

Resumen

Origen

El proceso de planificación es un proceso importante a la hora de determinar las prestaciones y el rendimiento de un conmutador. En los conmutadores actuales el trato diferenciado del tráfico (*QoS*), ya sea por asignación de prioridades o preestableciendo el reparto del ancho de banda según la clase de tráfico, hace que los algoritmos de planificación sean complejos. Esta complejidad se ve incrementada cuando los algoritmos de planificación tienen que contemplar el tráfico *Multicast*, es decir, aquellos paquetes que van destinados a más de un puerto de salida; consecuentemente la complejidad de estos algoritmos es lo que justifica que el número de clases de tráfico actualmente sea insuficiente para las previstas exigencias de velocidad de transmisión, y que incluso no hayan muchas aportaciones de algoritmos que contemplen tráfico *Multicast*. El tráfico *Multicast* acentúa los problemas que se le presentan al sistema de planificación, pues tiene que ser capaz de sincronizar todos los puertos de E/S, así como la matriz de conmutación, lo que puede resultar un grave problema a medida que aumenta la velocidad de conmutación debido, entre otros problemas, a los relacionados con el *skew* en la señal de reloj. Pero quizás a pesar de la importancia de lo citado, uno de los aspectos prácticamente no contemplados en las propuestas existentes es el tratamiento interno de paquetes de longitud variable en los conmutadores, y mucho menos en los algoritmos de planificación pues la complejidad, ya de por sí alta de estos algoritmos, se ve seriamente incrementada. El soportar paquetes de longitud variable, supondría un mayor aprovechamiento del ancho de banda, pues no se tendrían que fragmentar las tramas a un tamaño fijo, evitando así el tener que añadir por cada segmento la información necesaria para poder tratar y recomponer la trama adecuadamente.

Los conmutadores con colas a la salida presentan unas prestaciones óptimas de retardo-*throughput* para todas las distribuciones de tráfico. Esta característica los convierte en una buena opción en aquellas redes en las que el comportamiento del tráfico sea difícil de predecir, como será en la futura Internet. Sus prestaciones representan la referencia que las características de las arquitecturas existentes intentan emular. Estas arquitecturas con colas a la salida se han utilizado en redes de comunicación debido a su gran uso del ancho de banda, pero debido a las limitaciones de velocidad introducidas por las matrices de conmutación e incluso por las memorias situadas en las salidas, hasta la actualidad se han ido desechando en conmutadores de alta velocidad localizados en el núcleo de las redes de comunicación. Hoy en día la actividad investigadora de grupos con experiencia en el desarrollo de conmutadores, se han volcado en obtener una arquitectura con una técnica pura de almacenaje de colas a la salida en la que se subsanen los clásicos problemas de velocidad. Como resultado de algunas de estas investigaciones, en el seno del IUMA (Instituto de Microelectrónica Aplicada) de la ULPGC (Universidad de Las Palmas de Gran Canaria), se está desarrollando una arquitectura denominada *GMDS* (*Gigabit MultiDrop Switch*), que aporta una solución válida para la implementación de elementos de conmutación con colas a la salida capaces de proporcionar una alta funcionalidad y elevadas prestaciones. La arquitectura referenciada suplanta la matriz de conmutación por un método de transmisión serie punto a multipunto, como se muestra de forma esquemática en la Figura 1.3 para el caso particular de un conmutador 4x4.

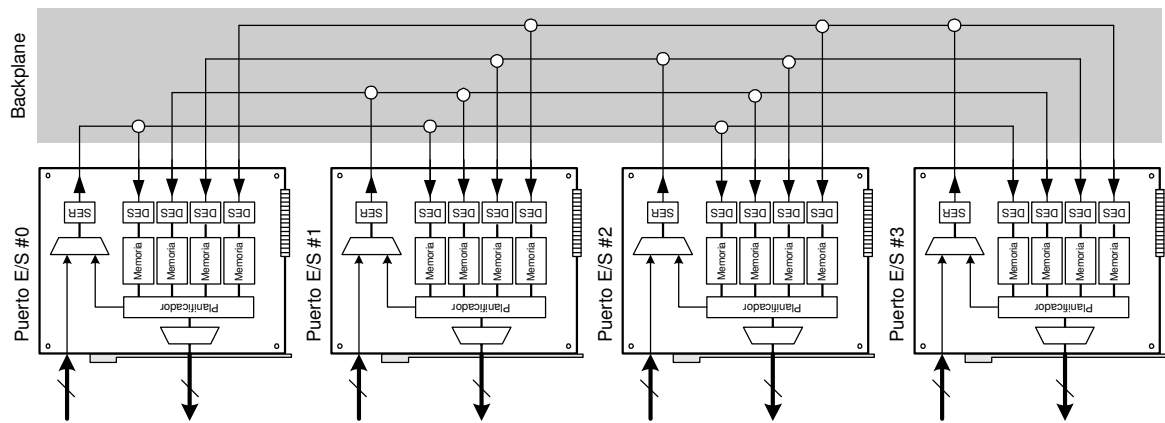


Figura 1: Interconexión E/S a través de un *backplane multidrop* (conmutador 4x4)

Este tipo de arquitectura permite superar las limitaciones de escalabilidad de la arquitectura convencional de un conmutador $N \times N$ con colas a la salida en el que se utiliza una matriz de conmutación para la interconexión entre los puertos de entrada y salida, cuya velocidad, junto con la de los elementos de almacenamiento, debe ser N veces superior a la de los enlaces serie. Para ello, esta arquitectura propone que la interconexión entre los puertos de entrada y los puertos de salida del conmutador se realice a través de un *backplane multidrop* en lugar de utilizar una matriz de conmutación.

El *backplane multidrop* está formado por N enlaces serie activos punto a multipunto, cada uno de ellos dedicado a un único puerto de entrada, y a través de este enlace realizará la transferencia de los paquetes recibidos, así como la información de control de flujo, hacia los correspondientes puertos de destino. Cada uno de los puertos de salida estará conectado a los N enlaces serie del *backplane multidrop*, monitorizando en todo momento la información transmitida a través de cada uno de ellos y extrayendo aquellos paquetes destinados a él. Estos paquetes serán posteriormente almacenados en los elementos de memoria de los puertos de salida. La selección del instante en el que los paquetes almacenados en las diferentes colas deben ser transferidos a la salida del conmutador para garantizar las diferentes calidades de servicio estará determinada por la política de servicio implementada en el planificador. En consecuencia, en esta arquitectura no es necesario que la velocidad de los enlaces serie aumente proporcionalmente con respecto a la velocidad de los enlaces externos en función del número de puertos de entrada del conmutador como pasa en los conmutadores que poseen los elementos de almacenamiento a la entrada. En el peor caso, en el que todos los puertos de entrada envíen un paquete a un mismo puerto de destino, la velocidad de los elementos de memoria localizados en los puertos de salida debe ser igual a la de los enlaces serie del *backplane*, constituyendo una arquitectura a partir de la cual es posible la implementación práctica de conmutadores con colas a la salida reales fácilmente escalables.

Objetivos

A nivel de red, la diferenciación de servicio, es deseable con el fin de poder acomodar los requerimientos de diferentes tipos de tráfico y necesidades de usuario. De esta manera, por ejemplo, los distribuidores de servicios, pueden aplicar diferentes baremos de precios, según el servicio prestado al usuario final.

La implantación de calidad de servicio (*QoS*) es esencial para el éxito de aplicaciones avanzadas como telemedicina, videoconferencia y VoIP (voz sobre IP o telefonía sobre IP). Estas aplicaciones demandan, además de gran ancho de banda, un servicio diferenciado. En muchos casos, es necesario garantizar que la transmisión de los datos se realice sin interrupción o pérdida de paquetes, por lo que la calidad de servicio en la red permitirá al administrador priorizar el flujo del tráfico de red, reservar el ancho de banda para cualquier tipo de demanda e imponerles un límite sobre el total.

Actualmente, tanto las arquitecturas de los conmutadores existentes en el mercado como los algoritmos de gestión que controlan el tráfico, tienen ciertas carencias.

El objetivo de este Proyecto Fin de Carrera era desarrollar un Planificador, parte de la estructura de un conmutador, que por su arquitectura particular, presentase varias ventajas respecto a los conmutadores clásicos actualmente en uso. El Planificador desarrollado tiene en cuenta la arquitectura global del sistema y potencia las características de éste, llegando a constituir un algoritmo *hardware* que contempla *QoS*, justicia, rapidez, y todo ello para tramas de **longitud variable**. A la hora de tomar decisiones, este Planificador es capaz de seleccionar la siguiente trama a transmitir en cinco ciclos de reloj a frecuencias de reloj elevadas, independientemente del número de fuentes que esté gestionando cumpliendo además una serie de características que se detallan posteriormente y que le posibilita poder estar presente de forma satisfactoria en las futuras redes globales de comunicación.

Desarrollo

Inicialmente en el trabajo presentado se analizaron las ventajas y desventajas de las diferentes técnicas de almacenamiento del tráfico en los conmutadores de las redes de comunicación, llegándose a la conclusión de que la técnica de almacenamiento pura con colas a la salida presenta unas prestaciones ideales, en términos de prestaciones y garantía de calidades de servicio, pero con ciertas limitaciones de velocidad de transferencia que la han hecho no estar presente de forma masiva en las redes actuales de comunicación. Los problemas de esta técnica de almacenamiento son debidos fundamentalmente al cuello de botella que representan las matrices de conmutación (*Switch Fabric*), y los elementos de memoria, a la hora de transferir tráfico en un conmutador desde los puertos de entrada a los de salida. En el peor caso, en un conmutador NxN basado en la técnica de colas a la salida (Figura 2) es necesario enviar N células desde los diferentes puertos de entrada a un mismo puerto de salida en un mismo ciclo, para lo cual la transferencia de la información a través de la matriz de conmutación y posterior almacenaje debe ser N veces más rápida que la velocidad del enlace externo (Figura 3).

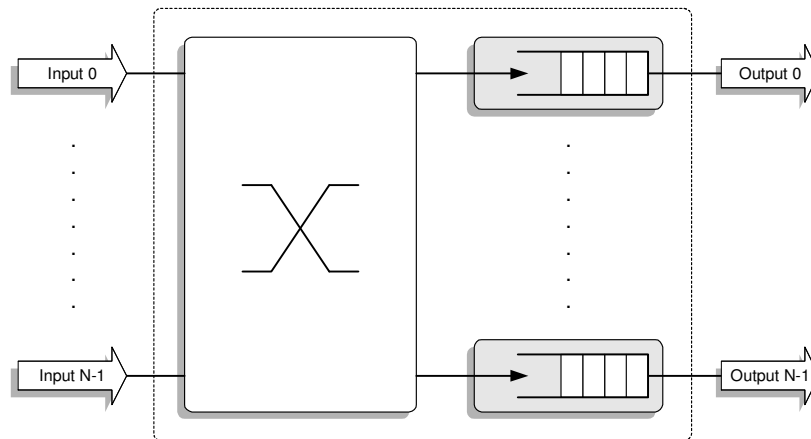


Figura 2: Estructura de un conmutador con colas a la salida

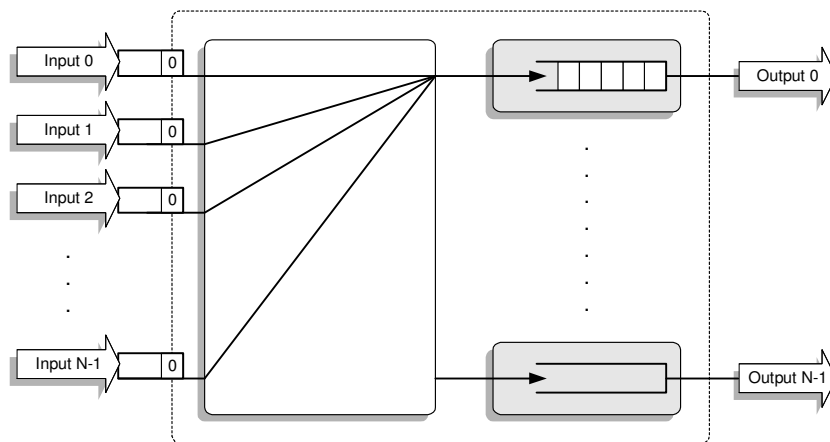


Figura 3: Efecto de N paquetes de entrada a un puerto de destino común en un conmutador con colas

A continuación, se realizó el planteamiento de una arquitectura de referencia que utiliza la técnica de colas a la salida. Esta arquitectura soluciona, en parte, los problemas que las arquitecturas clásicas tenían con los requisitos de las matrices de conmutación. La clave de la solución presentada, radica en que se ha sustituido la matriz de conmutación por una transmisión *multidrop* o punto a multipunto. Este tipo de enlace a altas velocidades se ha podido realizar gracias al *Backplane ATH desarrollado*, que permite transmitir en configuración *multidrop* a tasas de transferencia del orden de varios *Gbps*. Véase la patente y artículos publicados con respecto a lo comentado en la información incluida como anexos en la documentación entregada.

Seguidamente y puesto que se pretendía realizar el diseño de un algoritmo de planificación de altas prestaciones para un conmutador de alta velocidad, en este trabajo se analizaron distintos algoritmos existentes con la finalidad de obtener ciertas directrices que ayuden a que la solución aportada tuviese mayor garantías de éxito. No obstante, el éxito de diseño fue verificado posteriormente debido a los resultados obtenidos en las pruebas realizadas.

Finalmente se presentó la arquitectura final alcanzada tras el proceso de diseño de este Proyecto Fin de Carrera y las pruebas finales sobre ésta. La idea de esta arquitectura ha sido madurada, y desarrollada, para que cumpliera con todos los ambiciosos objetivos

buscados y especificados en el siguiente apartado, procurando que al mismo tiempo representara una solución *hardware* óptima.

Conclusiones

Con este PFC se deseaba desarrollar un Planificador para un conmutador que pudiese suponer un avance en el futuro de las redes de comunicación. Por lo tanto, y debido a que los conmutadores de borde en una red de comunicación son elementos muy numerosos éstos deben proporcionar una elevada funcionalidad en términos de *QoS* (*Quality of Service*), control de flujo, etc, no siendo determinante su capacidad de soportar altas tasas de transferencia. Por esta razón se utilizó como referencia para el diseño *hardware* del Planificador el conmutador *GMDS*, capaz de manejar tráfico *Multicast/Broadcast* sin contención, ni pérdidas de eficiencia. El conmutador *GMDS* entra en la categoría de los conmutadores no bloqueantes, pero con la peculiaridad de no presentar contención en los puertos de salida, lo que lo hace aún más eficiente que la mayoría de los conmutadores existentes en el mercado.

El algoritmo realizado en este proyecto Fin de carrera cumplió con los objetivos deseados de manera enormemente satisfactoria. Se consiguió tener un algoritmo **eficiente, escalable, flexible, simple, justo**, que **protege** las fuentes con buen comportamiento de las que no lo posean (fluctuaciones de carga de la red y tráfico sin calidad de servicio). Además, contempla las siguientes características que lo dotan de mayor robustez:

- Contempla un amplio abanico de **clases** de tráfico y **prioridades** de servicio dotando así al algoritmo la posibilidad de trato diferenciado del tráfico que gestiona.
- Posibilidad de tratar tráfico de **control de flujo** con la finalidad de evitar la pérdida de información debido a una saturación momentánea de los enlaces.
- Aprovechando las características de la arquitectura del conmutador en la que está integrado el algoritmo de planificación diseñado se obtiene la capacidad de realizar transmisiones *Unicast/Multicast/Broadcast*.
- En un número reducido de ciclos de reloj el planificador es capaz de iniciar la gestión del tráfico, esta **rápida inicialización** proporciona unas mínimas pérdidas de tramas por parte del planificador en el caso de un *reset* del sistema.
- Capacidad **hot swapping** que proporciona al conmutador donde esté integrado el planificador la capacidad de conexión y desconexión en caliente de las fuentes conectadas al sistema de conmutación.
- Capacidad de manejar **tramas de longitud variable**, a fin de evitar la segmentación e identificación de los paquetes IP de longitud variable en células de tamaño fijo para su transferencia a través del conmutador, y eliminando así la pérdida de ancho de banda asociado a los conmutadores clásicos basados en el uso de matrices de conmutación de paquetes de longitud fija que conllevan un inherente desaprovechamiento del ancho de banda.

■ Algoritmo de planificación **configurable** para que el diseño se pueda probar en dispositivos lógicos programables, de esta forma, se pueden hacer pruebas en el laboratorio con **sistemas reales** de conmutación como GMDS, conmutador en el que está integrado el Planificador diseñado.

■ La capacidad de suministrar a aquellas clases de tráfico que momentáneamente requieran más ancho de banda, el ancho de banda de las clases que no utilicen el que se les haya asignado mediante configuración. Característica que lo convierte en un algoritmo de planificación denominado como **conservativo**.

El algoritmo de planificación diseñado e implementado físicamente en este trabajo, además de constituir una parte de una **arquitectura real** de un conmutador, es válido para poder adaptarse a **cualquier arquitectura de conmutación** basada en **colas a la salida**. Por esta razón, puede estar presente en **todas las redes de datos** en la que puedan situarse los citados conmutadores.

Originalidad

Lo novedoso de este trabajo radica en la realización de un algoritmo *hardware* de planificación de elevadas prestaciones para una real e innovadora arquitectura de conmutación de alta velocidad que aún está siendo desarrollada y cuya génesis se sustenta en una idea patentada de su arquitectura que se incluye en los anexos de esta documentación. Para esta arquitectura no existía ningún otro algoritmo de planificación. Sin embargo, este algoritmo de planificación es capaz adaptarse a cualquier arquitectura existente con colas puras de almacenamiento a la salida.

La novedosa arquitectura de conmutación denominada *Gigabit MultiDrop Switch (GMDS)*, Figura 2, ha sustituido la clásica matriz de conmutación (*Switch Fabric*), elemento central en los conmutadores clásicos y que constituyen una limitación del *throughput*, por una transmisión *multidrop* o punto a multipunto, que aunque no es un concepto nuevo el grupo de desarrollo de la arquitectura ha conseguido reducir las limitaciones que imponían las longitudes de las líneas y las reflexiones de la señal en transmisiones a elevadas frecuencias de trabajo, logrando de esa manera multiplicar el *throughput* entre puertos. Véase la Figura 1 y los artículos incluidos en los anexos de esta documentación.

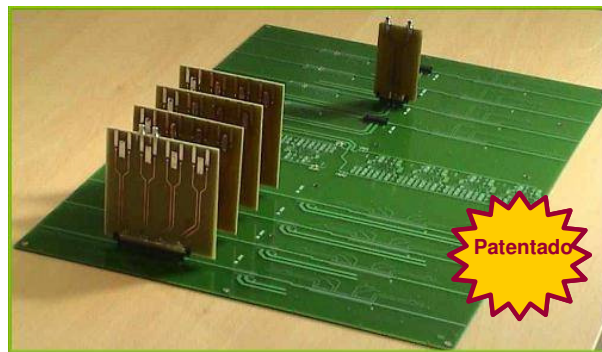


Figura 1: Backplane del GMDS

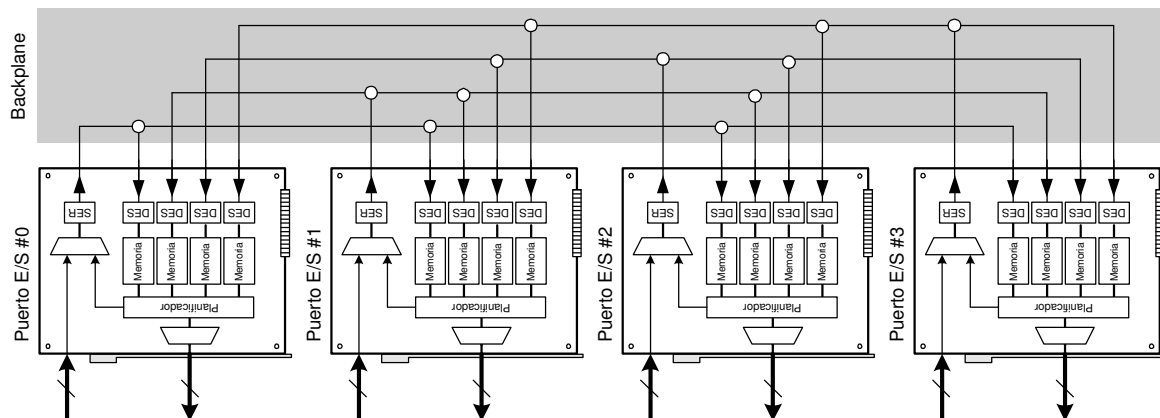


Figura 2: Interconexión E/S a través de un backplane multidrop (GMDS 4x4)

A continuación se presentan las características que según los estudios previos realizados debe tener un buen algoritmo de planificación y cómo se contemplan en el Planificador diseñado:

■ **Eficiencia:** el algoritmo diseñado tiene un alto grado de eficiencia tal y como se demuestra en el apartado de resultados.

■ **Escalabilidad:** el diseño presenta una gran escalabilidad puesto que es muy independiente del número de fuentes que puede llegar a gestionar. Permitiendo así que el sistema sea fácilmente escalable, lo que supone minimizar el coste *hardware* asociado a la escalabilidad de una arquitectura.

■ **Flexibilidad:** la arquitectura del planificador presentado ha sido planteada de forma modular, separando los bloques susceptibles de cambios de la arquitectura general del sistema, a fin de favorecer cualquier modificación del comportamiento del diseño para adaptarlo a futuras necesidades.

■ **Protección:** el algoritmo diseñado protege satisfactoriamente a las fuentes con buen comportamiento de las que no lo posean (fluctuaciones de carga de la red y tráfico sin calidad de servicio).

■ **Simplicidad:** el algoritmo diseñado realiza un complejo proceso de gestión basándose en una arquitectura con la simplicidad necesaria para permitir la implementación en redes de alta velocidad.

■ **Justicia:** el algoritmo diseñado es justo en el reparto de ancho de banda entre las fuentes conectadas al conmutador, siendo éste proporcional al peso de las clases de tráfico que transmitan. Este algoritmo realiza un reparto de manera que no se produzcan ráfagas, a fin de obtener un mayor rendimiento.

Además, se ha provisto al algoritmo diseñado de las siguientes características para darle mayor robustez al diseño.

■ Contempla un amplio abanico de **clases** de tráfico y **prioridades** de servicio dotando así al algoritmo la posibilidad de trato diferenciado del tráfico que gestiona. El número de clases para las que se diseñó el planificador son 16, lo cuál supone una cantidad de clases de tráfico más que suficiente para satisfacer los requerimientos futuros de las redes de comunicación. Además, a las 16 clases de tráfico se les puede asignar una de 4 posibles prioridades del sistema.

■ Posibilidad de tratar tráfico de **control de flujo** con la finalidad de evitar la pérdida de información debido a una saturación momentánea de los enlaces. El control de flujo tiene máxima prioridad respecto al resto de clases de tráfico.

■ Aprovechando las características de la arquitectura del conmutador en la que está integrado el algoritmo de planificación diseñado se obtiene la capacidad de realizar transmisiones *Unicast/Multicast/Broadcast*.

■ En un número reducido de ciclos de reloj el planificador es capaz de iniciar la gestión del tráfico, esta **rápida inicialización** proporciona unas mínimas pérdidas de tramas por parte del planificador en el caso de un *reset* del sistema.

- Capacidad *hot swapping* que proporciona al conmutador donde esté integrado el planificador la capacidad de conexión y desconexión en caliente de las fuentes conectadas al sistema de conmutación.

- Capacidad de manejar **tramas de longitud variable**, a fin de evitar la segmentación e identificación de los paquetes IP de longitud variable en células de tamaño fijo para su transferencia a través del conmutador, y eliminando así la pérdida de ancho de banda asociado a los conmutadores clásicos basados en el uso de matrices de conmutación de paquetes de longitud fija que conllevan un inherente desaprovechamiento del ancho de banda.

- Algoritmo de planificación **configurable** para que el diseño se pueda probar en dispositivos lógicos programables, de esta forma, se pueden hacer pruebas en el laboratorio con **sistemas reales** de conmutación como GMDS, conmutador en el que está integrado el Planificador diseñado.

- La capacidad de suministrar a aquellas clases de tráfico que momentáneamente requieran más ancho de banda, el ancho de banda de las clases que no utilicen el que se les haya asignado mediante configuración. Característica que lo convierte en un algoritmo de planificación denominado como **conservativo**.

Contemplar todas las características citadas entraña una gran dificultad de diseño e implementación debida a las limitaciones impuestas por la singularidad de la arquitectura y por las señales de su interfaz de comunicación.

Por otro lado, hay algoritmos ampliamente utilizados en arquitecturas reales que soportan un reducido espectro de calidades de servicio de todo punto insuficiente para satisfacer las necesidades de las futuras redes de comunicación, donde los elementos de conmutación deben soportar diferentes *QoS* para que la transferencia de paquetes a través de la red sea constante y predecible. Este es uno de los aspectos, que a juicio de la gran mayoría de los analistas, determinará el futuro desarrollo de Internet. Por lo que resulta imprescindible encontrar un algoritmo cuya implementación permita obtener unas prestaciones óptimas. Por citar unos ejemplos podríamos citar el algoritmo *PIM (Parallel Iterative Matching)* propuesto por Anderson, Owicki, Saxe y Thacker y sobre todo el algoritmo *iSLIP*, presentado por *McKeown*, siendo éstos utilizados en multitud de conmutadores comerciales, como la serie 12000 del conmutador *Gigabit Switch Router de Cisco Systems* y el *Versalar 25000 de Nortel Networks*.

En conclusión, teniendo en cuenta lo anterior y después de haber presentado la amplia cantidad de características que el algoritmo de planificación diseñado contempla de forma conjunta en una arquitectura de conmutación real, de por sí innovadora, para la que no existía un algoritmo de planificación previo, se puede considerar que Planificador realizado para el conmutador *GMDS* es de notable originalidad.

Aplicabilidad

Los conmutadores con colas a la salida presentan unas prestaciones óptimas de retardo-*throughput* para todas las distribuciones de tráfico. Esta característica los convierte en una buena opción en aquellas redes en las que el comportamiento del tráfico sea difícil de predecir, como será en la **futura Internet**. Sus prestaciones representan la referencia que las características de las arquitecturas existentes intentan emular. Estas arquitecturas con colas a la salida se han utilizado en redes de comunicación debido a su óptimo uso del ancho de banda, pero debido a las limitaciones de velocidad introducidas por las matrices de conmutación e incluso por las memorias situadas en las salidas, hasta la actualidad se han ido desechando en conmutadores de alta velocidad localizados en el núcleo de las redes de comunicación.

Hoy en día la actividad investigadora de grupos con experiencia en el desarrollo de conmutadores comercialmente disponibles, se han volcado en obtener una arquitectura con una técnica pura de almacenamiento de colas a la salida en la que se subsanen los clásicos problemas de velocidad. Como resultado de algunas de estas investigaciones, en el seno del IUMA (Instituto de Microelectrónica Aplicada) de la ULPGC (Universidad de Las Palmas de Gran Canaria), se está desarrollando una arquitectura denominada GMDS (*Gigabit MultiDrop Switch*), que aporta una solución válida para la implementación de elementos de conmutación con colas a la salida capaces de proporcionar una alta funcionalidad y elevadas prestaciones. Esta arquitectura, desarrollada en el trabajo presentado, ha servido de referencia para la realización del Planificador diseñado.

El algoritmo de planificación diseñado e implementado físicamente en este trabajo, además de constituir una parte de una **arquitectura real** de un conmutador, es válido para poder adaptarse a **cualquier arquitectura de conmutación** basada en **colas a la salida**. Por esta razón, puede estar presente en **todas las redes de datos** en la que puedan situarse los citados conmutadores.

Resultados

El algoritmo de planificación al ser descrito a nivel *hardware* se le ha podido dotar de la capacidad de poder **cumplir con las todas las características** descritas en la sección denominada “Originalidad”, incluso gestionando tramas de tamaño mínimo, que en el caso del conmutador *GMDS* tomado como referencia para este trabajo es equivalente a **5 ciclos de reloj**. Es decir, gracias a un **complejo sistema de control** llevado a cabo por varias máquinas de estados y diversos módulos, el Planificador es capaz de seleccionar en 5 ciclos de reloj la siguiente trama a enviar al interfaz de salida del conmutador. Por ejemplo, si tuviese que gestionar **128 fuentes** o puertos de entrada, el planificador sería capaz de decidir que trama sería la siguiente en ser transmitida al interfaz de salida del conmutador entre las 2048 tramas posibles (128 fuentes x 16 clases de tráfico) conservando todas las características con las que se diseñó y se detallaron el apartado de “Originalidad”.

Por otro lado, el sistema de verificación de un Planificador es un sistema complejo debido a la naturaleza aleatoria del tráfico gestionado y que además puede ser de tamaño variable, característica que complica la generación de datos y planificación de éstos. Por esa razón, se desarrolló un **entorno de simulación robusto de dimensión considerable y configurable que se encargó de realizar automáticamente la verificación funcional del sistema global** mediante un **elevado número de simulaciones funcionales** en las que se variaron las condiciones de tráfico generado. Al finalizar cada simulación, las herramientas generaban informes que posteriormente fueron analizados. Debido a que el número de **informes generados** asciende a un total de 2520 y como la cantidad de información que suministran es abrumadora, se analizó los datos y se representó gráficamente los resultados más relevantes para ratificar que se habían cumplido los objetivos buscados. No obstante, aunque las **pruebas globales** del sistema fueron numerosas, además, se realizaron **pruebas específicas** de casos que no se pudiesen dar con mucha probabilidad en una verificación con tráfico aleatorio como el creado para los *test* automáticamente realizados por las herramientas de simulación.

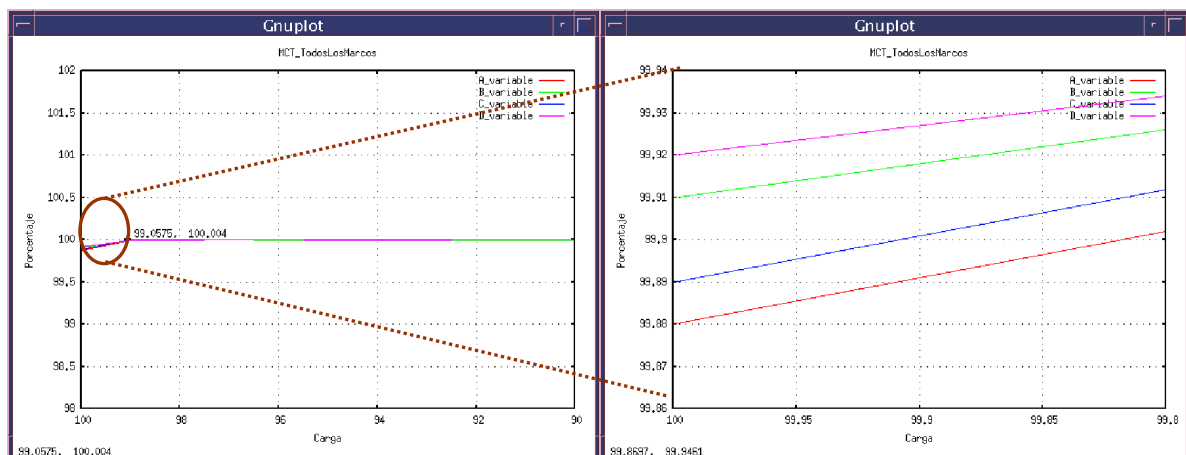


Figura 1: Gráfica de la eficiencia de gestión del Planificador

En la gráfica anterior se puede observar un **ejemplo de las gráficas** obtenidas de una de las simulaciones funcionales realizadas al planificador y utilizadas para analizar la eficiencia de gestión de éste al trabajar con tramas de longitud variable. En el eje de abscisas se puede apreciar las diferentes cargas de tráfico creadas por el generador de datos del sistema de verificación. En el eje de ordenadas se representa el porcentaje de

datos que el Planificador había gestionado hasta el momento en el que se detuvo la simulación. Previamente a que las tramas sean planificadas son creadas y almacenadas por el sistema de verificación, y se mantendrán a la espera de que el planificador las solicite al sistema de *test* para enviarlas al interfaz de salida del conmutador.

Se han realizado tres tipos de simulaciones denominadas **TCT**, **MCT** y **UCT**. En las simulaciones del tipo *TCT (Todas las Clases con Tráfico)* se han generado tramas de todas las clases de tráfico contempladas en el diseño, a fin de determinar si el Planificador distribuye correctamente el ancho de banda configurado para cada clase de tráfico. Sin embargo, las simulaciones *MCT (Mismas Clases con Tráfico)* y *UCT (Única Clase con Tráfico)*, se han planteado con la finalidad de determinar cuánto de conservativo es el algoritmo diseñado, es decir, la capacidad que tiene de suministrar a aquellas clases de tráfico que momentáneamente requieran más ancho de banda, el ancho de banda de las clases que no utilicen el que se les haya asignado mediante configuración. Los datos obtenidos en el tipo de simulación *MCT* muestran que el Planificador reparte el ancho de banda de salida no utilizado entre las clases que tengan tráfico y estén siendo planificadas, de forma proporcional al peso que tengan éstas en el ancho de banda global de salida. Por otro lado, los datos obtenidos en el tipo de simulación *UCT* indican que, incluso en el hipotético caso de que sólo exista una única clase de tráfico, y que ésta tenga asignado una pequeña porción del ancho de banda total de un enlace, por ejemplo, un 3%, el Planificador será capaz de suministrarle el 100% del ancho de banda de salida, de forma que no se encuentre ocioso mientras exista tráfico que gestionar. Además, de las distintas variaciones presentadas en *MCT*, *MCT* y *UCT* en lo que respecta a clases de tráfico activas, se han variado las fuentes que participaban en la simulación. Por tanto, las simulaciones que se denominan en las figuras como:

- **A:** se habrá simulado con las mismas fuentes durante toda la simulación.
- **B:** se habrá simulado *variando las fuentes* con tráfico durante *la simulación*.
- **C:** se habrá generado tráfico de *una única fuente*.
- **D:** se habrá simulado generando tráfico de *todas las fuentes*.

Los tipos de simulaciones *MCT* y *UCT* son situaciones de simulación planteadas para someter al planificador, aún más, a unas extremas condiciones de prueba. Estas situaciones son extremas pues en las redes de comunicación el tráfico debe ser conformado para asegurar que cada tipo de tráfico no sobrepase su límite de BW asignado, pero en el caso de que no fuese así, estas simulaciones demuestran que el Planificador sería capaz de gestionar todo el tráfico aún en esas condiciones. Nótese que mientras menos fuentes y clases de tráfico hayan con tramas, más desaprovechadas estarán las memorias del sistema de almacenamiento y más eficiente debe ser el Planificador para que no se desborden las memorias cuando se evalúan estas situaciones extremas.

Para la realización del Planificador, objeto de este trabajo, se ha seguido una metodología de diseño *bottom-up* altamente estructurada utilizando el lenguaje de descripción *hardware(HDL) Verilog*, siguiendo las reglas y recomendaciones existentes para realizar código sintetizable, garantizando de este modo, que se pudiese llegar a la **implementación física** del algoritmo de planificación realizado.

Para conseguir las ambiciosas prestaciones y todos los objetivos buscados con el complejo diseño realizado se hizo indispensable llegar a un **nivel de descripción hardware** del algoritmo de planificación que permitiese tener **control sobre la arquitectura resultante** después del proceso de síntesis del diseño. La necesidad de una descripción *hardware* a un nivel de detalle como la expuesta en este trabajo, es el resultado de implementar un algoritmo de Planificación para una arquitectura real de conmutación donde existen estrictas restricciones temporales que deben cumplirse sin mermar la eficiencia del proceso de planificación y procurando que presentara una **solución hardware óptima** que permitiese implementar el algoritmo en un dispositivo lógico programable.

No obstante, la arquitectura desarrollada concentra las decisiones en determinados bloques. El caso contrario dificultaría cualquier tipo de modificación en posteriores versiones, o incluso, la corrección de errores en los procesos de verificación. Además, toda la arquitectura, incluyendo los módulos de control del sistema, han sido cuidadosamente distribuidos procurando favorecer las fases posteriores a la fase de síntesis, es decir: planificación espacial (*Floor-Planning*), posicionamiento (*Placement*) y conexonado (*Routing*).

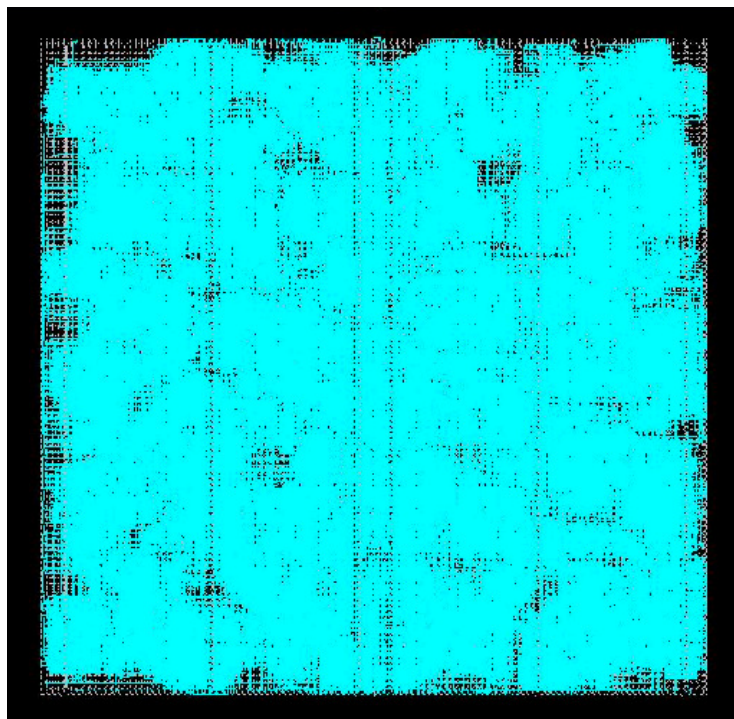


Figura 2: Detalle de ruteado del Planificador en la FPGA modelo xc2v8000-5-ff1152 de la familia Virtex II PRO.

Finalmente indicar que la **documentación** presentada como memoria de este trabajo constituye una guía de referencia especificada a un **nivel de detalle** muy alto que favorece las futuras labores de investigación y desarrollo sobre el sistema de Planificación.