proporciona un control de Congestión mas completo que los tradicionales en los niveles inferiores de congestión, o sea cuando no hay congestión.

Este método lo englobaremos en el tercer nivel del modelo de Gestión de red explicado en la parte E, ya que integra y complementa las funciones de Control de Admisiones y de función de Policia.

Con este procedimiento podría ocurrir que un usuario acabara la conexión después de haber enviado mas información que la estadísticamente contratada con la red, aunque esto habría sido así sin degradación del QOS de nadie. Esta información será simplemente traspasada del nivel tres de gestión de red al nivel uno a través del nivel dos, a fin de que se proceda al cargo extra correspondiente o bien a la reconsideración del contrato del usuario con la red.

Recordemos que la clasificación propuesta en esta tesis de las funciones de policía queda determinada de la siguiente forma:

Nivel celda: *Funciones no modificadoras del flujo

- Independientes carga
- Dependientes de la carga
  - . de red
  - . de nodo (STW)

*Funciones modificadoras del flujo

Nivel ráfaga: Funciones FRP/DT y FRP/IT

Si suponemos que estamos trabajando en un sistema de función de Policia a nivel de celda del tipo Leaky Bucket el algoritmo de STW sería el siguiente:

1. El contador del leaky Bucket se incrementará en uno cuando llega una celda y se decrementará de acuerdo con la velocidad media de celdas contratada.

2. Si el contador excede el valor máximo fijado para un determinado QOS, el procedimiento STW incrementará la velocidad de salida del Leak. Esto quiere decir que STW estudiará si la celda puedce entrar en la red sin modificar el QOS de las otras conexiones en servicio. Esto será solo posible si lo permite el algoritmo de Control de Admisiones al haber recursos suficientes, y debe de tomar la decisisón en el tiempo de una celda.

STW permitirá que la conexión se adapte automaticamente y de una forma dinámica a las condiciones de congestión de la red, siendo capaz por otro lado de incrementar o decrementar el ancho de banda utilizado de una forma continua de acuerdo con las condiciones de la red, pero siempre considerando las condiciones del contrato.

Localizaremos STW en el acceso de usuario a la red y en todos los enlaces entre nodos de la red. Pienso que se debe utilizar STW en todos aquellos lugares de la red en donde exista una multiplexación N a 1, considerando un conmutador N x M como M multiplex de N a 1.

H2. Parámetros que definen la actuación de STW

STW estará en función de la tasa de pérdida de celdas, el retardo máximo permitido y el "delay jitter" permitido. Todos estos parámetros nos definen la Calidad de servicio del sistema.

STW deberá vigilar que todos estos parámetros se mantienen dentro de márgenes en todas las conexiones del nodo.

Dadas las características generales de las redes ATM, ya explicadas en la parte B de esta tesis, el retardo es un parámetro que depende sustancialmente del tiempo de propagación dados los tamaños que se preveen de los buffers en los nodos, por lo que no será manipulable facilmente. Esto quiere decir que ante cualquier actuación para medir o alterar la calidad de servicio no deberemos tener en cuenta los retrasos producidos en los nodos.

En referencia al "delay jitter" y dado que es un parámetro intrínseco a la existencia de multiplexores estadísticos en la red, y dados los márgenes autorizados superiores en todo caso a los que puede introducir un nodo, no se considera su control imprescindible para la gestión del control de la congestión según el modelo presentado de STW.

En una red convencional, se supone que el número máximo de tránsitos será de tres o cuatro.

Esto quiere decir que el parámetro en el que nos basaremos para controlar la congestión será el de la probabilidad de pérdida o tasa de error.

H3. Control de la congestión con STW.

Los problemas generales detectados en el control de la congestión con los métodos propuestos hasta ahora son :

1. Las funciónes del tipo PICK-UP como el Leaky Bucket pueden controlar la congestión de forma general, pero a su vez pueden ser fácilmente engañadas. Según [HEI92] y considerando un modelo de fuente de dos estados igual que el que se utiliza en esta tesis, los "tricky users" denominados "clever" aumentan la tasa de pérdida de celdas de una forma sustancial dependiendo de como haya sido dimensionado el LB y de su "safety factor". Esto ha sido estudiado en la parte G de esta tesis.

2. Con los métodos propuestos hasta ahora no se puede controlar la velocidad de pico de una forma eficaz. Cualquier variación de la misma sobre la declarada puede producir congestión. El jitter es determinante en este proceso. Asi mismo se ha estudiado este aspecto en la parte G.

3. El "delay jitter" que inevitablemente se produce en los multiplexores de la red hace que el tráfico de ráfagas real en la red no sea igual que el producido por la fuente. Este jitter hace que incluso una conexión CBR llegue como VBR al LB, con lo cual dificulta mucho su dimensionamiento. [DIR92] hace un estudio de ésto y dimensiona el LB para que el hecho de que el CBR llegue como VBR no le penalize. Esto ha sido estudiado en la parte

anterior, pero surge siempre la misma pregunta:

¿Para qué se utilizan las funciones de policía?

La respuesta sería: Para evitar que haya congestión debido a que alguien utilize mas recursos de los que tiene contratados.

Por lo tanto siempre que considere los efectos del jitter en el dimensionamiento de la FP me variará los tamaños de algunos parámetros, haciéndolos mayores y disminuyendo por tanto el efecto de control previsto originalmente para la FP.

H4. Análisis y evaluación del STW.

La implementación de STW y su evaluación a sufrido diversas modificaciones desde su idea original que nos configura la siguiente evolución:

1. Funciones de policía con asignación dinámica del ancho de banda utilizado [SANb90] .

2. Control de la congestión con STW basado en la tasa de error [SANc91].

3. Control de la congestión con STW basado en la policía de nodo, presentado en esta tesis.

Analizaremos estos tres apartados por separado con indicación de los resultados obtenidos.

H4.1 Funciones de Policía con asignación dinámica del ancho de banda utilizado [SANb90].

Este trabajo corresponde a la idea original sobre la modificación de las funciones de policía de tal forma que se varian de una forma dinámica los parámetros mas representativos de la FP manteniendo la calidad de servicio requerida .

En esta propuesta los algoritmos de CAC basados en la carga se combinan con funciones de policía del tipo Leaky Bucket y con la ganancia de multiplexación producida en la red con el fin de conseguir no sólo una mejor utilización del ancho de banda del enlace sino una menor acción punitiva de la función de policía.

Durante la fase de conexión se reserva un mínimo de ancho de banda para cada conexión (método de CAC basado en la carga). Durante la conexión la función de policía asigna un ancho de banda para cada conexión dependiendo de la demanda global de las conexiones que comparten un enlace (ganancia de multiplexación estadística).

Cuando se llena el bucket, antes de descartar o marcar celdas, se amplia la velocidad de salida del LB si las condiciones lo permiten. Esto quiere decir que periodos de alta actividad con velocidades de transmisión altas o de pico, que un leaky bucket normal castigaría, pueden ser aceptados sin producir congestión si hay suficientes recursos disponibles en el enlace.

En este algoritmo se considera el concepto de congestión en el enlace y no en el nodo. Modelos posteriores perfeccionarán este método considerando el nodo.

Con el fin de garantizar la velocidad media de transmisión contratada , el algoritmo tendrá en cuenta los periodos de baja actividad en los cuales la velocidad de transmsión está por debajo de la media contratada, asignando una menor velocidad de salida del LB de la contratada durante estos intervalos compensando así los excesos de tráfico de los periodos de alta actividad.

Este mecanismo conduce a un tratamiento menos severo de las ráfagas, y a una menor degradación de la calidad de servicio de una determinada conexión asumiendo que por supuesto no se degrada la QOS del resto de conexiones.

El incremento de la velocidad de salida del LB puede verse como la admisión de una nueva conexión con un ancho de banda igual a la diferencia entre la velocidad instantánea y la velocidad media tal como se indica en la figura.

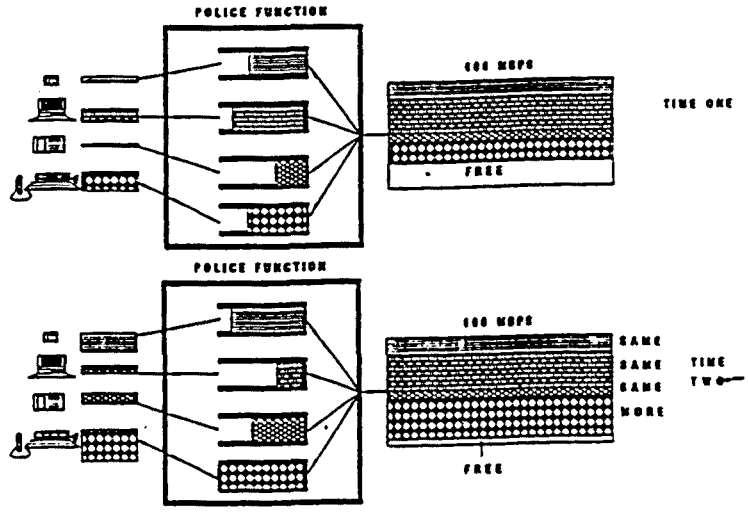

Figura H1. Asignación dinámica del ancho de banda

Podría suceder que una conexión después de haber tenido el crédito concedido por el algoritmo concluyera sin haber compensado la diferencia. Entonces el usuario ha violado el contrato y se le debe cargar una facturación extra por ello. El algoritmo debe controlar este tipo de situaciones y evitar que ocurran cuando perjudican la QOS de los demás.

Concretamente el algoritmo probado y evaluado sigue el siguiente esquema:

a) La función de policía monitoriza la velocidad de transmisión instantánea.

b) El contador del Leaky Bucket se incrementa por uno y se decrementa de acuerdo con la velocidad de transmisión contratada.

c) Si el contador excede el valor máximo contratado se activa el procedimiento de incremento de la velocidad de salida del bucket. En un principo la velocidad de salida se hace igual a la velocidad instantánea de entrada. Esto sólo será posible si el algoritmo de CAC lo permite y ello quiere decir que hay suficiente ancho de banda disponible en el enlace. Si este no es el caso entonces el remanente de ancho de banda será compartido por todas aquellas conexiones que exceden su velocidad media contratada.

d) Si el contador está por debajo de un valor mínimo prestablecido, la velocidad de salida se puede decrementar a un valor mínimo, asignado por el algoritmo de CAC. Esta situación continuará hasta que se recupere el valor medio contratado.

Escenario de evaluación

La evaluación de esta propuesta se ha realizado por simulación con ordenador.

La fuente de celdas ATM se ha modelado usando el modelo de estados MMDP que se explica detalladamente en la parte D. Por simplicidad de cálculo se utiliza el modelo de dos estados. En cada estado las celdas se generan a una velocidad constante. El tiempo entre estados está distribuido geometricamente. La probabilidad de transición de un estado a otro es igual a 1, de tal forma que los periodos de alta y baja actividad se generan consecutivamente.

Para evaluar el Leaky Bucket en su aproximación estática, sólo se utiliza el tráfico generado por una fuente ya que la función de policía se aplica sobre un único circuito virtual independientemente de los otros.

Como que la propuesta toma ventajas de la ganancia de multiplexación, el tráfico de entrada será un tráfico mixto correspondiente a diferentes circuitos virtuales que comparten el mismo enlace.

Los tipos de tráfico seleccionados están reflejados en la tabla siguiente y se han elegido fuentes de gran agresividad,

bastante lejanas de la realidad, pero comparables con las que ha analizado en [RACb90] que nos permite estudiar correctamente los resultados.

	State Mean Time		Mean Traffic Intensity	
Burstiness	High Activity	Low Activity	High Activity	Low Activity
2	3.5 ms	3.5 ms	70 Mbps	0
3	3.5 ms	7 ms	105 Mbps	0
5	3.5 ms	14 ms	175 Mbps	0

Burstiness = High activity bitrate / Mean bitrate

Figura H2. Tráficos seleccionados

Los tipos de tráfico se han definido de acuerdo con el número de estados, la velocidad de transmisión para cada estado, el tiempo medio de estancia para cada estado y la probabilidad de estancia en un estado dado.

El entorno de simulación está reflejado en el dibujo siguiente:

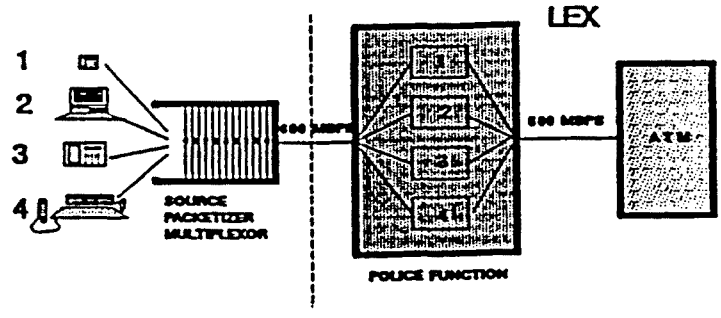

Figura H3. Entorno de simulación

La cola en la entrada de la función de policía se modela como un multiplexor estadístico ideal. Las celdas generadas por las fuentes de tráfico son encoladas y entregadas al sistema de policía en orden FIFO.

La unidad de tiempo para el simulador es de un ciclo, elcual es el tiempo de transmisión de una celda ATM a 600 Mbps. La celda ATM esta formada por 5 bytes de cabecera y 48 bytes de información. Esto nos da aproximadamente 0.7 microsegundos por ciclo. Para obtener resultados apurados el proceso de simulación

se ejecuta durante 32 millones de ciclos que corresponde a 22 segundos de tiempo real aproximadamente. Hemos utilizado un tamaño de bucket constante de 10000 celdas en todas las simulaciones.

El simulador es un programa especial que optimiza el tiempo de CPU y está escrito en VMS PASCAL y ejecutado en una estación 3200 VAX.

Para cada tráfico seleccionado, se han repetido las simulaciones para el algoritmo propuesto (Leaky Bucket with dynamic bandwidth allocation) y para el Leaky Bucket normal con el fin de comparar los resultados.

Las simulaciones han sido hechas en cada caso con 2,10 y 14 fuentes, lo que quiere decir con una intensidad de tráfico de 0.116, 0.583 y 0.816 respectivamente.

En la tabla siguiente se ha recogido los resultados en términos de números de celdas marcadas. Observando estos resultados podemos ver que en la versión modificada del leaky Bucket el número total de celdas marcadas es muy reducido en comparación con el número de celdas marcadas por el Leaky Bucket normal.

# of Sources	Burstiness	Leaky Bucket	Leaky Bucket with Dynamic Bandwidth Allocation
2	2	174269	0
	3	415119	0
	5	538158	0
10	2	997181	0
	3	2348004	27
	5	2877511	14330
14	2	1396169	0
	3	3423917	109520
	5	3812735	344400

Figura H4. Resultados generales de la simulación.

Obviamente el número de celdas marcadas crece cuando se incrementa el grado de rafagueo y la carga. Aunque incrementado el tamaño del bucket en la función de policia normal se podrian obtener los mismos resultados, en términos de celdas marcadas, se debe notar que entonces se obtendría una menor actuación como función de policia.

La siguiente tabla contiene el número de celdas marcadas para cada una de las fuentes en el caso de 14 fuentes y rafagueo b = 5. En esta tabla se puede observar que la ganancia obtenida por el algoritmo modificado del leaky bucket no es solo global sino individualmente para cada conexión.

Source	Leaky Bucket	Leaky Bucket with Dynamic Bandwidth Allocation
1	257869	24948
2	269168	30186
3	428044	47440
4	211124	20025
5	240405	15534
6	227366	18321
7	324975	42697
8	272820	13951
9	257949	11034
10	275447	25194
11	222193	25150
12	311928	34453
13	185807	4865
14	257540	30582
Total	3812735	344400

Figura H5. Resultados concretos para b = 5

La conclusión que obtenemos de esta primera veresión del algoritmo propuesta es que optimiza el ancho de banda disponible, mantiene las condiciones generales de una función de policía y no restringe el ancho de banda que necesita un usuario si no es necesario. Por otro lado cuando un usuario excede las condiciones contractadas, se garantiza el QOS del resto de usuarios que comparten el enlace. La red a su vez descartará las celdas marcadas si es necesario y utilizará protecciones adicionales contra la congestión, que aquí no se indican.

Esta última consideración que queda pendiente en este trabajo, se soluciona con una modificación en el algoritmo propuesto y que se describe a continuación.

H4.2 Control de la congestión con Smart Traffic Warden (STW) basado en la tasa de error [SANc91].

En este caso aplicaremos el mismo concepto de variación

dinámica del ancho de banda, y que denominaremos de ahora en adelante "Smart Traffic Warden" o vigilante inteligente del tráfico.

La idea básica es la misma que en el caso anterior, sólo que ahora se aplicará sobre la congestión en un nodo y no sobre el enlace.

Según al clasificación vista anteriormente esta sería una función de policía de nodo, que refuerza alguna otra función de policia. En nuestro caso nos basaremos en el Leaky Bucket. Tal como se indica en la figura H6 ,STW combinará el algoritmo de control de admisiones con el de función de policia.

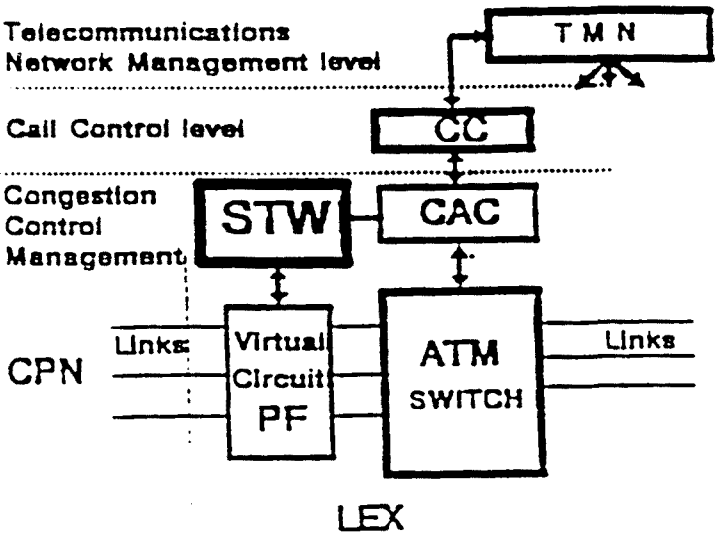

Figura H6. Ubicación del STW

El algoritmo STW planteado se define como sigue:

1. El contador del Leaky Bucket se incrementa en uno cuando llega una celda al sistema de policia, y se decrementa al ritmo previamente acordado en el contrato con la red.

2. Si el contador excede un valor predeterminado, dado para una calidad de servicio determinada, el procedimiento de aumentar la velocidad de salida del bucket se activa (STW). Esto queire decir que STW decide cuando la celda puede entrar en la red sin modificar la calidad de servicio de las otras conexiones contratadas. Esto será posible si el Control de admisiones lo permite ya que hay suficientes recursos para ello.

Esto quiere decir que STW permitirá de forma automática y dinámica que la conexión se adapte a las condiciones de congestión de la red, ya que será capaz de incrementar o decrementar el ancho de banda utilizado de acuerdo con las condiciones de carga de la red, pero tomando siempre en cuenta

las condiciones de contratación. Si el usuario acaba la conexión en condiciones diferentes de las contratadas, esto habrá sucedido sin congestión, y STW pasará esta información al TMN a través del CC con el fin de cargar aquellas facturaciones extras necesarias o revisar el contrato con el usuario.

Localizaremos STW en el acceso de usuario y en todos los enlaces de la red. Creemos que se debe utilizar STW en todos aquellos lugares de la red donde hay multiplexores estadísticos N x 1, considerando un switch N x M como M multiplexores de N x 1.

En la siguiente figura se puede observar esta ubicación:

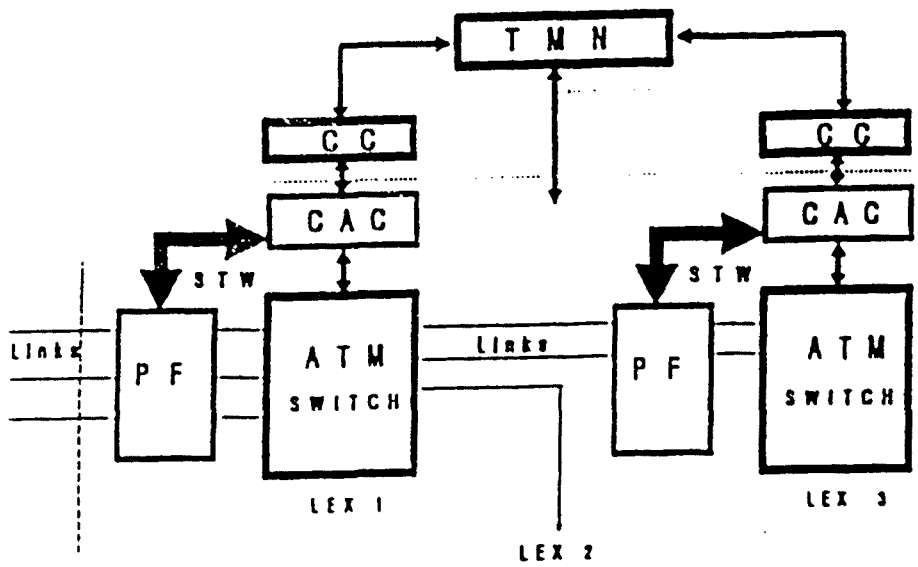

Figura H7. Ubicación general en la red del STW

Aquí es importante redefinir el concepto de congestión que se aplicará con STW, concepto que se aplicará para cada circuito virtual.

"La congestión en un circuito virtual será aquel estado de un nodo en el que no es capaz de absorver mas celdas de ese circuito virtual sin modificar de una forma sustancial su QOS".

La función STW será función de la tasa de pérdidas de celdas, del máximo retardo producido en el nodo, y del máximo delay jitter permitido en el nodo.

$$STW = f (Clr, D, Dj)$$

STW permitirá que una celda que haya violado la función de policía establecida pase al buffer de salida del nodo en su ruta correspondiente, si ello no representa un incremento del Clr, D, y Dj para cada uno de los circuitos virtuales que comparten ese buffer de salida, aplicando la función AND

```
IF Clr < Max AND D < Max AND Dj < Max
   THEN la celda puede entrar en el nodo
ELSE la función de policía arresta la celda
```

El problema consiste en saber como afecta la entrada de una nueva celda en el buffer de salida en la QOS del resto de conexiones que comparten esa ruta de salida.

Para ello se hacen las siguientes propuestas:

1. No se considera el efecto del incremento del retardo que se produce en un buffer por el hecho de introducir en él una celda que según la función de policía no debería haber entrado. Tal como se ha indicado anteriormente en la parte B de esta tesis, el retardo de propagación en una ruta de 1000 Kms es del orden de 5000 celdas, y los bufferes serán del orden de 64 celdas con un máximo de 3 nodos por ruta lo que nos da un retardo máximo de 192 celdas en el caso peor.

2. No se considerará en general el efecto sobre el delay jitter por el hecho de modificar la posición en la que debería entrar una celda en el buffer por el STW, dado que este fenómeno es inherente a la existencia del proceso de multiplexación estadístico, y en todo caso es incontrolable ya que se desconoce el "delay jitter" que pueda llevar acumulado una conexión virtual. No obstante en esta simulación si se ha tenido en cuenta el efecto que sobre el jitter puede haber tenido el STW, y los resultados han sido despreciables.

3. Sólo se considera el efecto que se produce respecto a la tasa de pérdida de celdas, por el hecho de introducir en el buffer por el STW una celda de una conexión, en el resto de conexiones.

Escenario de evaluación

La evaluación de esta propuesta se ha ejecutado por simulación de ordenador.

La fuente de celdas ATM se ha modelado siguiendo el esquema MMDP explicado en la parte D. Por simplicidad se ha utilizado el modelo de dos estados. En cada estado las celdas se generan a una velocidad constante. El tiempo de permanencia para cada estado está geometricamente distribuido. La probabilidad de transición de un estado a otro es igual a 1, queriendo decir que los periodos de alta y baja actividad se generan sucesivamente de una forma alternada.

El tráfico seleccionado se presenta en la tabla siguiente. Estos tipos de tráfico se han seleccionado de acuerdo con el

número de estados, la velocidad de transmisión para cada estado y la probabilidad de estar en un determinado estado.

	State Mean Time		Mean Traffic Intensity	
Burstiness	High Activity	Low Activity	High Activity	Low Activity
2	3.5 ms	3.5 ms	70 Mbps	0
3	3.5 ms	7 ms	105 Mbps	0
5	3.5 ms	14 ms	175 Mbps	0

Burstiness = High activity bitrate / Mean bitrate

Figura H8. Tráfico seleccionado

La función de policía utilizada es el Leaky Bucket, y para evaluar con este tráfico las características de dicha función, se ha realizado una aproximación estática con el tráfico generado por una sola fuente ya que la función d epolicía se aplica sobre un único circuito virtual independiente de los otros.

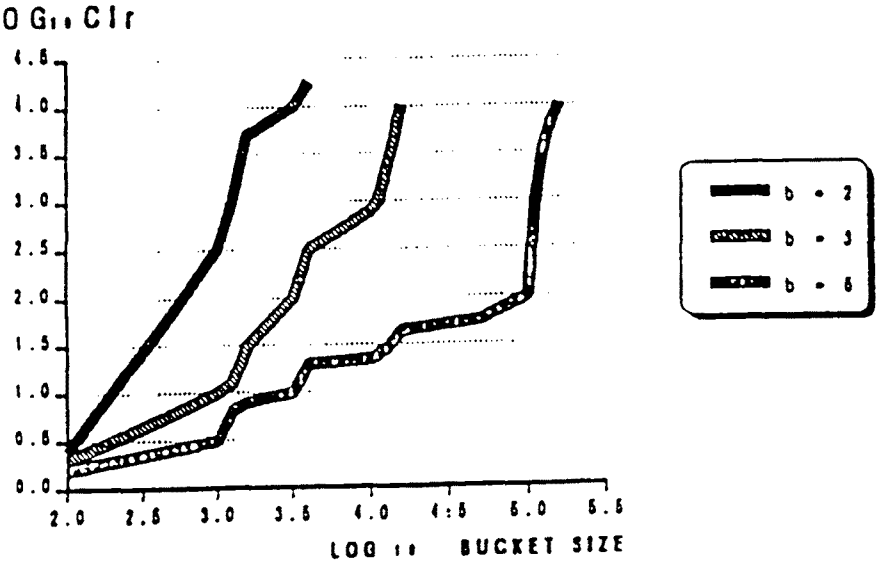

Figura H9. Relación tamaño del bucket y tasa de error

Esto nos permite calcular el tamaño del bucket para una determinada calidad de servicio QOS, considerando como se ha indicado antes unicamente la tasa de pérdida de celdas.

Este resultado se puede apreciar en la figura H9 anterior, en donde se puede comprobar que la calidad de servicio mejora cuando el bucket es mayor, ahora bien se reduce el efecto de control

sobre la congestión que se desea con la función de policía.

En las mismas condiciones que en el apartado anterior, la unidad de tiempo para el simulador es de 1 ciclo, el cual es el tiempo de transmisión de una celda ATM a 600 Mbps.

Se han utilizado diferentes tamaños de bucket para cada simulación dependiendo del rafagueo, y un tamaño constante del buffer de salida del nodo. Dado que el tráfico utilizado es muy activo, y que lo que se pretende con la simulación es comprobar la diferencia de funcionamiento en el control de la congestión con y sin STW, y no la puesta en marcha real de un sistema de estas características, el tamaño de buffer elegido es de 1000 celdas.

Los tamaños elegidos de bucket para una tasa de pérdidas de celdas de 10E-03 son de

Rafagueo:	b= 2	b= 3	b= 5	
Bucket:	5000	10.000	115.000	celdas

En la tabla siguiente se puede observar el conjunto de la simulación y los parámetros que intervienen:

LB = Leaky Bucket normal

STW = Smart Traffic Warden

b = Rafagueo

# Fuentes	b=2		b=3		b=5	
	LB	STW	LB	STW	LB	STW
1	9x10E-04	0	1.6x10E-03	0	2.9x10E-03	0
2	6.5x10E-04	0	3.4x10E-03	0	4.9x10E-03	0
8	5.4x10E-04	0	4.7x10E-03	0	2.9x10E-02	2x10E-02
14	1x10E-03	9.9x10E-04	1x10E-02	1x10E-02	1x10E-01	1x10E-01

En la simulación propuesta sólo se considera un nodo. Se considera que cada enlace tiene un único circuito virtual que ataca el buffer de salida, para obtener una situación de que varias celdas quieran entrar en el buffer de salida del nodo simultaneamente.

En la figura H10 siguiente se indica el entorno de simulación:

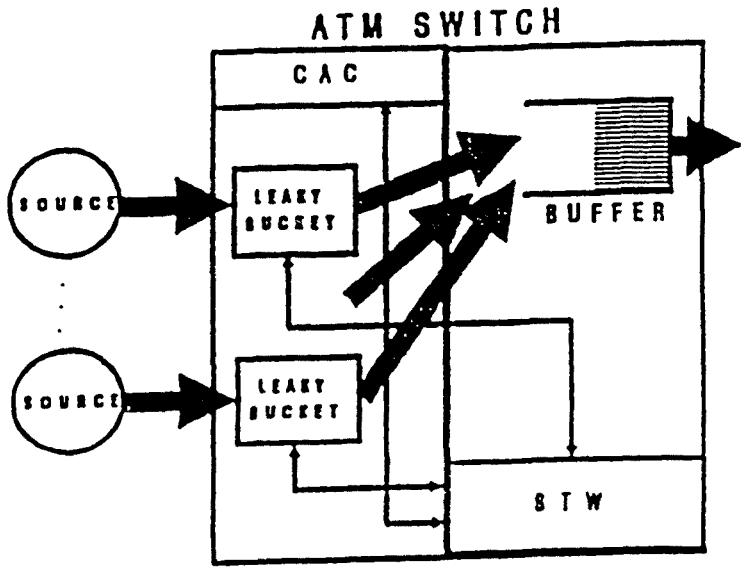

Figura H10. Entorno de simulación.

STW se activará cuando el contador del Leaky Bucket esté en situación de overflow. En este caso cuando una celda llegue al nodo, STW estudiará la posibilidad de que esa celda que la función de policía arrestaría, entre en la red. Para ello STW calculará el efecto en el QOS de las otras conexiones el hecho de que efectivamente esa celda entre. Como que consideraremos la tasa de pérdida de celdas como factor determinante del QOS, se considerará que el hecho de que entre una celda de esa conexión virtual, supone que se perderá una celda para el resto de conexiones virtuales.

Este es el fundamento básico en que está basado este procedimiento del STW en esta fase que estamos considerando. A pesar de que es un método muy restrictivo nos permite mantener la QOS de todas las conexiones dentro de los márgenes establecidos.

Por lo tanto el método consiste en ir controlando la tasa de error acumulada para cada circuito virtual. El límite fijado para esta simulación es de 10E-03. Esto quiere decir que sólo se permite perder una celda por cada mil en cada uno de los circuitos virtuales. De hecho este es el valor elegido para el dimensionamiento del bucket de la función de policía.

Podemos observar en la figura H11 siguiente la distribución de la ocupación del buffer, en donde se aprecia que este se llena completamente, y que en la última posición hay un incremento relativo de celdas,dado que existe rebasamiento de la cola. De esta figura , que corresponde al caso de rafagueo b = 5, se podría obtener en caso de que se deseara el valor del retardo y de la desviación de retardo para aplicar estos parámetros también en el STW, circunstancia que recordemos se ha desestimado.

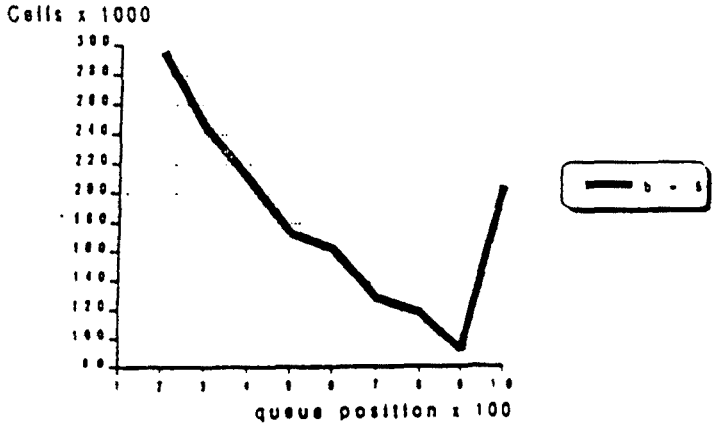

Figura H11. Distribución de ocupación del buffer

Para cada modelo de tráfico seleccionado se ha repetido la simulación del modelo STW propuesto con la función Leaky Bucket y se ha comparado sus resultados con la utilización exclusiva del leaky Bucket como función de policía. La simulaciones se han realizado en cada caso con diferente número de fuentes (carga).

La tasa de pérdida de celdas es la suma de las celdas perdidas en el Leaky Bucket y en el buffer del nodo de salida.

La figura H12 siguiente nos muestra los resultados comparados:

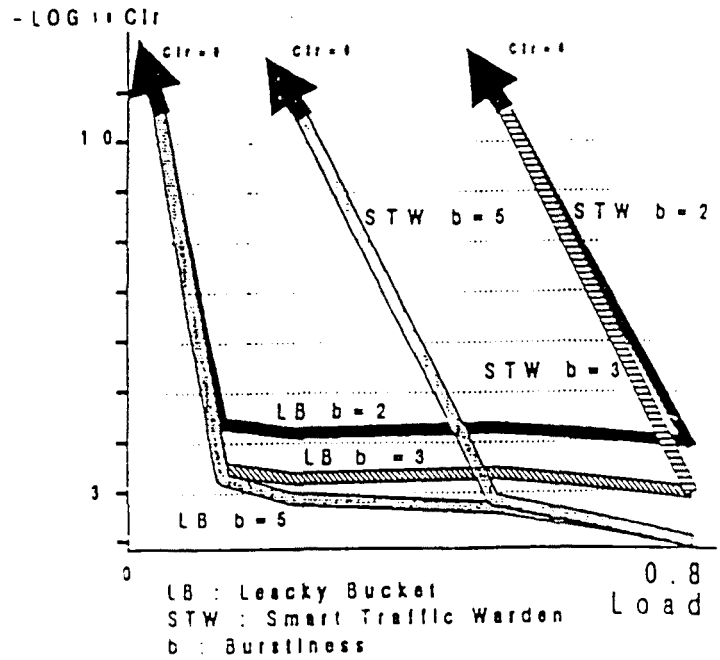

Figura H12. Resultados comparados para diferentes cargas

Podemos observar que los resultados son mejores para bajos rafagueos y pocas fuentes (carga). Cerca de la situación de congestión, el algoritmo STW no puede trabajar ya que no se puede modificar la velocidad de salida del LB sin alterar la QOS del resto de conexiones.

Una consecuencia importante es que los resultados no dependen exclusivamente del rafagueo y dependen principalmente de la carga. Con STW se elimina o reduce la influencia del rafagueo cuando la carga lo permite.

La conclusión mas importante que obtenemos de estos resultados es que la función de control de la congestión STW propuesta optimiza los recursos instantáneos disponibles manteniendo las condiciones generales de comportamiento de la función de policía,

pero no restringe el ancho de banda deseado por un usuario cuando es mas grande que el contratado, sino es necesario. Cuando un usuario sobrepasa el ancho de banda contratado, se hace sin perjudicar la QOS del resto.

Si se han elegido correctamente los parámetros de la función de policía, STW permite un mejor aprovechamiento de los recursos de la red sin congestión.

STW protege de la congestión y protege la QOS de los usuarios, eliminando celdas solo si es necesario, y salvaguardando los intereses de las compañias operadoras de las redes, sin prodedimientos adicionales de control de flujo (tickets, permits, etc,...) y sin introducir mayor complejidad en la gestión de los bufferes (virtual LB).

No obstante, tal como está planteado STW, es de muy difícil implementación para tasa de errores muy pequeñas, ya que entonces se reduce el campo de aplicación, al exigir un intervalo temporal de aplicación del STW muy grande, lo que lo haría ineficaz.

Manteniendo la misma idea original, pero variando la forma de aplicación, surge la tercera aproximación sobre el algoritmo que se explica a continuación:

H4.3 Control de la congestión con STW basado en la policía de nodo .

En esta propuesta se utilizarán dos versiones dependiendo de si STW se considera como complemento de una función de policía o bien como sistema de control de congestión independiente basado en una función de control de admisiones dinámica y continua y sin función de policía adicional.

H4.3.1 Propuesta 1 de STW

Se utilizará una función de policía tradicional como el Leaky Bucket y se dimensionará suponiendo que no hay jitter para una probabilidad de error determinada. Como que el jitter existe, el LB arrestará celdas inocentes pero se les dará una última oportunidad antes de eliminarlas, con el STW. Esta oportunidad será independiente de si las celdas son inocentes o culpables. El LB habrá cumplido su función, y si pueden pasar sin producir congestión, se les dejará pasar.

En el nodo habrá una función de policía que actuará sobre cada circuito virtual.

STW actuará de la siguiente forma:

Cuando una celda sea arrestada por el LB antes de descartarla se comprobará el hecho de que a pesar de que entre en el nodo no produzca congestión. Para ello se comprobará que el hecho de introducir esta celda en el multiplexor de salida del nodo no modifica la QOS del resto de VC de ese mismo port de salida.

Se considera que no hay congestión cuando no se pierden celdas por overflow en los buffers de los multiplexores de los ports de salida de los nodos correspondientes.

El funcionamiento de este caso será el siguiente:

A) Si varios VC necesitan activar STW en el mismo instante de tiempo se debe actuar de forma aleatoria para cada VC.

B) Cada VC tiene un contador T

C) Cada VC ha contratado una máxima Velocidad de pico Vp

D) T vale inicialmente cero T = 0

E) Cuando llega una celda el contador pasa a valer T = R donde R = Vt / Vp siendo Vt la velocidad del enlace de salida del multiplexor donde deben ir a parar las celdas.

Después de actualizar el contador al valor indicado anteriormente,este se va decrementado en uno por cada slot hasta cero.

En cualquier momento una celda arrestada por LB puede querer entrar. La posición teórica en la que entraría en el multiplexor de salida se denomina L. En este caso se supone que por el hecho de entrar esa celda arrestada en la posición L no debe hacer que hasta que salga del multiplexor pueda perderse alguna celda de otro VC. Para ello se debe calcular el máximo número de celdas que puede llegar por VC durante ese tiempo considerando su Vp (Será el caso peor teórico). Para ello se realiza el siguiente cálculo en todos los VC de ese mismo multiplexor:

$$\frac{L - T}{R} \text{ truncado} = S$$

Se aceptará la entrada si el sumatorio de S de todos los VC es mas pequeño que W-L

Este cálculo se puede ajustar más de la siguiente forma:

a) Para cada valor desde L = L hasta L = 1 se hace el calculo anterior y se aplica la operación AND.

Todos estos cálculos están hechos pensando que las Vp de los circuitos virtuales son las contratadas, pero el jitter hace que

no sea así. Esto quiere decir que se introduce una probabilidad de error. El cálculo analítico de esta probabilidad de error dependerá del tráfico que se estudie, y en todo caso se aplicará el método utilizado en el apartado E para el Leaky Bucket. Dada la diversidad de tráfico presente no se realiza este cálculo, en si muy complejo, y se basa el estudio de nuevo en la simulación por ordenador.

H4.3.2 Propuesta 2 de STW

Aquí STW no es un complemento de una función de policia sino que se convierte en una función de policia de valor de pico.

En este caso no se considera la actuación previa del LB y se deja actuar exclusivamente al STW. Esto quiere decir que cuando una celda no supera la Vp se deja siempre pasar. Se supone que el sistema de control de admisiones ya ha actuado y a aplicado su criterio para permitir el acceso de esa conexión. Sino cabe en el buffer se perderá. Cuando se supera la Vp se aplica el criterio STW del apartado anterior.

H4.3.3 Evaluación del STW (1 y 2).

Para el estudio de los dos algoritmos propuestos de STW se realizarán simulaciones por ordenador utilizando como tráfico de entrada el correspondiente a video real explicado en la Parte D.

Dimensionamiento del Leaky Bucket

La función de policia que se utilizará para el STW1 será el Leaky Bucket.

Para el dimensionamiento del Leaky Bucket se utilizará un método propuesto por esta tesis basado en la aproximación de la gráfica del Bucket en función del porcentaje de celdas arrestadas.

Hasta ahora esto se había hecho obteniendo por simulación y para un tráfico determinado una representación gráfica que indicaba el número de veces que el contador habia estado en el valor 1, en el valor 2, etc, de tal forma que luego al decidir el tamaño del bucket se debería perder todas aquellas celdas que se hubieran almacenado en las posiciones del bucket posteriores al valor seleccionado, lo que nos indicaría una tasa de error determinada.

Este método que en principio parece adecuado, no es válido ya que el comportamiento del contador no es el mismo si este es infinito, como correspondería al caso mencionado, que si es finito que correspondería al caso del Leaky Bucket después de haber elegido un valor para una probabilidad de error determinada.

El método que aquí se propone está derivado del anterior pero considerando la confección de la gráfica origen que permita el corte para una tasa de error determinada de una forma discreta.

De esta gráfica conocemos un detalle importante y es que su comportamiento es de crecimiento muy rápido . Idealmente, para encontrar esta gráfica en un cierto dominio, deberíamos tener tantos LB como valores posibles dentro del dominio, teniendo cada contador del LB un valor dentro del dominio. Esto sería muy lento incluso para simularlo por ordenador si el valor del dominio es alto.

Para ello se toma un valor pequeño del dominio, por ejemplo 6, y se calcula el porcentaje de celdas arrestadas para cada LB.

Ponemos los valores del dominio elegidos y los porcentajes de error en una gráfica y se unen los puntos resultantes por una recta, con el fin de conocer una aproximación de los valores intermedios.

Como que la gráfica es con crecimientos muy rápidos, los valores elegidos deben de estar también distanciados de una forma muy rápida (comportamiento del tipo exponencial).

Por ejemplo si el dominio es de 1 a 100 tomamos los valores 1, 7, 12, 25, 50 y 100. Estos valores corresponden a los límites del contador de 6 Leaky Buckets. Se ejecuta un programa de simulación del comportamiento del LB para un tráfico concreto, y para cada uno de los tamaños de Bucket indicados. Si algún LB arresta la celda, se incrementa el porcentaje correspondiente a ese LB. Con ello calculamos la tasa de error correspondiente a cada Leaky Bucket y la ubicamos en unas coordenadas cartesianas. Posteriormente encontramos la ecuación de la recta que une los puntos consecutivos. Se hace un bucle para todo el dominio, y según el intervalo requerido se aplica una u otra ecuación de recta, obteniendo así una tabla o una gráfica.

En la figura H13 siguiente se indica un ejemplo de gráfica en donde se observa el resultado correspondiente a una simulación realizada con el fin de obtener unos resultados indicativos del método utilizado.

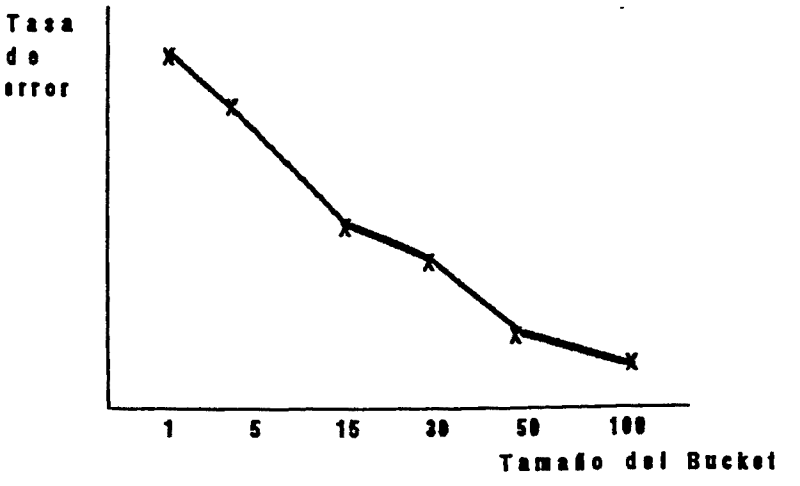

Figura H13. Ejemplo de dimensionamiento de bucket

Escenario de la evaluación

El escenario elegido de evaluación sigue la misma tónica que en los otros dos casos anteriores, y se indica en la figura H14 siguiente:

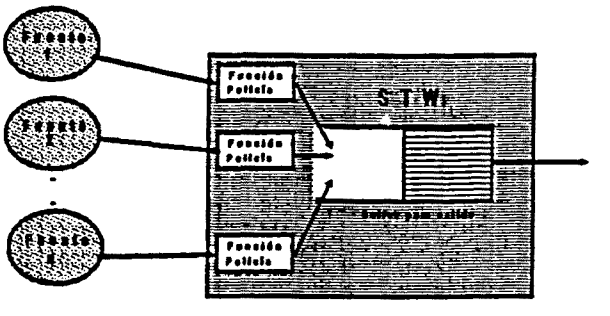

Figura H14. Escenario de evaluación.

Fases de evaluación

Las fases realizadas para la evaluación del algoritmo propuesto son las siguientes:

Fase 1. Corresponderá a la utilización de un tráfico con Distribución 1 y Distribución 2 (Parte D2) de la fuente real de vídeo. Se pretende estudiar el comportamiento del STW1 y STW2 frente al Leaky Bucket ante un tráfico sin jitter y con jitter desmesurado, con los parámetros del Leacky Bucket correctamente dimensionados para el caso sin jitter.

Fase 2. Corresponderá a la utilización de un porcentaje de fuentes con tráfico Distribución 1 y otro porcentaje con fuentes

con tráfico Distribución 2. Se pretende estudiar el comportamiento frente a la congestión del STW1 y STW2 frente al Leacky Bucket cuando existen fuentes infractoras o erróneas.

Fase 3. Corresponderá a la utilización de fuentes con tráfico Distribución 3, para observar el comportamiento del STW1 y STW2 frente al Leacky Bucket ante la presencia de un jitter razonable.

Fase 4. Corresponderá a la utilización de fuentes con tráfico de distribución 1 a la que se le introduce un porcentaje de fuentes de las denominadas "tricky users". Se estudiará la evolución de la congestión.

Fase 1.

Utilización del tráfico "Flower Garden" con diversas intensidades de tráfico y utilizando fuentes homogéneas. El objetivo de esta fase es comprobar el funcionamiento del STW en sus dos versiones y compararlo con los resultados que ofrece una función de policía convencional como es el Leaky Bucket.

Para ello se han realizado simulaciones con las siguientes condiciones:

Tráfico: Flower Garden 501

Fuentes: 4, 12, 18 y 22

Intensidad de tráfico: 0.16, 0.48, 0.73 y 0.88

Tamaño del Bucket: 5 celdas

Tamaño del Buffer: 10 celdas

Ciclos de simulación: 21 millones que corresponden a mas de dos periodos completos de la imagen elegida.

Las simulaciones se han repetido para cada una de las intensidades de tráfico indicadas.

Dado que la fuente es un codificador de video real, y como se ha explicado en la parte D, las celdas resultantes por trama se pueden distribuir dentro de ella de diversas formas. En estas simulaciones se ha diseñado el tamaño del bucket para una distribución de celdas dentro de la trama homogénea o sea distribuidas equidistantemente (distribución 1 Parte D.2). También se ha simulado con la distribución acumulada (distribución 2 Parte D.2) con el fin de comprobar el efecto con un jitter desmesurado.

Los resultados se pueden comprobar en la figura H15 que se indica a continuación.

Podemos observar la tendencia a mejora de utilización de los recursos que presentan los algoritmos STW respecto al caso tradicional del Leaky Bucket.

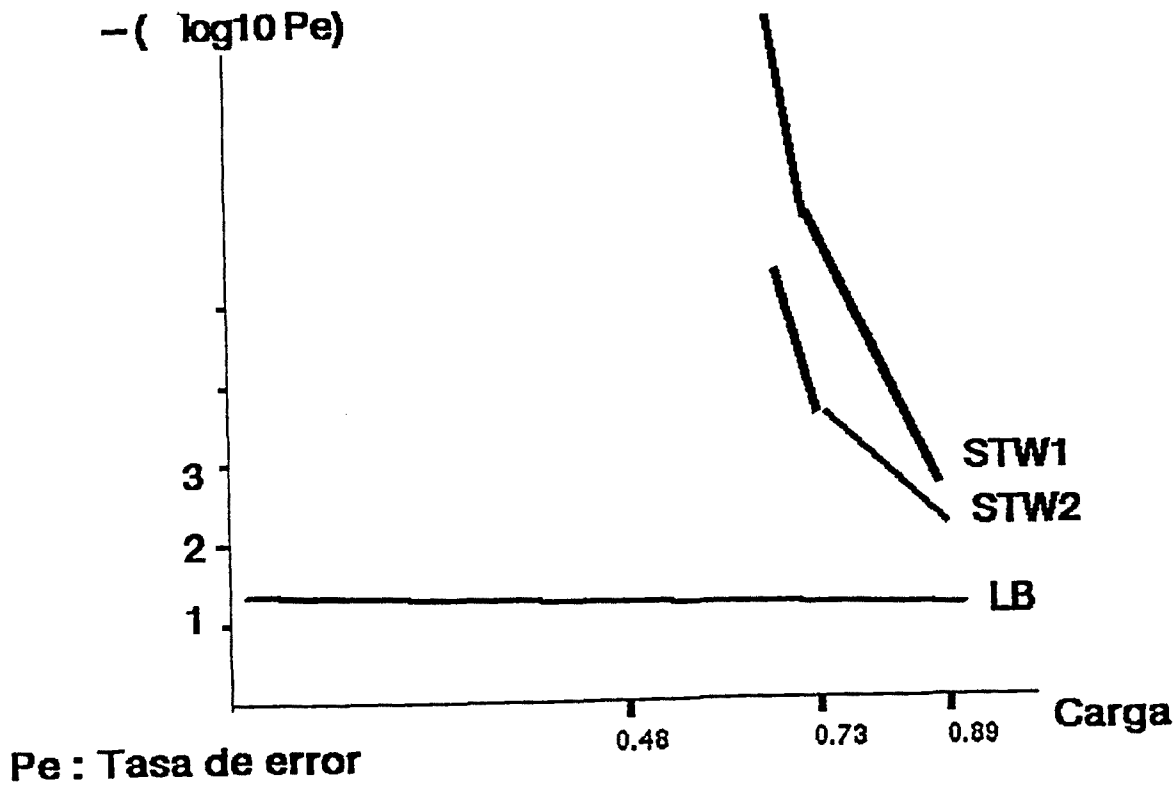

Figura H15. Resultados fase 1 y tráfico distribución 1.

Con el fin de poder apreciar el efecto que se está produciendo se indica a continuación los resultados fuente a fuente para el caso de mayor carga que se ha simulado.

Número de fuentes: 22
Número de ciclos: 7000000
Longitud máxima del buffer: 10
Intensidad de tráfico total: 8.97257E-01
STW activado: NO

Ocupación media del buffer: 2.21294E+00
Ocupación máxima del buffer: 10
Número total de celdas generadas: 6894694
Porcentaje de celdas perdidas: 4.31732E-02

FUENTE	EMITIDAS	ARRESTADAS	STW	MARCADAS	NO MARCADAS
0	312053	12642	12642	0	0
1	312574	12466	12466	0	0
2	313877	14269	14269	0	0
3	311730	12621	12621	0	0
4	311961	12405	12405	0	0
5	314942	15331	15331	0	0
6	311151	12298	12298	0	0
7	312850	12864	12864	0	0
8	311868	12237	12237	0	0
9	314496	13937	13937	0	0
10	315522	15556	15556	0	0
11	316234	15647	15647	0	0
12	314772	14102	14102	0	1
13	311727	12200	12200	0	0
14	312030	12376	12376	0	1
15	312625	12790	12790	0	0
16	314430	14554	14554	0	0
17	316050	15725	15725	0	0
18	311747	12512	12512	0	0
19	312727	12580	12580	0	0
20	314798	14196	14196	0	0
21	314530	14356	14356	0	0

Número de fuentes: 22
Número de ciclos: 7000000
Longitud máxima del buffer: 10
Intensidad de tráfico total: 8.97257E-01
STW activado: 1

Ocupación media del buffer: 2.49084E+00
Ocupación máxima del buffer: 10
Número total de celdas generadas: 6894694
Porcentaje de celdas perdidas: 2.32063E-06

FUENTE	EMITIDAS	ARRESTADAS	STW	MARCADAS	NO MARCADAS
0	312053	12642	1	0	0
1	312574	12466	1	0	0
2	313877	14269	0	0	0
3	311730	12621	1	0	0
4	311961	12405	0	0	0
5	314942	15331	0	0	0
6	311151	12298	0	0	0
7	312850	12864	0	0	0
8	311868	12237	0	0	0
9	314496	13937	1	0	1
10	315522	15556	2	0	0
11	316234	15647	0	0	0
12	314772	14102	2	0	1
13	311727	12200	0	0	0
14	312030	12376	0	0	1
15	312625	12790	0	0	0
16	314430	14554	3	0	0
17	316050	15725	0	0	0
18	311747	12512	0	0	0
19	312727	12580	0	0	0
20	314798	14196	2	0	0
21	314530	14356	0	0	0

Número de fuentes: 22
Número de ciclos: 7000000
Longitud máxima del buffer: 10
Intensidad de tráfico total: 8.97257E-01
STW activado: 2

Ocupación media del buffer: 2.49052E+00
Ocupación máxima del buffer: 9
Número total de celdas generadas: 6894694
Porcentaje de celdas perdidas: 2.11757E-05

FUENTE	EMITIDAS	ARRESTADAS	STW	MARCADAS	NO MARCADAS
0	312053	0	12	0	0
1	312574	0	2	0	0
2	313877	0	10	0	0
3	311730	0	4	0	0
4	311961	0	6	0	0
5	314942	0	12	0	0
6	311151	0	5	0	0
7	312850	0	6	0	0
8	311868	0	6	0	0
9	314496	0	11	0	0
10	315522	0	7	0	0
11	316234	0	6	0	0
12	314772	0	6	0	0
13	311727	0	2	0	0
14	312030	0	4	0	0
15	312625	0	3	0	0
16	314430	0	6	0	0
17	316050	0	10	0	0
18	311747	0	4	0	0
19	312727	0	9	0	0
20	314798	0	12	0	0
21	314530	0	3	0	0

Para apreciar el efecto beneficioso que produce el STW en sus dos versiones, se han repetido las simulaciones para una distribución de celdas no homogénea dentro de la trama, sino concentradas al principio de la misma, de tal forma que se configura un tráfico mucho mas agresivo. Esto se ha medido sin variar los parámetros anteriormente establecidos en el Leaky Bucket, con el fin de comprobar el comportamiento del STW. Se supone que la fuente envía información de una forma no pactada en el contrato en cuanto se refiere al rafagueo, aunque si en sus valores medios. Esto no tiene porque ser necesariamente un usuario malicioso, sino que puede ser un jitter extramadamente acumulado.

Los resultados los podemos comprobar en la figura H16 que se indica a continuación.

Podemos observar el efecto beneficioso debido a la utilización del algoritmo STW en sus dos versiones, tanto mas como menor es la carga, tal como corresponde a la definición propuesta del algoritmo STW.

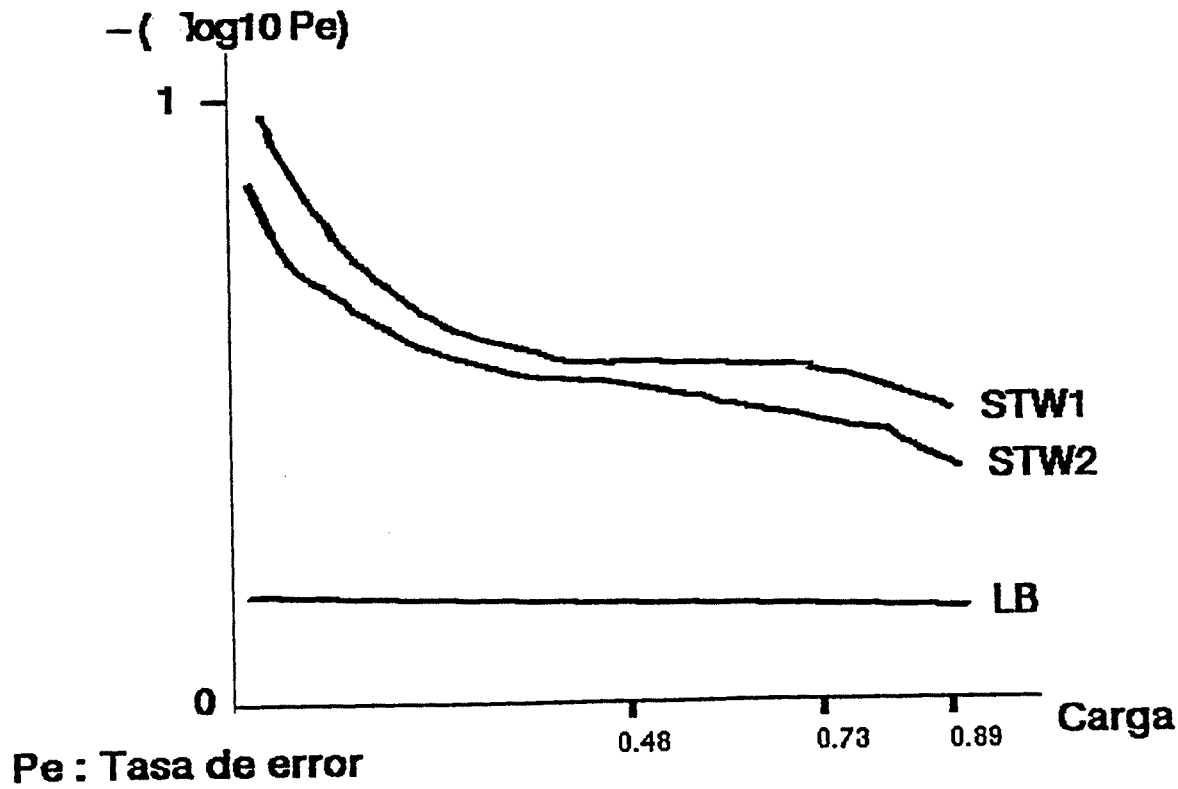

Figura H16. Resultados fase 1 tráfico distribución 2.

Con el fin de poder estudiar fuente a fuente los resultados, se indican a continuación los correspondientes a una carga de 18 fuentes.

Número de fuentes: 18
Número de ciclos: 21000000
Longitud máxima del buffer: 10
Intensidad de tráfico total: 7.34119E-01
STW activado: NO

Ocupación media del buffer: 4.84544E-02
Ocupación máxima del buffer: 3
Número total de celdas generadas: 16935268
Porcentaje de celdas perdidas: 9.55697E-01

FUENTE	EMITIDAS	ARRESTADAS	STW	MARCADAS	NO MARCADAS
0	937445	895908	895908	0	0
1	938497	896915	896915	0	0
2	939627	897990	897990	0	0
3	937793	896243	896243	0	0
4	939900	898263	898263	0	0
5	941006	899325	899325	0	0
6	940712	899036	899036	0	0
7	939688	898047	898047	0	0
8	937109	895584	895584	0	0
9	944925	903063	903063	0	0
10	942208	900483	900483	0	0
11	943366	901573	901573	0	0
12	946403	904492	904492	0	0
13	940299	898640	898640	0	0
14	941916	900186	900186	0	0
15	942197	900454	900454	0	0
16	940113	898459	898459	0	0
17	942064	900324	900324	0	0

Número de fuentes: 18
Número de ciclos: 21000000
Longitud máxima del buffer: 10
Intensidad de tráfico total: 7.34119E-01
STW activado: 1

Ocupación media del buffer: 3.20149E+00
Ocupación máxima del buffer: 10
Número total de celdas generadas: 16935266
Porcentaje de celdas perdidas: 3.09635E-01

FUENTE	EMITIDAS	ARRESTADAS	STW	MARCADAS	NO MARCADAS
0	937445	895908	144227	0	3971
1	938497	896915	170387	0	372
2	939627	897990	127150	0	805
3	937793	896243	484542	0	944
4	939900	898263	661197	0	12582
5	941006	899325	341384	0	789
6	940712	899036	803250	0	9553
7	939688	898047	263513	0	2622
8	937109	895584	33681	0	0
9	944925	903063	684775	0	3205
10	942208	900483	204116	0	0
11	943366	901573	83267	0	235
12	946403	904492	757501	0	13556
13	940299	898640	16348	0	0
14	941915	900186	33660	0	0
15	942196	900454	335538	0	0
16	940113	898459	0	0	0
17	942064	900324	50574	0	0

Número de fuentes: 18
Número de ciclos: 21000000
Longitud máxima del buffer: 10
Intensidad de tráfico total: 7.34119E-01
STW activado: 2

Ocupación media del buffer: 3.18130E+00
Ocupación máxima del buffer: 9
Número total de celdas generadas: 16935266
Porcentaje de celdas perdidas: 3.09635E-01

FUENTE	EMITIDAS	ARRESTADAS	STW	MARCADAS	NO MARCADAS
0	937445	0	139546	0	0
1	938497	0	170282	0	0
2	939627	0	93769	0	0
3	937793	0	503435	0	0
4	939900	0	690016	0	0
5	941006	0	329350	0	0
6	940712	0	834278	0	0
7	939688	0	243020	0	0
8	937109	0	26315	0	0
9	944925	0	706707	0	0
10	942208	0	211482	0	0
11	943366	0	77972	0	0
12	946403	0	781452	0	0
13	940299	0	14843	0	0
14	941915	0	20568	0	0
15	942196	0	348630	0	0
16	940113	0	0	0	0
17	942064	0	52079	0	0

Conclusiones de la fase 1

La conclusión mas importante que obtenemos de la aplicación de STW en sus dos versiones es que evita la pérdida de celdas aunque la función de policía haya actuado sobre ellas, siempre que ello no suponga la disminución de la calidad de servicio contratada para el conjunto de conexiones que comparten un nodo.

Esto quiere decir que STW mejorará la utilización de los recursos disponibles sin congestión, dando un margen importante en el diseño de los parámetros de la función de policía ,con el fin de absorver las posibles irregularidades de llegada del tráfico debido al efecto jitter.

Otra conclusión que obtenemos es que STW1 se comporta practicamente igual que STW2 para tráficos que no permitan el vaciado completo del buffer del nodo, con lo cual podemos afirmar que STW es una función de policía en sí, y no solo como complemento de otras funciones de policía como el Leaky Bucket, para estos casos.

Para los casos de tráficos que sí permiten el vaciado del nodo, STW1 se comporta mejor que STW2, aunque en ambos casos los resultados son muy satisfactorios, respecto a la mejora en la utilización de recursos sin congestión.

STW2, y dada su definición, no tiene margen de maniobra ya que no dispone de parámetros que sean manipulables excepto la propia velocidad de pico, por lo que pensamos que no será útil su utilización, y que dadas las circunstancias , haya casos en los que se comporte de forma insatisfactoria respecto a la calidad de servicio contratada.

Fase 2.

Del tráfico anterior se eligen una intensidad de tráfico y se introduce el concepto de tráfico no homogéneo en el sentido de que se variará en algunas fuentes la distribución de las celdas en la trama. Habrá un 10 % de fuentes que tendrán el tráfico intenso acumulando las celdas al principio de la trama (distribución 2), y un 90 % con tráfico distribuido a lo largo de la trama (distribución 1). Todo ello manteniendo los parámetros del Leaky Bucket correspondientes al tráfico distribuido.

El objetivo de esta fase , es analizar el comportamiento del sistema cuando alguna de las fuentes viola las condiciones del contrato o tiene un jitter excesivo, y comprobar la evolución de la congestión global.

Tráfico: Flower Garden 501
Fuentes: 6, 18 y 22
Intensidad de tráfico: 0.24, 0.73 y 0.89
Tamaño del Bucket: 5 celdas
Tamaño del Buffer: 10 celdas
Ciclos de simulación: 7 millones que corresponden a mas de un periodo completos de la imagen elegida.

Los resultados globales de comportamiento de la red se indican a continuación, teniendo en cuenta que a las fuentes infractoras o descuidadas se les ha cargado un porcentaje muy importante de las celdas arrestadas, comportándose el algoritmo STW en general como efectivo en el aprovechamiento de los recursos, y en reducir al máximo el impacto debido a las fuentes de tráfico agresivo y no previsto.

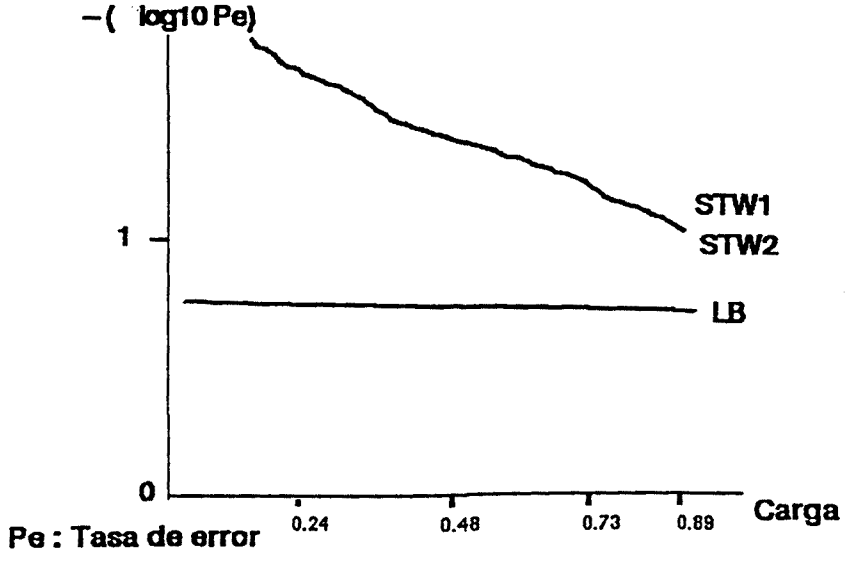

Figura H17. Resultados fase 2.

Con el fin de poder observar los resultados fuente a fuente se indican los resultados completos para el caso de mayor carga 0.89.

Número de fuentes: 18
Número de ciclos: 7000000
Longitud máxima del buffer: 10
Intensidad de tráfico total: 7.34119E-01
STW activado: NO

Ocupación media del buffer: 1.10496E+00
Ocupación máxima del buffer: 10
Número total de celdas generadas: 5641889
Porcentaje de celdas perdidas: 1.44789E-01

FUENTE	EMITIDAS	ARRESTADAS	STW	MARCADAS	NO MARCADAS
0	312842	298979	298979	0	0
1	312574	12466	12466	0	0
2	313877	14269	14269	0	0
3	311730	12621	12621	0	0
4	311961	12405	12405	0	0
5	314942	15331	15331	0	0
6	311152	12298	12298	0	0
7	312850	12864	12864	0	0
8	311868	12237	12237	0	0
9	314496	13937	13937	0	0
10	315522	15556	15556	0	0
11	316234	15647	15647	0	0
12	314772	14102	14102	0	0
13	311727	12200	12200	0	0
14	312030	12376	12376	0	0
15	312625	12790	12790	0	0
16	314430	14554	14554	0	0
17	316257	302254	302254	0	0

Número de fuentes: 18
Número de ciclos: 7000000
Longitud máxima del buffer: 10
Intensidad de tráfico total: 7.34119E-01
STW activado: 1

Ocupación media del buffer: 1.91991E+00
Ocupación máxima del buffer: 10
Número total de celdas generadas: 5641889
Porcentaje de celdas perdidas: 7.22602E-02

FUENTE	EMITIDAS	ARRESTADAS	STW	MARCADAS	NO MARCADAS
0	312842	298979	168022	0	1040
1	312574	12466	1046	0	2328
2	313877	14269	1183	0	3287
3	311730	12621	914	0	3100
4	311961	12405	859	0	3107
5	314942	15331	850	0	3504
6	311152	12298	616	0	4131
7	312850	12864	800	0	4044
8	311868	12237	716	0	4123
9	314496	13937	834	0	4774
10	315522	15556	884	0	4459
11	316234	15647	822	0	4219
12	314772	14102	767	0	4707
13	311727	12200	332	0	2418
14	312030	12376	577	0	2978
15	312625	12790	504	0	3914
16	314430	14554	739	0	3052
17	316257	302254	166970	0	1064

Número de fuentes: 18
Número de ciclos: 7000000
Longitud máxima del buffer: 10
Intensidad de tráfico total: 7.34119E-01
STW activado: 2

Ocupación media del buffer: 1.88591E+00
Ocupación máxima del buffer: 9
Número total de celdas generadas: 5641889
Porcentaje de celdas perdidas: 7.23155E-02

FUENTE	EMITIDAS	ARRESTADAS	STW	MARCADAS	NO MARCADAS
0	312842	0	82048	0	0
1	312574	0	24227	0	0
2	313877	0	23776	0	0
3	311730	0	19730	0	0
4	311961	0	18836	0	0
5	314942	0	18121	0	0
6	311152	0	18008	0	0
7	312850	0	16799	0	0
8	311868	0	14900	0	0
9	314496	0	15363	0	0
10	315522	0	12971	0	0
11	316234	0	12310	0	0
12	314772	0	12012	0	0
13	311727	0	6354	0	0
14	312030	0	8036	0	0
15	312625	0	10203	0	0
16	314430	0	7288	0	0
17	316257	0	87014	0	0

Conclusiones de la Fase 2

Las conclusiones que obtenemos de la fase 2 son las siguientes:

1. El hecho de que varias fuentes envien un tráfico no pactado , sin jitter, no supone un aumento de la congestión, ya que el leaky Bucket actúa impidiendo el paso de aquellas celdas que podrían perturbar la congestión.

2. La aplicación del algoritmo propuesto STW1 mejora considerablemente la tasa de error conseguida en general, tanto para las fuentes infractoras como para el resto. En este caso las fuentes menos beneficiadas son las infractoras, por lo que podemos afirmar que el comportamiento de STW1 es satisfactorio en el conjunto de la simulación.

3. La aplicación del algoritmo propuesto STW2 mejora igualmente la tasa de error conseguida general, ahora bien trata a todas las fuentes de una forma "democrática" por lo que todas salen beneficiadas por igual, perjudicando al fin la calidad de servicio individual, respecto al STW1, de las fuentes que no han modificado su tráfico. Por ello fuentes que no han modificado su tráfico respecto al contratado ven perjudicada su calidad de servicio respecto al Leaky Bucket cuando la carga es elevada, no así con carga baja.

Por ello se piensa que este método no se comporta de una forma tan satisfactoria como el STW1. Por otro lado presenta una difícil parametrización para actuar sobre sus efectos, ya que implica modificar la velocidad de pico esperada.

Estos resultados nos indican que con carga elevada STW2 no debe ser utilizado como función de policía.

Fase 3

En esta fase se va a considerar el efecto jitter. Para ello se selecciona la distribución 3 explicada en la parte D2.

tráfico "flower Garden"

carga de 0.73 con 18 fuentes

7 millones de ciclos de simulación

bucket de 5

buffer de 10 celdas.

A continuación se indica la figura H18 en donde se puede ver el comportamiento general del sistema con estas condiciones.

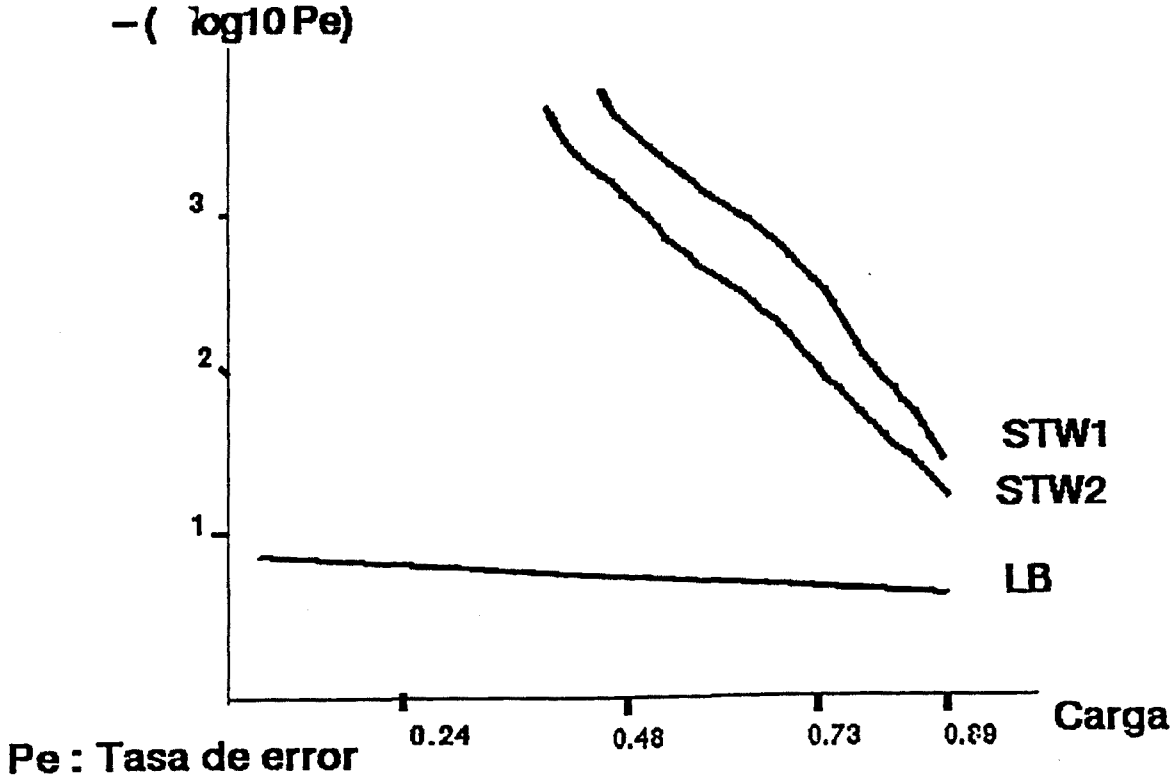

Figura H18. Resultados fase 3.

Los resultados obtenidos se indican a continuación fuente a fuente.

Número de fuentes: 18
Número de ciclos: 7000000
Longitud máxima del buffer: 10
Intensidad de tráfico total: 7.34119E-01
STW activado: NO

Ocupación media del buffer: 1.33967E+00
Ocupación máxima del buffer: 10
Número total de celdas generadas: 5640920
Porcentaje de celdas perdidas: 1.18318E-01

FUENTE	EMITIDAS	ARRESTADAS	STW	MARCADAS	NO MARCADAS
0	312074	36761	36761	0	19
1	312574	36877	36877	0	30
2	313840	37726	37726	0	21
3	311730	36025	36025	0	22
4	311953	36168	36168	0	23
5	314934	38785	38785	0	22
6	311161	35843	35843	0	19
7	312825	36352	36352	0	15
8	311882	36034	36034	0	18
9	314495	37367	37367	0	19
10	315517	38669	38669	0	17
11	316238	38409	38409	0	16
12	314779	37241	37241	0	18
13	311741	36113	36113	0	35
14	312038	35933	35933	0	19
15	312675	36439	36439	0	17
16	314424	37246	37246	0	12
17	316040	39083	39083	0	12

Número de fuentes: 18
Número de ciclos: 7000000
Longitud máxima del buffer: 10
Intensidad de tráfico total: 7.34119E-01
STW activado: 1

Ocupación media del buffer: 1.91282E+00
Ocupación máxima del buffer: 10
Número total de celdas generadas: 5640920
Porcentaje de celdas perdidas: 1.38470E-03

FUENTE	EMITIDAS	ARRESTADAS	STW	MARCADAS	NO MARCADAS
0	312074	36761	267	0	180
1	312574	36877	273	0	192
2	313840	37726	283	0	194
3	311730	36025	261	0	174
4	311953	36168	254	0	163
5	314934	38785	279	0	152
6	311161	35843	252	0	177
7	312825	36352	239	0	163
8	311882	36034	265	0	173
9	314495	37367	272	0	151
10	315517	38669	257	0	158
11	316238	38409	254	0	172
12	314779	37241	250	0	166
13	311741	36113	253	0	170
14	312038	35933	270	0	161
15	312675	36439	266	0	175
16	314424	37246	262	0	176
17	316040	39083	296	0	161

Número de fuentes: 18
Número de ciclos: 7000000
Longitud máxima del buffer: 10
Intensidad de tráfico total: 7.34119E-01
STW activado: 2

Ocupación media del buffer: 1.89910E+00
Ocupación máxima del buffer: 9
Número total de celdas generadas: 5640920
Porcentaje de celdas perdidas: 1.91174E-03

FUENTE	EMITIDAS	ARRESTADAS	STW	MARCADAS	NO MARCADAS
0	312074	0	595	0	0
1	312574	0	627	0	0
2	313840	0	613	0	0
3	311730	0	601	0	0
4	311953	0	586	0	0
5	314934	0	591	0	0
6	311161	0	586	0	0
7	312825	0	584	0	0
8	311882	0	609	0	0
9	314495	0	606	0	0
10	315517	0	575	0	0
11	316238	0	552	0	0
12	314779	0	598	0	0
13	311741	0	597	0	0
14	312038	0	584	0	0
15	312675	0	643	0	0
16	314424	0	628	0	0
17	316040	0	609	0	0

Conclusiones Fase 3.

1. El comportamiento del Leaky Bucket convencional dimensionado para una determinada calidad de servicio sin jitter, empeora el resultado de la tasa de error en todos los casos con presencia de jitter. Esto nos confirma la parte G1.

2. STW1 mejora de una forma considerable la utilización de los recursos aún en presencia de jitter, disminuyendo enormemente la tasa de error en cada uno de los circuitos virtuales. Esto nos confirma la posible utilización de STW1 en todos los puntos de la red en donde exista multiplexación, y no solo en las puertas de entrada a la red, como correspondería a las simulaciones de las Fase 1 y 2.

3. STW2 mejora considerablemente los resultados respecto al Leaky Bucket, aunque no en la misma medida que el STW1.

4. Se podrían obtener los mismos resultados que con el STW1 aumentando el tamaño del Bucket del Leaky Bucket, pero esto implicaría sin lugar a dudas un aumento de la congestión, ya que entrarian en todo caso muchas mas celdas en el buffer que las calculadas para la calidad de servicio que se deseaba ofrecer en el caso de no haber Jitter.

Fase 4

En esta fase se va a considerar la presencia de "tricky users". Para ello se hará una modificación en el Leaky Bucket introduciendo el "safety margin" o margen de seguridad, que recordemos consistía en aumentar el leak por encima del valor medio contratado en el Leaky Bucket, que nos va a permitir realizar con el tráfico "Flower garden" un tricky user. Incrementando un % el leak, se permite que con un tamaño de bucket de 2 se pueda engañar al Leaky Bucket enviando 2 celdas seguidas y no enviando nada el tiempo justo para que el bucket se vacíe, y todo ello permitiendo que en el tiempo de trama se puedan enviar casi todas la celdas, considerando el peor caso o la trama con mayor intensidad de tráfico.

El hecho de no poder enviar todas las celdas de la trama debido a esta modificación, no tiene en estas pruebas mayor importancia, y se ha estimado hacerlo así dado que se pretende observar la tendencia del resultado, y no un cálculo exacto.

Se supondrá un 10% de "tricky" frente a un 90% de fuentes que cumplen su contrato.

Se elige el tráfico correspondiente a 22 fuentes, que es una carga elevada, ya que éste produce congestión en el buffer.

Los resultados son los siguientes:

Número de fuentes: 22
Número de ciclos: 7000000
Longitud máxima del buffer: 10
Intensidad de tráfico total: 8.97257E-01
STW activado: NO

Ocupación media del buffer: 2.63505E+00
Ocupación máxima del buffer: 10
Número total de celdas generadas: 6874370
Porcentaje de celdas perdidas: 5.14534E-03

FUENTE	EMITIDAS	ARRESTADAS	STW
0	306240	0	0
1	312574	1613	1613
2	313877	2371	2371
3	311730	1617	1617
4	311961	1613	1613
5	314942	2694	2694
6	311151	1613	1613
7	312850	1616	1616
8	311868	1611	1611
9	314496	1613	1613
10	307787	0	0
11	316234	2583	2583
12	314772	1611	1611
13	311727	1612	1612
14	312030	1613	1613
15	312625	1616	1616
16	314430	2429	2429
17	316050	2679	2679
18	311747	1613	1613
19	312727	1613	1613
20	314798	1615	1615
21	307754	0	0

```
---------------------------------------------------------------
Número de fuentes:  22
Número de ciclos:   7000000
Longitud máxima del buffer: 10
Intensidad de tráfico total:  8.97257E-01
STW activado:  1
---------------------------------------------------------------

Ocupación media del buffer:  2.68527E+00
Ocupación máxima del buffer: 10
Número total de celdas generadas:    6874370
Porcentaje de celdas perdidas:  4.50950E-06

---------------------------------------------------------------
```

FUENTE	EMITIDAS	ARRESTADAS	STW
0	306240	0	0
1	312574	1613	0
2	313877	2371	0
3	311730	1617	0
4	311961	1613	0
5	314942	2694	1
6	311151	1613	0
7	312850	1616	2
8	311868	1611	0
9	314496	1613	0
10	307787	0	0
11	316234	2583	0
12	314772	1611	0
13	311727	1612	0
14	312030	1613	0
15	312625	1616	0
16	314430	2429	0
17	316050	2679	0
18	311747	1613	0
19	312727	1613	0
20	314798	1615	0
21	307754	0	0

```
---------------------------------------------------------------
```

```
-------------------------------------------------------------
Número de fuentes:  22
Número de ciclos:   7000000
Longitud máxima del buffer:         10
Intensidad de tráfico total:  8.97257E-01
STW activado:  2
-------------------------------------------------------------

Ocupación media del buffer:  2.68444E+00
Ocupación máxima del buffer: 9
Número total de celdas generadas:     6874370
Porcentaje de celdas perdidas:  3.86945E-05

-------------------------------------------------------------
```

FUENTE	EMITIDAS	ARRESTADAS	STW
0	306240	0	9
1	312574	0	9
2	313877	0	17
3	311730	0	11
4	311961	0	12
5	314942	0	19
6	311151	0	23
7	312850	0	12
8	311868	0	10
9	314496	0	10
10	307787	0	21
11	316234	0	8
12	314772	0	9
13	311727	0	2
14	312030	0	10
15	312625	0	10
16	314430	0	9
17	316050	0	15
18	311747	0	3
19	312727	0	5
20	314798	0	15
21	307754	0	27

```
-------------------------------------------------------------
```

Conclusiones Fase 4

Las conclusiones que obtenemos de esta fase son las siguientes:

1. El Leaky Bucket se comporta correctamente ante el tricky user, dado que no se ha considerado jitter, y elimina todas aquellas celdas que superan las condiciones del contrato. En condiciones con jitter y según se ha visto en la Parte G2, puede en determinadas circunstancias aumentar la congestión global debido a este efecto. Recordemos que se concluía que el efecto de congestión era manejable dependiendo de la carga total y en base al tamaño del buffer.

2. El STW1 no puede trabajar en las fuentes "tricky" dado las características de diseño de dicho algoritmo. En este caso STW1 no aporta nada.

Recordemos que tal como se indicaba en la parte G2, la red podía detectar de una manera razonable la existencia de "tricky users" y actuar sobre ellos. Esta detección se podría incluir como una función más del STW1, ahora bien, creemos que esto no es necesario, por las razones expuestas en la parte G2.

3. STW1 si que mejora los resultados de utilización de recursos en el resto de las fuentes que no son "tricky", tal como se había indicado ya en las fases precedentes.

4. STW2 mejora globalmente el resultado de tasa de error y de aprovechamiento de recursos, tanto para las fuentes "tricky" como para las cumplidoras del contrato. Siguiendo con el concepto de "democrático", STW2 trata por igual a todas las fuentes, por lo que digamos que las fuentes infractoras no perjudican los resultados de las demás, y ellas no salen tan beneficiadas de su infracción.

Conclusiones Parte H.

STW1 se muestra como un complemento muy adecuado para las funciones de policía del tipo Leaky Bucket, con el fin de aprovechar de una forma mas efectiva los recursos de la red. Esto nos permite diseñar los parámetros del leaky Bucket con mayor tranquilidad, ya que podemos hacerlo considerando como si no existiera jitter, y tanto si éste se produce como si no, STW1 se encargará de ajustar el resultado lo más posible a la calidad de servicio contratada sin producir congestión.

La utilización de STW1 es tanto más efectiva cuanto menor sea la carga que presenta la red, pudiéndose aplicar no obstante en todo caso, aunque sus resultados no serán tan beneficiosos cuando haya mucha carga. En ninguna situación, la aplicación de STW1 será contraproducente para el mantenimiento de la Calidad de servicio contratada por todas las fuentes.

Por las razones indicadas se considera a STW1 como un complemento de la función Leaky Bucket muy adecuado para la mejor utilización de los recursos disponibles sin alterar la calidad de servicio de todas las fuentes que comparten los recursos y sin aumentar la congestión que de por sí pueda permitir el leaky Bucket.

STW2 se muestra como una función de policía efectiva para los casos generales, y sólo ante situaciones críticas de mucha carga y parcialmente infractora en la violación de la velocidad de pico contratada, se muestra como contraproducente en el mantenimiento de la calidad de servicio de las fuentes no infractoras. Esto es debido a su caracter "democrático".

Por otro lado STW2, no se muestra suficientemente flexible dado que no ofrece parámetros que permitan modificar su comportamiento, excepto la velocidad de pico.

Por todo ello se considera que STW2 no es una solución de función de policía generalmente aplicable.

La conclusión general que se obtiene de esta parte es que "la vigilancia inteligente" es un complemento adecuado a las funciones de policía, proporcionanado aquella visión global (distribuida) de la congestión que las funciones convencionales no tienen.

Parte I. Implementación física de la función de Policía

Esta parte explicará cuales son las implementaciones tecnológicas que permiten realizar desde un punto de vista hardware las funciones de policía estudiadas en esta tesis.

Se indican ejemplos de los diseños correspondientes a las funciones tradicionales tales como el Leaky Bucket y las funciones de ventana, las necesidades hardware apuntadas en algunos trabajos para los protocolos FRP/DT, y el estudio correspondiente al posible diseño e implementación de la función STW.

En lo que concierne al espaciamiento de celdas, aún no se ha publicado ningún trabajo que trate sobre este tipo de problemática.

I1. Leaky Bucket

Se presenta un ejemplo de diseño de un chip concreto que realize la función de policía Leaky Bucket la cual deberá cumplir los siguientes requisitos [COR91].

Bucket size (tamaño del bucket) : BS
Splash amount (ritmo de entrada): S
Leak rate (ritmo de salida): L
Bucket ocupado: BL

En la figura I1 se indican estos parámetros.

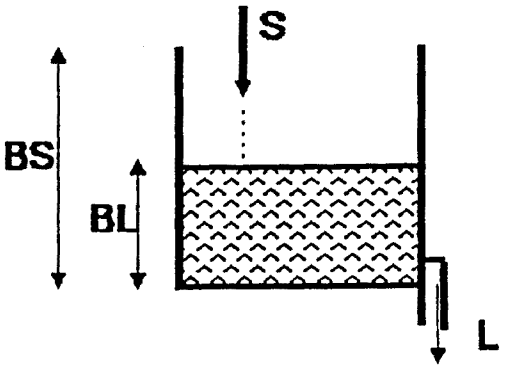

Figura I1. Parámetros del Leaky Bucket.

Los criterios de diseño adoptados serán los siguientes:

Average cell rate C = L / S

Burtiness B = BS / (S-L)

La función que deberá realizar el chip será:

BL := BL - ßt.L + S

donde ßt ("elapsed time") es el tiempo desde la última celda transferida tal como se indica en la figura I2.

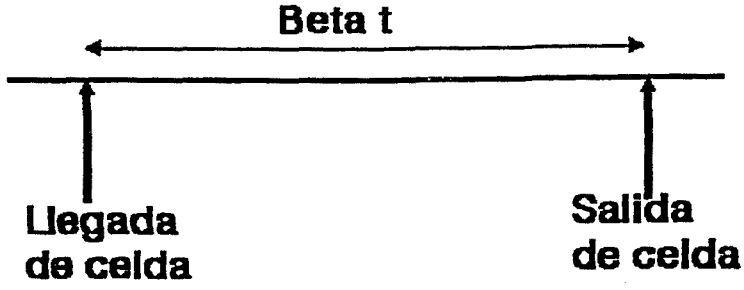

Figura I2. Procedimiento trasferencia de celdas

Se producirá una violación del contrato cuando BL > BS

Los requerimientos para el diseño especificados en el caso que se presenta son:

1. Velocidad media desde 16 Kbps hasta 622.08 Mbps
2. 2.5 10E-05 ≤ C ≤ 1
3. Resolución 1%
4. Velocidad de cálculo < 300 ns

Para ver los requerimientos hardware de la puesta en marcha de un sistema de estas características debemos ver el número de bits necesario para cada parámetro.

* Elapsed time:

ßt = Momento de llegada - Momento de la última transferencia
↓ ↓
contador slots almacenado en memoria

Necesitaremos 1023 indices para indicar que cada bucket se actualiza por lo menos una vez cada 1023 slots, lo que quiere decir que ßt necesitará 10 bits.

* Leak rate será para valores de potencia de dos por lo que solo necesitamos definir el exponente. Se eligen 16 valores

diferentes desde -8 hasta 7, lo que nos da 4 bits.

* Splash amount. Se elige un valor de 0...255, por lo que se necesitan 8 bits.

* Bucket size. Si el rafagueo elegido ha de ser ≤ 32

$$B = \frac{BS}{S - L} \leq 32$$

$$BS \leq 32.\ (S-L) \leq 32.\ Smax \leq 2^{13}$$

Esto nos indica un tamaño de 13 bits.

En resumen tenemos los siguientes tamaños:

Splash amount	:	8 bits
Leak Rate	:	4 bits
Bucket size	:	13 bits
Bucket level	:	13 bits
Memoria última transferencia	:	10 bits
Total	:	48 bits

El esquema general hardware del sistema será el que se indica a continuación en la figura I4:

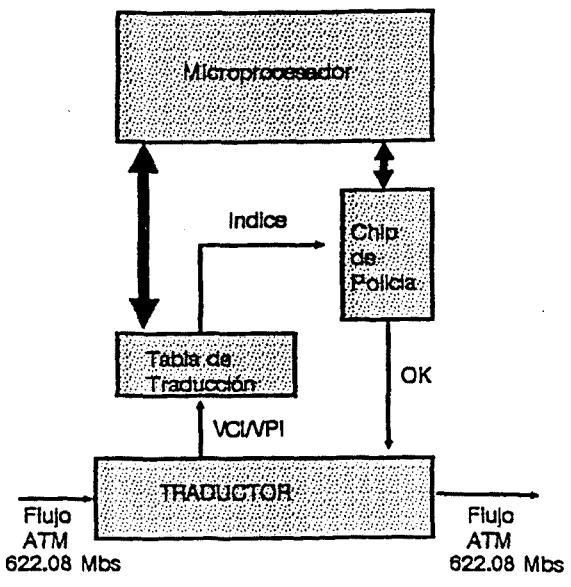

Figura I4. Esquema general hardware del sistema

Todo esto nos conduce a un chip con las siguientes características:

Tecnología	: ES2 1.2 μm CMOS
Tamaño	: 92.67 mm^2
Almacenamiento	: 84-pin PLCC
Frequencia reloj	: 25 Mhz.
Consumo	: 0.6 W (a 25 Mhz)
Velocidad ATM	: 622.08 Mbps.
Complejidad	: 320 K transistores
	290 K transistores SRAM
	30 K transistores lógica PCC

I2. Funciones de ventana

En el chip anterior se puede implementar de una manera muy fácil las funciones de ventana sin modificaciones aparentes de diseño [COR91].

I3. Protocolos FRP/DT

Desde un punto de vista funcional el FRP/DT formará parte del sistema de señalización de la IBCN [BOY91]. Sin embargo se podría considerar como una función de control dentro del nivel ATM. El proceso de reservas es tipicamente un proceso de canal asociado y debería ejecutarse por hardware muy rápido, con el fin de limitar la ineficiencia de la red ante las diversas ráfagas.

El usuario FRP/DT debe intercambiar mensajes mono-celdas con la unidad FRP que deberá ser ejecutada por un microprocesador.

Consideramos una unidad de control FRP asociada a una línea multiplex a 155.52 Mbps (SDH) en el interface Tb.

Debe asociarse un contexto FRP/DT con cada conexión; este contiene ßmax, que es el valor de la máxima velocidad que requiere la conexión que supone 24 bits, ß que es la velocidad utilizada en un momento determinado y supone también 24 bits, D el time-out requerido que son 8 bits, un campo de control de 2 bits y un contador para las sucesivas retransmisiones de 6 bits.

En resumen se requieren 88 bits para el contexto de conexión del FRP/DT en la unidad FRP.

Todos los mensajes enviados a la red están asociados a temporizadores. El contexto de los mismos contienen la identificación de la conexión con 16 bits y un ßreq como la velocidad requerida con 24 bits. El contexto de temporizadores RAM no debería introducir bloqueo en el proceso del FRP/DT. Un estudio basado en el procedimiento Erlang-B nos indica que en el

peor caso [BOY91] la previsión debería ser para un contexto de 64 temporizadores (por ejemplo 320 bytes). En la implementación propuesta por [BOY91] se requiere una RAM de 64 bits y 128 puertas lógicas.

Se precisará un microprocesador para el intercambio de los mensajes mono-celdas del protocolo.

Los procedimientos correspondientes en el protocolo a "celda de solicitud de reserva" y "desconexión de solicitud de reserva" suponen 1300 puertas lógicas por port de salida.

Suponiendo un número maximo de 2E16 conexiones FRP/DT en un enlace de 622.08 Mbps (SDH), se requiere una RAM de 1.6 Mbps por port de salida. Esto es posible implementarlo con la tecnología actual, y con costes en franco decrecimiento [BOY91].

En referencia a los temporizadores se podrá necesitar circuitos integrados "customized" compartidos por todas las puertas de salida requiriendo una RAM de 2048 bits y 1024 puertas lógicas.

La consecuencia que obtenemos de estas indicaciones de [BOY91] es que tecnologicamente es posible realizar fisicamente el protocolo de una forma razonable, aunque como ya se ha indicado en esta tesis, éste se encuentra en una fase muy primaria de investigación.

I4. Smart Traffic Warden (STW1)

La aplicación de STW implicará la utilización de un chip adicional a las funciones de policía del tipo Leaky Bucket.

Los parámetros de diseño serán los siguientes:

Por cada circuito virtual del nodo se necesitará almacenar su velocidad de pico, aunque este parámetro ya es conocido por las funciones de policía por lo que no debemos considerarlo en este diseño.

Al contador T para una velocidad de pico mínima de 16 Kbps y a una velocidad del enlace de 622.08 Mbps le corresponden un valor máximo R de 800 slots por celda que corresponde a 10 bits. Para T se necesitarán también 10 bits.

Será necesario mantener la última posición ocupada del buffer L, y para ello serán necesarios como máximo 6 bits.

Para la operación S=(L-T)/R se necesitarán 6 bits para el resultado.

Se necesitará tener estos bits para cada uno de los circuitos virtuales y además mantener el sumatorio de S de todos los circuitos virtuales 6 bits y el tamaño del buffer 6 bits.

Las operaciones necesarias serán por circuito virtual:

. Decrementar T

. Hacer T = R cuando llega una celda

. Calcular R = Vt/Vp

. Calcular (L-T)/R truncando la operación = S

y para todos los circuitos virtuales en general comparar el sumatorio de S con W-L.

Necesitaremos en total por circuito virtual:

. 38 bits

. Una decrementación

. Una comparación

. Dos divisiones

. Dos restas

Necesitaremos en general

. 6 bits

. Una comparación

Todo ello nos lleva a pensar que la realización de un chip que solvente el algoritmo STW no debe ser mas compleja que la de un chip de policía Leaky Bucket como el indicado en el apartado I1, y su diseño queda como línea abierta de esta tesis.

Parte J. Conclusiones generales de la tesis

J1. Conclusiones generales.

En esta tesis se ha hecho un estudio del comportamiento frente a la congestión de las redes ATM, y se han visto los métodos propuestos para resolver este problema.

Los métodos tradicionales de funciones de policía,estudiados con detalle en esta tesis, se han mostrado ineficaces para controlar la congestión y simultáneamente garantizar la calidad de servicio contratada.

Para solucionar esta ineficacia, se han estudiado los métodos propuestos actualmente, posteriores a las funciones tradicionales, basados o bien en la reserva previa de recursos como los protocolos FRP/DT y FRP/IT, o bien en la complementación de funciones tradicionales de policía como el espaciamiento de celdas, aunque ambos métodos están todavía en fase de investigación sin resultados publicados por el momento.

Como un algoritmo ambivalente, como complemento de una función tradicional de policía (STW1), o como función específica de policía (STW2), se ha propuesto y analizado en esta tesis la vigilancia inteligente del tráfico, denominada "Smart Traffic Warden (STW)", englobada en la clasificación propuesta en esta tesis como Función de policía a nivel de celda y dependiente de la carga de nodo, que actúa sobre las celdas eliminadas por las funciones de policía cuando ello es posible.

Los resultados presentados indican los siguiente:

1. STW1 se presenta como una función complementaria a la función de policía Leacky Bucket, que mejora de forma sustancial y en función de la carga general del nodo, la tasa de error obtenida y la utilización general de recursos, sin introducir congestión y sin alteraciones sustanciales en los parámetros que definen la calidad de servicio.

2. STW2 se presenta como una función de policía de nodo en sí, que, igual que STW1, mejora los resultados globales de ocupación de recursos, sin menoscabo de la calidad de servicio, aunque sus resultados no son tan satisfactorios como el STW1, ya que con cargas elevadas y tráficos parciales infractores muy intensos, al aplicar su función de forma "democrática", la calidad de servicio de las fuentes no infractoras se ve perjudicada. Por ello se considera que no es aplicable de forma general.

3. STW rompe el compromiso de mantenimiento de la calidad de servicio y control de la congestión, ya que permite ambas funciones, tanto mejor como la red esté alejada de su máxima carga.

4. STW2 mejora el resultado global de la red frente a la congestión y ocupación de recursos ante los usuarios "tricky".

En este caso STW1 no aporta nada, aunque se indica la no necesidad de detección de dichos usuarios, y en todo caso ésta es casi siempre posible en general.

5. Se podría dimensionar el Leacky Bucket con un bucket tal que permitiera los mismos resultados en calidad de servicio que el STW, con o sin jitter, pero esto supondría aumentar la congestión ,tal como se indica en el apartado G, aspecto que evita STW.

6. El problema de la congestión en general en redes ATM es de muy difícil solución desde un punto de vista exacto, y por lo tanto se cree que es posible que la solución se decante definitivamente hacia la sobredimensión de recursos, cada vez más baratos, y es en este entorno donde STW puede proporcionar grandes ventajas de utilización para la rentabilización de esos recursos por parte de los operadores de red.

J2. Lineas de investigación abiertas

1. STW se ha basado en este estudio en la velocidad de pico contratada. Se podría estudiar el comportamiento de STW basado en otros parámetros estocásticos del proceso de rafagueo, como son la velocidad media o el tiempo medio de ráfaga.

2. STW introduce en su aplicación una probabilidad de error, que depende del tipo de tráfico, del tipo de multiplexores estadísticos utilizados en la red y del número de nodos en los que se aplique. Un estudio analítico de este proceso ayudaría a su comprensión y a determinar sus límites de aplicación.

3. Sería útil realizar comparaciones de STW con los métodos emergentes de reserva de recursos FRP/DT y FRP/IT,y con los de espaciamiento de celdas "cell spacing".

4. Un análisis del comportamiento de STW ante un tráfico no homogéneo y formado por diversas fuentes con tráficos diversos (fuentes complejas) con una intensidad de tráfico global determinada, nos acercaría todavía más a situaciones reales.

5. Un análisis del comportamiento de las funciones de policía respecto al STW en relación a la secuenciación de pérdidas de celdas. Este aspecto determinante de la calidad de servicio, no es lo mismo perder cien celdas seguidas que repartidas, no está suficientemente estudiado, y es muy importante en determinados servicios.

6. Estudio y diseño de un chip que permita la realización del algoritmo STW en línea con las velocidades que se manejan.

BIBLIOGRAFIA

[AKY92] Ian F. Akyildiz and Yu Gong
"Average rate enforcement for real-time traffic in ATM networks".
Seminario banda ancha. UPC. 1992.

[AMS89] Stanford R. Amstutz.
"Burst switching-an update"
IEEE Communication Magazine, September 1989.

[ARM86] Heinrich Armbrüster
"Applications of the future Broad-Band services in the office and home"
IEEE Journal on Selected Areas in Communications,July 1986.

[ARM87] Armbrüster, G. Arndt.
"Broadband communication and its realization with broadband ISDN"
IEEE Communication Magazine ,Noviembre de 1987.

[ARM89] Heinrich Armbrüster
"World-wide Approaches to Broadband ISDN"
Telecommunications, pp. 49-54, May 1989.

[BAT88] D.V. Batorsky et al.
"The Evolution of Broadband Network Architectures"
IEEE Globecom'88, Vol. I, pp. 367-373.

[BOR91] Flaminio Borgonovo and Luigi Fratta
"Policing procedures: Implications, definitions and proposals"
ITC-91. 13 th International Teletraffic Congress

[BOY91] Pierre E. Boyer and Didier P. Tranchier
"A reservation principle with applications to the ATM traffic control".
IEEE Globecom, 2-5 December 1991.

[BOY92] P. Boyer and F. Guillemin
"ATM based network congestion"
Traffic and Performance aspects in IBCN.January 1992. Aveiro (Portugal)

[BUR91] John Burgin and Dennis Dorman.
"Broadband ISDN Resource Management:The role of virtual Paths".
IEEE Communications Magazine.September 1991.

[BUT91] Milena Butto et alt.
"Effectiveness of the "Leaky Bucket" Policing Mechanism in ATM Networks".
IEEE Journal on Selected Areas in Communications April 1991.

[BYR89] William R. Byme et alt.
"Broadband ISDN Technology and Architecture".
IEEE Network. January 1989.

[CAS89] L.N. Cassell et alt.
"Network management architectures and protocols"

IEEE JSAC. Septiembre 1989.
[CAS91] P. Castelli et alt.
"Policing and call admission problems in ATM networks"
ITC-91. 13 th International Teletraffic Congress.
[CCI84] CCITT COM XVIII-228-E March 1984.
[CCI88] CCITT Rec. I.121, Geneva, June 1988.
[CCT88] CCITT Recommendation M30, Principles for a TMN,marzo 1988
[CCI90] CCITT Recommendation I.311. Enero 1990.
[CHE88] Thomas M. Chen and David G. Messerschmitt
"Integrated voice/data switching"
IEEE Communications magazine, june 1988
[CID91] Israel Cidon et alt.
"Bandwidth Management and Congestion Control in plaNET"
IEEE Communications Magazine October 1991
[COR91] Jordi Cortadella
Curso de doctorado Banda Ancha. UPC 1991.
[DAD89] J. E. Daddis y H. C. Torng.
"A taxonomy of broadband integrated switching architectures"
IEEE Communication Magazine, vol.27, nº 5 (Mayo de 1989).
[DEC90] M. Decina et alt.
"Bandwidth assignment and virtual call blocking in ATM networks".
Globecom IEEE 1990.
[DIR92] M.J.G. Dirksen
"Deriving Leaky Bucket parameters for jettered CBR streams".
RACE Workshop.Portugal january 1992.
[DIT91] Lars Dittmanin et alt.
"Flow Enforcement Algorithms for ATM Networks".,
IEEE Journal on Selected Areas in Commnunications April 1991.
[DZI90] Zbigniew Dziong et alt.
"Admission Control and Routing in ATM Networks"
North Holland.Computer Networks and ISDN Systems 20(1990)
[ECK89] Adrian E. Eckberg et alt.
"Bandwidth Management:A Congestion Control strategy for Broadband Packet Networks-Characterizing the throughput-burstiness filter.
ITC specialist seminar.25-29 sept.1989,Adelaide(Aus).
[ECK91] Adrian E. Eckberg et alt.
"Controlling congestion in B-ISDN/ATM:Issues and strategies".
IEEE Communications Magazine.September 1991.
[EXP88] Expert Group on ATM Switch Structures.
"Final report of cost-202bis". December 1988.
[FEN91] K. W. Fendick et alt.
"An aproach to high performance,High speed data Networks".
IEEE Communications Magazine October 1991

[FIC91] G. Fiche and C. Le Paud
"Acceptance criteria of bursty traffics in a frame concentrator"
ITC 91.Elsevier Science Publishers B.V.
[FRA91] Alexander G. Fraser
"Designing a Public Data Network"
IEEE Communications Magazine October 1991
[GAL89] G. Gallassi et alt.
"ATM:Bandwidth assignment and Bandwidth anforcement policies".
Globecom IEEE-1990
[GAL90] G. Gallassi et alt.
"Resource Management and dimensioning in ATM Networks"
IEEE Network Magazine. May 1990.
[GAR92] Jorge García.
"Modelos analíticos para la evaluación de mecanismos de control de tráfico en redes ATM".
Tesis doctoral. UPC 1992.
[GEC89] Jerry Gechter and Peter O`Reilly.
"Conceptual issues for ATM".
IEEE Network.January 1989.
[GER89] P.Gerke, J.F. Huber.
"Fast packet switching.A principle for future systems generations".
IEEE Communications Magazine. Febrero 1989.
[GIB91] Henry Gilbert et alt.
"Developing a cohesive traffic Management Strategy for ATM Networks"
IEEE Communications Magazine October 1991
[GIL91] Emilio Gil Delicado
"Evolucióbn de la transmisión:Sistemas síncronos".
Mundo electrónico.Octubre 1991.
[GON91] A. Gonzalez et alt.
"Redes de banda ancha.Planes experimentales en España"
Mundo electrónico.Octubre 1991.
[GUI90] RACE project 1003
"An implementation architecture for the Telecommunications management network"
RACE programme. Marzo 1990.
[GUI92] Fabrice Guillemin et alt.
"The spacer-controller: Architecture and first assessments".
Broadband Communications. Estoril, Portugal. Enero 1992.
[HAA91] Zygmunt Haas and Jack H. Winters
"Congestion Control by adaptative admission"
Globecom IEEE 1991.
[HAC89] Anna Hac and Hasan B. Mutlu
"Synchronous optical network and broadband ISDN protocols".
IEEE Computer. November 1989.
[HAB91] Ibrahim W. Habib and Tarek N. Saadawi.
"Controlling Flow and Avoiding Congestion in Boradband

Networks."
IEEE Communications Magazine October 1991

[HAH91] E.L. Hahne et alt.
"Fairness and congestion control on a large ATM data network with dynamically adjustable windows"
ITC-91. 13 th International Teletraffic Congress

[HAN89] Rainer Handel.
"Evolution of ISDN towards broadband ISDN"
IEEE Network.January 1989.

[HAR91] K.R. Harrison
"The new CCITT Synchronous Digital Hierarchy: Introduction and overview".
B.T. Engineering, vol 10, july 1991.

[HEI92] Herbert Heiss
"Impact of Jitter on peak cell rate policing with a Leaky Bucket".
RACE Workshop.Portugal january 1992.

[HEM91] Hilde Hemmer and Per Thomas Huth
"Evaluation of policing functions in ATM Networks"
ITC 91. Elsevier Science Publishers B.V.

[HON91] Duke Hong and Tatsuya Suda
"Congestion Control and Prevention in ATM Networks"
IEEE Network Magazine, july 1991.

[HON91] Duke Hong et alt.
"Survey of techniques for prevention and control of congestion in an ATM network".
ITC-91

[HUI88] Joseph Y. Hui
"Resource Allocation for Broadband Networks"
IEEE Journal on selected areas in communications. Dec. 1988

[HUI89] Joseph Y. Hui.
"Network transport, and switching integration for broadband communications"
IEEE Network, Mars 1989.

[IVE92] Villy B. Iversen and Annemarie Bohn Nielsen
"Statistical multiplexing in ATM networks.
RACE Workshop. Portugal, january 1992.

[JAI90] Raj Jain.
"Congestion Control in Computer Networks"
IEEE Network Magazine.May 1990.

[JAL91] A. Jalali and L.G. Mason
"Open loop schemes for network congestion control".
ITC-91

[JON91] H. Jonathan Chao
"Design of leaky bucket access control schemes in ATM networks".
ITC-91

[JOO89] P. Joos and W. Verbiest
"A statistican bandwidth allocation and usage monitoring algorithm for ATM networks".
Globecom IEEE 1989.

[KEN91] Ken-Ichi Sato et alt.

"The roll of virtual path crossconnection"
IEEELTS, august 1991

[KOH88] C. Kohli, D. S. Biring y G. L. Raya
"Emerging broadband packet-swith twechnology in integrated information networks"
IEEE Network, November 1988.

[KRO90] Hans Krönen et alt.
"Queuing models for ATM systems-A comparison".
7 th ITC Seminar, Morristown. October 9-11 1990.

[LAZ91] Aurel A. Lazar and Giovanni Pacifici
"Control of Resources in Broadband Networks with Quality of Service Guarantees".
IEEE Communications Magazine October 1991

[LIN91] Julio Linares.
"Sistemas de Gestión de Red"
Mundo electrónico.Octubre 1991.

[MIN89] Steven E. Minzer and Dan R. Spears.
"New directions in signaling for Broadband ISDN"
IEEE Communications Magazine.February 1989.

[MIZ89] Steven E. Minzer
"Broadband ISDN and asynchronous transfer mode (ATM)"
IEEE Communication Magazine, september 1989.

[MOR91] Andreas Morgenroth
"Improving Transmission in Public Networks"
Communications International.Abril 1991.

[MUR91] Tutomu Murase et alt.
"A call admission control for ATM networks based on individual multiplexed traffic characteristics".
ITC-91

[NAK89] Kyoichi Nakamaki et alt.
"Traffic Control for ATM Networks"
Globecom IEEE 1989.

[NIE90] G. Niestegge
"The leaky bucket policing method in the ATM network"
Int.J.of Digital&Analog Comm.systems, Vol 3.1990.

[NIK92] Ioanis Nikolaidis and Ian F. Akyildiz
"Source characterization and statistical multiplexing in ATM networks".
Tec.Report. Georgia Inst.of Tec. Atlanta Julio 1992.

[NOV91] Bernard P. Novak
"Desarroolo del marco legal e institucional de las telecomunicaciones en EEUU,Japón y Europa.
Mundo Electrónico,Octubre 1991.

[NUS88] E. Nussbaum.
"Communications networks needs and technologies-a place for photonic switching"
IEEE Journal on Selected Areas on Communications, august 1988.

[OHN86] Horst Ohnsorge
"Introduction and Overview of Broand-Band Communication Systems"
IEEE Journal on Selected Areas in Communications,July 1986.

[OKA91] Tadanobu Okada et alt.
"Traffic Control in Asynchronous transfer mode"
IEEE Communications Magazine.September 1991.
[PER87] Stewart D. Personik
"Communication switching-from operators to photonics"
Proceedings of the IEEE, October 1987.
[RACa90]RACE Project 1022 "Technology for ATD"
"The relationship of the telecommunications Management Network to connection acceptance control and source policing"
November 1990.
[RACb90]RACE Project 1022 "Technology for ATD"
"Updated results of traffic simulation of the policing experiment"
December 1990.
[RAS91] Carsten Rasmussen et alt.
"Source-Indpendent Call Acceptance Procedures in ATM Networks"
IEEE Journal on Selected Areas in Commnunications April 1991.
[RAT91] Erwin P. Rathgeb
"Modeling and Performance Comparison of Policing Mechanisms for ATM Networks".
IEEE Journal on Selected Areas in Communications April 1991.
[RID89] Michael J. Rider.
"Protocols for ATM access networks".
IEEE Network. January 1989.
[ROB88] J.Roberts and A. Simoniam
"Some queueing models for an ATM multiservice network"
COST 224.Septiembre 1988.
[ROB91] James W. Roberts.
"Variable bit-rate traffic control in B-ISDN".
IEEE Communications Magazine.September 1991.
[SANa90]Germán Santos y Josep Solé.
"Comunicaciones en banda ancha:Gestión de red".
Mundo Electrónico.Septiembre 1991.
[SANb90]Germán Santos y Jordi Domingo.
"Leaky Bucket with dynamic bandwitdh allocation"
Technical Report. UPC 1990.
[SANa91]Germán Santos Boada.
"Present i futur de les telecomunicacions a Barcelona".
Barcelona Economia. 1er. Trimestre 1991.
[SANb91]Germán Santos Boada.
"La Red Digital de Servicios Integrados"
MTV-TV4. Junio y Julio-Agosto 1991.
[SANc91]Germán Santos, Jordi Domingo y J.L. Pascual.
"Congestion Control in ATM Networks with Smart Traffic warden".
IEEE Singapore Conference on Networks.3-6 September 1991.
[SANd91]Germán Santos, Jordi Domingo y Josep Solé
"Redes de área metropolitana. Transmisión de datos a

alta velocidad"
Tutorial. Convención Informática Latina 1991. (CIL-91)
[SAT91] Ken-Ichi Sato et alt.
"The roll of virtual path Crossconection"
IEEELTS , Agosto 1991.
[SEG91] Hubert Segot et alt.
"Gestion de reseau:état de la normalisation"
Annu.Telecommunications,nº 7-8 1991.(CNET)
[SOL90] Josep Solé y Germán Santos.
"Comunicaciones en banda ancha:Sistemas de conmutación".
Mundo electrónico.Abril 1990.
[SOL91] Josep Solé Pareta
"Estudi i proposta d'esquemes d'avaluació per a dispositius ATM"
Tèsi Doctoral. UPC 1991.
[SPE87] Dan R. Spears
"Broadband ISDN switching capabilities from a services perspective".
IEEE Journal on selected areas in communications. Oct. 1987
[STA85] W. Stallings.
"Integrated services digital network (ISDN)" Tutorial.
IEEE computer Society, 1985.
[SUR90] José A. Suruguay Monteiro et alt.
"Leacy Bucket input rate control in ATM Networks"
ICC90,New Delhi,India 1990.
[TAK89] Kenzo Takahashi et alt.
"Communications Quality Analysis for ATM Networks"
Globecom IEEE 1989.
[TAN88] A. S. Tanenbaum.
"Computer networks"
2nd edition, Prentice Hall 1988.
[TEL90] Telefónica.
Manuales de Tecnología.Especificaciones ISDN.
5 tomos. 1990.
[TIM89] Stephen Timms
"Broadband communications. The commercial impact" IEEE Network, July 1989.
[TSE91] Kent H. Tseng and Man-Tung T. Hsiao
"Admission control of voice/data integration in an ATM network".
ITC-91
[TUR86] J. Turner.
"New directions in communications".
IEEE Communication Magazine,vol.24,nº 10 (Octubre de 1986).
[VAL89] T. Valovic.
"Metropolitan area nerworks: a status report"
Telecommunications, pp. 25-32, July 1989.
[VAK91] Faramak Vakil and Hiroshi Saito
"On Congestion Control in ATM Networks"
IEEELTS, Agosto 1991.
[VOR88] J. P. Vorstermans, A. P. de Vleeschouwer.
"Layered ATM systems and architectural concepts for

subscribers premises network"
IEEE Journal on Selected Areas in Communications, Vol.6, nº 9, pp. 1545-1555, December 1988.

[WAL89] D. Wallace.
"FDDI makes the fibre connection" Telecommunications, pp. 29-30, October 1989.

[WAL91] Steve Walters.
"A new direction for broadband ISDN"
IEEE Communications magazine.September 1991.

[WAM91] E. Wallmeier and T. Worster
"A cell spacing and policing device for multiple virtual connections on one ATM pipe".
Commission of the European Communities. RACE 1012.

[WOL91] S. Wolf et alt.
"How will we rate telecommunications system performance?"
IEEE Communications Magazine October 1991

[WRI91] T.C. Wright.
"SDH multiplexing concepts and methods".
Britsh Telecom. Engineering.Vol 10, july 1991.

[YOK89] Tadahiro Yokoi and Kunio Kodaira.
"Grade of service in the ISDN era".
IEEE communications Magazine.April 1989.