

Proporcionar calidad de servicio en Advanced Switching*

Raúl Martínez, Francisco J. Alfaro, José L. Sánchez
Dept. de Informática. Escuela Politécnica Superior
Universidad de Castilla-La Mancha
02071- Albacete, España
{raulmm, falfaro, jsanchez}@info-ab.uclm.es

Resumen

Advanced Switching Interconnect (ASI) es una nueva tecnología basada en PCI Express diseñada para proporcionar en un estándar abierto la funcionalidad de los sistemas de interconexión propietarios que han estado en el núcleo de sistemas empujados, de almacenamiento y comunicaciones.

Proporcionar calidad de servicio (QoS) en entornos de computación y comunicaciones es actualmente el centro de muchos esfuerzos en investigación por parte de la industria y el ámbito académico. ASI incorpora mecanismos que apropiadamente usados se espera que proporcionen QoS. En este trabajo examinamos estos mecanismos y mostramos cómo proporcionar garantías de QoS basadas en requisitos de ancho de banda y latencia. Además explicamos cómo proporcionar simplemente un tratamiento diferenciado, pero sin proporcionar garantías. Así mismo, se propone un nuevo algoritmo, el Weighted Fair Queuing Credit Aware, como una implementación concreta de uno de los planificadores de salida que sugiere ASI.

1. Introducción

El bus PCI ha servido bien a la industria durante los últimos 10 años y todavía se usa hoy en día ampliamente en gran variedad de siste-

mas tanto personales como corporativos. Sin embargo, los procesadores y dispositivos de entrada/salida actuales y futuros demandan mucho más ancho de banda del que PCI puede proporcionar. La razón de esta limitación es precisamente la arquitectura de bus de PCI. PCI Express es una nueva generación de PCI que sustituye el tradicional bus por una red conmutada, manteniendo los modelos de uso y las interfaces software.

Advanced Switching Interconnect o simplemente Advanced Switching (AS) es una nueva tecnología de interconexión de altas prestaciones que cuenta como gran ventaja el hecho de ser la expansión natural de PCI Express, al estar construida sobre las capas físicas y de enlace de dicha tecnología. En los últimos días de 2003, se publicó la versión 1.0 de la especificación de AS [1].

Por otra parte, proporcionar QoS en entornos de computación y comunicaciones es actualmente el centro de muchos esfuerzos de investigación por parte de la industria y en el ámbito académico. Por ejemplo, la Internet Engineering Task Force (IETF) tiene dos propuestas para proporcionar QoS en Internet. La primera, denominada Servicios Diferenciados [3], distingue entre diferentes clases de tráfico y proporciona un tratamiento diferenciado a estas categorías. La segunda se denomina Servicios Integrados [4], y proporciona garantías de QoS a los distintos flujos basados en los requisitos solicitados durante la fase de establecimiento de la conexión.

AS incorpora mecanismos que apropiadamente usados se espera que proporcionen QoS.

*Este trabajo ha sido parcialmente financiado por la CICYT con el proyecto TIC2003-08154-C06 y por el Ministerio de Educación y Ciencia con una beca FPU.

En concreto, AS permite utilizar canales virtuales, clases de tráfico, arbitraje en los puertos de salida y ofrece la posibilidad de usar un mecanismo de control de admisión. En este trabajo examinamos estos mecanismos y mostramos cómo proporcionar a las aplicaciones requisitos basados en ancho de banda y latencia. Así mismo, se propone un nuevo algoritmo, el Weighted Fair Queuing Credit Aware, como una implementación concreta de uno de los árbitros de salida que sugiere AS.

La estructura de este artículo es la siguiente: La Sección 2 presenta un resumen de los aspectos generales de la especificación de AS, incluyendo los mecanismos más importantes que incorpora AS para proporcionar QoS. En la Sección 3 presentamos nuestra propuesta de cómo emplear los citados mecanismos. Los detalles de la plataforma experimental y la evaluación de las prestaciones se muestran en la Sección 4. Por último, se muestran algunas conclusiones y se propone trabajo futuro.

2. Advanced Switching

La arquitectura AS está construida sobre las capas de enlace y físicas de PCI Express. La capa física consiste en una pareja de canales unidireccionales implementados como una pareja emisor/receptor. La codificación empleada sigue el esquema 8b/10b. La frecuencia de los enlaces es de 2,5 Gb/s. El ancho de banda de los enlaces puede ser linealmente escalado añadiendo en paralelo varios de estos canales serie. La capa de enlace es responsable de la integridad de los datos y añade un número de secuencia y un CRC a la capa de transacción, retransmitiendo automáticamente los paquetes corruptos que se reciban. La capa de enlace se encarga además de implementar un control de flujo basado en créditos, que asegura que los paquetes sólo sean transmitidos cuando el buffer en el otro extremo sea capaz de recibirlos.

AS soporta tráfico unicast y multicast. Para el primero, la capa de transacción de AS proporciona encaminamiento fuente. AS encapsula los paquetes de datos y les agrega una cabecera que permite su encaminamiento a través

de la red, sin importar el formato original del paquete. De este modo, en teoría, cualquier protocolo de las capas de transporte, red o enlace de datos puede ser encaminado en una red AS. Se puede construir cualquier topología como estrellas, árboles, mallas o topologías irregulares. La técnica de conmutación que emplea AS es virtual cut-through.

2.1. Soporte de AS para QoS

AS proporciona mecanismos que correctamente usados permiten soportar QoS. Específicamente, AS incorpora canales virtuales, un identificador de tipo de tráfico y arbitraje a la salida de los elementos de red, así mismo, contempla el uso de un control de admisión de conexiones (CAC).

Los canales virtuales (CVs) proporcionan un mecanismo para dar soporte a múltiples caminos de datos independientes sobre un mismo canal. AS soporta hasta 20 CVs clasificados en tres tipos diferentes: hasta 8 CVs unicast con cierta capacidad de adelantamiento (BVCs), hasta 8 CVs unicast ordenados (OVCs) y hasta 4 CVs multicast (MVCs).

La cabecera del paquete AS contiene un campo de tres bits para un identificador de tipo de tráfico, que permite especificar ocho posibles tipos de tráfico. El identificador de tipo de tráfico se transmite, sin modificarse, desde la fuente hasta el destino a través de la red AS. En cada salto en la red AS, el identificador de tipo de tráfico se utiliza para seleccionar el CV específico que se le asigna. Por cada tipo de CV se establece una correspondencia entre identificador de tipo de tráfico y CV, que depende del número de CVs implementados en cada elemento de la red.

AS define dos árbitros de salida para resolver los conflictos entre los hasta 20 CVs compitiendo por el ancho de banda del enlace de salida. Estos árbitros son el planificador basado en tabla de arbitraje de CVs y el planificador de enlace de salida de ancho de banda mínimo o simplemente MinBW. Una implementación concreta puede elegir cualquiera de los dos o implementar su propio mecanismo propietario.

El planificador basado en tabla emplea una tabla de arbitraje para seleccionar el paquete

te que debe ser transmitido en cada instante. Esta tabla consiste en un vector de registros con entradas que contienen el identificador de CV que indica qué CV debería ser el siguiente en transmitir un paquete. La tabla se recorre secuencialmente y si el CV indicado por la entrada actual no tiene paquetes para transmitir, esa entrada se salta. El número de entradas de la tabla puede ser 32, 64, 128, 256, 512 ó 1024.

El planificador MinBW está diseñado para una asignación más precisa del ancho de banda teniendo en cuenta el tamaño de paquete. La estructura de este planificador consta de dos partes. La primera es un mecanismo para proporcionar a un único CV, el Canal de Gestión de la Red, prioridad por encima de cualquier otro CV, pero con un ancho de banda limitado por un *token bucket*. Este canal de gestión corresponde al BVC de mayor índice implementado en cada elemento de red. La segunda parte es un mecanismo para distribuir el ancho de banda entre el resto de CVs de acuerdo a unos pesos que indican el ancho de banda relativo que le corresponde a cada CV. AS no proporciona un algoritmo específico o implementación para el árbitro MinBW, pero ésta debe respetar ciertas propiedades [1].

Además, el software de gestión de la red puede llevar a cabo un control de admisión, permitiendo a los nuevos flujos de datos entrar en la red sólo si hay suficientes recursos disponibles. La especificación de AS solamente cita la posibilidad de usar este mecanismo, pero no da indicaciones de cómo implementarlo.

3. Proporcionar QoS sobre AS

En esta sección mostramos nuestra propuesta de cómo utilizar los mecanismos de AS citados anteriormente para proporcionar QoS. Lo primero es especificar un conjunto de clases de servicio (Service Classes, SCs) con distintos requisitos. Conforme los distintos flujos acceden a la red serán agregados en las distintas SCs dependiendo de sus características. En AS podemos distinguir hasta 16 SCs, que es el número máximo de CVs unicast. Emplearemos una SC para el tráfico de control de la red, algunos para tráfico con requisitos de QoS y el

resto para best-effort. En este trabajo consideramos dos tipos de requisitos de QoS, mínimo ancho de banda y/o latencia máxima. Por tanto, mostraremos cómo configurar los dos planificadores propuestos por AS, el MinBW y el basado en tabla, para proporcionar estos requisitos. El tráfico de control de red lo asignaremos al Canal de Gestión de la Red para que obtenga la máxima prioridad en el caso de usar el MinBW. En el caso del planificador basado en tabla, este CV se maneja de la misma forma que cualquier otro CV, y por lo tanto lo trataremos como un CV con unos requisitos de latencia elevados. Las SCs de tráfico best-effort sólo se caracterizan por la prioridad relativa entre las mismos, por lo tanto, les asignaremos una cantidad de ancho de banda reducida proporcional a su importancia.

3.1. Proporcionar requisitos de QoS con el planificador basado en tabla

En [2] explicamos cómo configurar este tipo de tabla de arbitraje (en el entorno de InfiniBand) para proporcionar garantías de ancho de banda y latencia. Para proporcionar un determinado ancho de banda mínimo a un CV hay que asignar un número total de entradas en la tabla proporcional al ancho de banda que se quiere asignar. Para proporcionar requisitos de latencia máxima a un CV se tendrá en cuenta la distancia máxima que hay entre dos entradas consecutivas de dicho CV en la tabla de arbitraje. Esta distancia acota el número de paquetes que pueden salir entre dos paquetes pertenecientes a dicho CV, y por tanto el tiempo de espera en los buffers de salida.

Esta forma de asignar entradas de la tabla tiene el problema de ligar la asignación de entradas realizadas para proporcionar requisitos de ancho de banda y latencia. Nótese que si se asigna una determinada distancia máxima entre entradas de la tabla de un mismo CV, se está indirectamente asignando una cierta cantidad de entradas, y por tanto, un mínimo ancho de banda al CV en cuestión. Otro gran problema de la tabla es que no funciona de manera adecuada con paquetes de tamaño variable. Este problema se obvia en este estudio usando paquetes de tamaño fijo. Sin embargo,

estamos estudiando diferentes formas de usar la tabla para solucionar esta limitación.

3.2. Proporcionar requisitos de QoS con el planificador MinBW

Proporcionar requisitos de mínimo ancho de banda a un CV con el planificador MinBW, consiste simplemente en asignar a dicho CV un peso igual a la proporción de ancho de banda del enlace de salida deseada. La especificación de AS indica que algunas implementaciones de la técnica de planificación Weighted Fair Queuing (WFQ) [5] exhiben las propiedades deseadas para el planificador MinBW. Parekh and Gallager [7] analizaron las prestaciones de WFQ desde el punto de vista del peor caso. Basándonos en este estudio asignaremos una cantidad de ancho de banda mayor del estrictamente necesario a aquellos CVs con unos requisitos elevados de latencia para obtener unas latencias media y máxima adecuadas.

El algoritmo WFQ original es una técnica *work-conserving* (el enlace de salida nunca permanece inactivo si hay paquetes que transmitir) que distribuye el ancho de banda entre varios flujos de acuerdo a un conjunto de pesos. WFQ mantiene control del conjunto de flujos, en nuestro caso CVs, que están activos en cada instante. Un CV se considera que está activo si tiene paquetes para transmitir. El conjunto de CVs activos se usa para actualizar el reloj virtual interno que utiliza este algoritmo. Este reloj virtual se usa a su vez para calcular el *virtual finishing time*, es decir el momento en que un paquete debería haber sido completamente transmitido en un sistema teórico que funcionase a nivel de bit. Cuando un paquete llega al planificador se marca con este *virtual finishing time*. A la hora de seleccionar el paquete que debe salir se selecciona aquel con un *virtual finishing time* menor.

Este algoritmo tiene en nuestra opinión dos problemas a la hora de ser usado en el entorno de AS. El primero es que no tiene en cuenta el tiempo usado en transmitir paquetes del Canal de Gestión de la Red, que no están controlados por este mecanismo. Esto produce que cuando la carga de tráfico de control es elevada, el reloj virtual no esté correctamente calculado y

los resultados sean inciertos. El segundo problema es que este algoritmo fue diseñado sin tener en cuenta la posibilidad de que se emplee un control de flujo, pudiéndose dar el caso de que CVs que tienen paquetes para transmitir, pero que no disponen de créditos suficientes para hacerlo, se consideren activos, asignándose ancho de banda a los mismos. Estos CVs tomarán ventaja de ese ancho de banda asignado cuando vuelvan a tener créditos suficientes, con el consiguiente efecto desestabilizador.

Para solucionar estos problemas proponemos una nueva versión del algoritmo WFQ que hemos llamado WFQ Credit Aware (WFQ-CA). El WFQCA funciona de la misma manera que el WFQ excepto en estos aspectos:

- El valor del reloj virtual no cambia durante el tiempo que se está transmitiendo paquetes de control.
- Un CV se considera que está activo sólo cuando tiene paquetes almacenados y suficientes créditos para transmitir el paquete en cabeza de la cola del CV.
- El conjunto de CVs activos puede cambiar cuando un paquete llega al CV, cuando un paquete deja el CV o cuando se reciben créditos de control de flujo.
- Cuando un CV teniendo paquetes que transmitir está inactivo a causa de la falta de créditos, cuando pasa a estado activo al haber recibido suficientes créditos, los paquetes almacenados en dicho CV son remarcados con su *virtual finishing time* como si acabaran de llegar en ese instante.

Con estas modificaciones la transmisión de paquetes de control no afecta al funcionamiento del planificador. Además, el planificador tiene en cuenta el control de flujo existente, no asignando ancho de banda a aquellos CVs que no pueden transmitir paquetes debido a la falta de créditos. El algoritmo resultante es una versión muy parecida al WFQ original que se ajusta al entorno de AS.

Tabla 1: Tipos de tráfico sugeridos por el estándar IEEE 802.1D-2004.

Tipo	Nombre	Descripción
Best-effort	Background (BK)	Transferencias masivas y otras actividades que no deberían tener impacto en el uso de la red por parte de otras aplicaciones.
Best-effort	Best-effort (BE)	Tráfico de red de área local tal como se ha conocido hasta ahora.
Best-effort	Excellent-effort (EE)	El tipo de servicio best-effort que una organización de servicios de información ofrecería a sus clientes preferentes.
QoS	Carga controlada (CL)	Tráfico de aplicaciones sujetas a alguna forma de control de admisión basado en ancho de banda.
QoS	Vídeo (VI)	Tráfico con requisitos de latencia y jitter máximo de 100 ms.
QoS	Voz (VO)	Tráfico con requisitos de latencia y jitter máximo de 10 ms.
Control	Control de red (NC)	Tráfico para mantener la infraestructura de la red caracterizado por unos requisitos de máxima prioridad.

3.3. Consideraciones finales

Para proporcionar garantías de QoS se debe emplear un control de admisión de conexiones (CAC). Sin un CAC sólo es posible obtener un esquema de prioridades donde las distintas SCs obtienen un tratamiento diferenciado, pero sin que pueda darse ninguna garantía de QoS. En cualquier caso, este CAC no se implementaría ni sobre el tráfico de control de la red, ni sobre el tráfico best-effort.

Hay dos posibles formas de configurar los planificadores. La primera posibilidad es configurar el planificador por adelantado, definiendo un conjunto de SCs con unos requisitos dados de ancho de banda mínimo y latencia máxima [8]. Esta distribución se haría teniendo en cuenta el uso esperado de cada SC. La segunda posibilidad es configurar los planificadores de acuerdo a los requisitos de las conexiones de una forma dinámica. Esto permite una mayor flexibilidad y un uso más preciso de los recursos. Nótese que la segunda posibilidad sólo es posible cuando se usa un CAC.

4. Evaluación de prestaciones

En esta sección evaluamos mediante simulación el comportamiento de nuestras propuestas. Primero describiremos las principales características de la red simulada, el modelo de tráfico y la configuración de los planificadores. En segundo lugar presentamos y analizamos los resultados obtenidos.

4.1. Arquitectura simulada

La red que se ha simulado es una red multietapa de 64 nodos con un patrón de conexión *perfect-shuffle*. El modelo de conmutador empleado tiene la típica arquitectura de buffers a la entrada y a la salida y un *crossbar* como elemento de conmutación. Para evitar el *head of line blocking* la entrada se ha implementado *virtual output queuing* (VOQ) a nivel de conmutador. Esto se traduce en que para cada CV hay un número de colas igual al número de puertos del conmutador que comparten la misma cuenta de créditos.

En nuestras pruebas utilizamos enlaces de 2.5 Gb/s, pero, con la codificación 8b/10b, el ancho de banda efectivo es de 2Gb/s. Asumimos una aceleración del *crossbar* de (x1.5), cómo es usual en muchas de las implementaciones comerciales. Como AS da libertad a la hora de elegir el algoritmo de planificador del *crossbar*, nosotros hemos elegido implementar el planificador *First Come, First Served* (FCFS). Los paquetes, de tamaño fijo, son de 128 bytes.

4.2. Modelo de tráfico

Para simular un tráfico suficientemente realista, se ha inyectado tráfico perteneciente a los tipos de tráfico que define el estándar 802.1D-2004 [6] en el Anexo G. Dichos tipos de tráfico se muestran en la Tabla 1.

La Tabla 2 muestra la proporción de tráfico que inyecta cada nodo respecto al máximo permitido por los enlaces. Esta situación

Tabla 2: Tráfico inyectado y configuración de los planificadores.

CV	Tipo	Tráfico inyectado			C. Tabla		C. MinBW
		Min. %	Max. %	Patrón	nº entr.	Dist.	Peso
7	NC	1	1	autosimilar	16	4	-
6	VO	18.75	18.75	CBR 64Kb/s	16	4	0.25
5	VI	18.75	18.75	trazas MPEG-4 750 Kb/s	12	6	0.1875
4	CL	18.75	18.75	CBR 750 Kb/s	12	-	0.1875
3	EE	9.25	29.25	autosimilar	5	-	0.078125
2	BE	9.25	29.25	autosimilar	2	-	0.03125
1	-	0	0	-	0	-	0
0	BK	9.25	29.25	autosimilar	1	-	0.015625
		85	145		64		0.75

responde al uso de un CAC. Aparte de una cierta cantidad de tráfico de control de fondo, la cantidad de tráfico con QoS que se inyecta es la máxima que permite el CAC y permanece constante durante las simulaciones. Para comprobar que el tráfico con QoS se comporta adecuadamente bajo cualquier carga, la cantidad de tráfico best-effort inyectada aumenta hasta niveles muy superiores al máximo permitido por los enlaces. Se han empleado distintos patrones de generación para emular los diferentes tipos de tráfico. La voz, el vídeo y la carga controlada están compuestos por conexiones punto a punto del ancho de banda indicado. En el caso del vídeo, los *frames* de las trazas se trocean en paquetes que son inyectados de manera equidistante a lo largo del tiempo de *frame* (40 ms). La distribución de los destinos es uniforme.

4.3. Configuración de los planificadores

En nuestras pruebas comparamos las prestaciones ofrecidas por los dos planificadores que propone AS. Para el planificador basado en tabla hemos empleado una tabla de arbitraje de 64 entradas. Para implementar el MinBW hemos usado el algoritmo WFQCA propuesto en la Sección 3.2. En la Tabla 2 se muestra la configuración de los mismos. Para configurar la tabla, hemos asignado al CV que acomoda al tráfico de control una distancia máxima de 4 entre entradas consecutivas. Esta distancia es un compromiso entre una buena latencia y no gastar demasiadas entradas para este tipo

de tráfico. Se ha asignado un pequeño número de entradas al tráfico best-effort en proporción a su importancia relativa. El resto de entradas se ha asignado al tráfico con requisitos de QoS. Se ha asignado una distancia máxima de 4 y 6 a la voz y al vídeo, respectivamente. Hemos elegido estos valores porque nuestras pruebas han mostrado que son apropiados. Para la carga controlada se ha asignado el mismo número de entradas que para el tráfico de vídeo, pero no se ha tenido en cuenta ninguna restricción en cuanto a distancia máxima a la hora de rellenar la tabla.

Para configurar el planificador MinBW, no se ha asignado ningún peso al CV del tráfico de control de red, ya que, con este planificador, este CV tiene prioridad absoluta sobre los demás. Para poder comparar, al resto de CVs se les ha asignado un peso igual a la proporción de entradas de cada CV en la tabla de arbitraje. Nótese que esto se traduce en el caso del CV de la voz, en una asignación de ancho de banda mayor del que estrictamente requiere, esto es necesario, como ya se ha dicho, para ofrecerle mejores prestaciones en cuanto a latencia.

En ambos casos o bien se ha asignado un 25 % del ancho de banda al tráfico de control de la red (planificador basado en tabla) o se ha dejado sin asignar (planificador MinBW). Teniendo en cuenta que en nuestras pruebas tan sólo inyectamos un 1 % de tráfico de control, esto puede parecer un desperdicio de recursos. Sin embargo, es necesario dejar un cierto margen, primero porque en un caso real no sabemos a priori la cantidad exacta de tráfico de

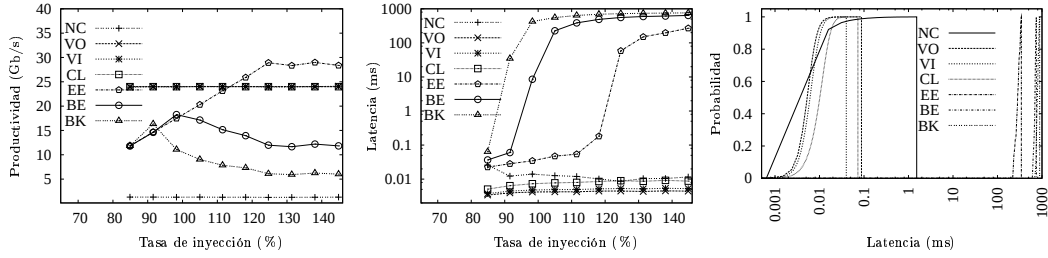


Figura 1: Indices de prestaciones usando el planificador basado en tabla.

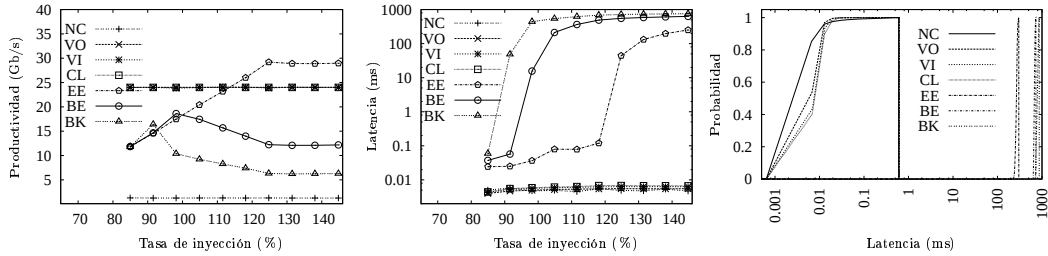


Figura 2: Indices de prestaciones usando el planificador MinBW.

control que se va a inyectar, y segundo porque la productividad de la red no es del 100 %, y por tanto no podemos reservar ancho de banda hasta el máximo teórico de los enlaces. En cualquier caso, este ancho de banda no se desperdicia, ya que será aprovechado por el resto de CVs, en especial por los de best-effort.

Esta configuración de los planificadores responde al modelo de QoS en que, o bien se ha fijado la tabla por adelantado, o bien emula el estado de la red tras un periodo transitorio en el que se han establecido las conexiones y la configuración de los planificadores se ha modificado en concordancia.

4.4. Resultado de las simulaciones

En esta sección se muestran las prestaciones de nuestra propuesta atendiendo a tres índices de QoS: productividad, latencia media, y función de distribución acumulativa (FDA) de la latencia para el punto de máxima carga. Esta función representa la probabilidad de que la transmisión de un paquete tenga una latencia igual o menor que un cierto valor dado, e indi-

rectamente indica también la latencia máxima obtenida. Los resultados obtenidos se muestran en las Figuras 1 y 2.

Respecto a productividad los dos planificadores proporcionan unas prestaciones muy similares. Las SCs de control y QoS obtienen todo el ancho de banda que están inyectando. Sin embargo, cuando la carga de la red es muy alta las SCs de best-effort obtienen un ancho de banda menor del que inyectan. Nótese que este ancho de banda es proporcional a su importancia relativa. Además, puede observarse que la productividad de la red no alcanza el 100 %, situándose en torno al 90 %.

Las SCs de control y QoS obtienen una latencia media pequeña y estable, independientemente de la carga de la red. Con el planificador basado en tabla se obtienen latencias medias menores para los SCs con QoS, pero una latencia peor para el tráfico de control. Además, se observa una clara diferencia entre el tráfico de vídeo y el de carga controlada debido a la diferencia entre las distancias máximas de las entradas de la tabla de sus respectivos CVs. Con el planificador MinBW esta distin-

ción no puede llevarse a cabo si, cómo es el caso, se asigna la misma cantidad de ancho de banda mínimo. Las SCs de best-effort obtienen un tratamiento diferenciado atendiendo a su prioridad.

Los resultados muestran que los dos planificadores cumplen con los requisitos de latencia máxima, 10 ms para la voz y 100 ms para el vídeo, aunque la tabla obtiene máximos más pequeños.

Estos resultados muestran que somos capaces de alcanzar con ambos planificadores nuestros objetivos de QoS: baja latencia para el tráfico de control, garantías de ancho de banda y latencia para el tráfico con QoS y un tratamiento diferenciado al tráfico best-effort.

5. Conclusiones

En este trabajo hemos propuesto cómo usar los mecanismos de AS para proporcionar calidad de servicio (QoS) a las aplicaciones. Para ello, segregamos el tráfico en varias clases de servicio (SCs), que deben ser procesadas adecuadamente por el planificador de salida para obtener sus requisitos. AS define dos planificadores: el planificador basado en tabla de arbitraje de canales virtuales y el planificador de enlace de salida de ancho de banda mínimo (MinBW). En este artículo hemos mostrado cómo configurarlos adecuadamente. Además, si queremos proporcionar garantías de QoS se debe usar un control de admisión de conexiones.

Hemos probado el comportamiento de la metodología propuesta mediante simulación, usando varios tipos de tráfico y paquetes de tamaño fijo. Para poder llevar a cabo un estudio comparativo hemos propuesto una implementación del planificador MinBW que satisface todas las propiedades requeridas. Hemos llamado a este algoritmo Weighted Fair Queuing Credit Aware (WFQCA). Los resultados muestran que, aunque existen diferencias significativas entre el comportamiento de los dos planificadores, ambos son capaces de proporcionar los requisitos de QoS requeridos.

Como trabajo futuro nos centraremos en mejorar el comportamiento del planificador basado en tabla. En concreto, estamos estu-

diando diferentes formas de solucionar el problema que surge al usar paquetes de tamaño variable y en romper el acople existente en la asignación de entradas de la tabla para proporcionar requisitos de ancho de banda y latencia.

Referencias

- [1] Advanced Switching Interconnect Special Interest Group. *Advanced Switching Core Architecture Specification. Revision 1.0*, December 2003.
- [2] F. J. Alfaro, J. L. Sánchez, and J. Duiato. QoS in InfiniBand subnetworks. *IEEE Transactions on Parallel and Distributed Systems*, 15(9):810–823, September 2004.
- [3] S. Blake, D. Back, M. Carlson, E. Davies, Z. Wang, and W. Weiss. An Architecture for Differentiated Services. Internet Request for Comment RFC 2475, Internet Engineering Task Force, December 1998.
- [4] R. Braden, D. Clark, and S. Shenker. Integrated Services in the Internet Architecture: an Overview. Internet Request for Comment RFC 1633, Internet Engineering Task Force, June 1994.
- [5] A. Demers, S. Keshav, and S. Shenker. Analysis and simulations of a fair queuing algorithm. In *ACM SIGCOMM*, 1989.
- [6] IEEE. 802.1D-2004: Standard for local and metropolitan area networks. <http://grouper.ieee.org/groups/802/1/>, 2004.
- [7] A. K. Parekh and R. G. Gallager. A generalized processor sharing approach to flow control in integrated services networks: the single-node case. *IEEE/ACM Transactions on Networking*, 1(3):344–357, 1993.
- [8] S.A. Reinemo, F.O. Sem-Jacobsen, T. Skeie, and O. Lysne. Admission control for diffserv based quality of service in cut-through networks. In *Proceedings of the 10th International Conference on High Performance Computing (HiPC 2003). Hyderabad, India*, pages 118–128, December 2003.