

Evaluación de una alternativa para aumentar el número de puertos de los conmutadores

Juan A. Villar, Francisco J. Andújar, José L. Sánchez, Francisco J. Alfaro¹ y José Duato²

Resumen— En las redes de interconexión basadas en conmutadores, el aumento del número de puertos de un conmutador conlleva una reducción del número total de componentes de la red, lo que provoca una reducción significativa del coste total del sistema. Además, los conmutadores *high-radix* suponen una alternativa interesante para mejorar el rendimiento de la red en términos de latencia pues reducen el número de saltos de los paquetes. Sin embargo, existen varios problemas relacionados con la escala de integración para diseñar estos conmutadores en un único chip. En este artículo se explica y evalúa una alternativa de construcción de conmutadores *high-radix* más allá de los límites permitidos por la escala de integración actual. La idea consiste básicamente en combinar varios conmutadores, cada uno en un chip, obteniendo un conmutador con un número de puertos agregado mayor. Esta propuesta es independiente de la evolución de los conmutadores de un único chip y seguirá siendo válida conforme la escala de integración continúe evolucionando. Los resultados de simulación han mostrado que con un diseño adecuado del conmutador, las redes construidas con conmutadores que implementan esta alternativa pueden alcanzar un rendimiento similar al rendimiento de redes construidas con conmutadores en un chip que proporcionen el mismo número de puertos, pero dichos conmutadores no existen en la actualidad en el mercado porque la escala de integración no permite construirlos en un único chip.

Palabras clave— Conmutadores High-Radix, Redes de Altas Prestaciones, Evaluación de Rendimiento.

I. INTRODUCCIÓN

Las redes de interconexión son un componente clave para una amplia gama de sistemas multiprocesador que van desde los supercomputadores a chips multinúcleo. Las redes de alto rendimiento son esenciales para estos sistemas, donde se requiere una fiabilidad alta, además de grandes tasas de transferencia y latencias muy bajas. A menudo, la red de interconexión es el subsistema que requiere un diseño personalizado. Por ejemplo, el superordenador Tianhe-1A [1], número uno del Top500 (noviembre 2011), está compuesto por procesadores comunes, y una red de interconexión personalizada. El diseño de dicha red ha eliminado los cuellos de botella contribuyendo significativamente al rendimiento global del Tianhe-1A.

El diseño de la red está determinado por la tecnología de integración cuyos avances han mejorado sustancialmente el rendimiento de sus componentes básicos: enlaces y conmutadores. Estos últimos son responsables de gran parte del nivel de prestaciones de la red, por lo que son objeto de mayor investiga-

ción. Uno de los principales parámetros que caracterizan a los conmutadores es su número de puertos (o grado), el cual tiene una fuerte influencia en el coste, consumo y rendimiento en todo el sistema.

Dado un sistema multiprocesador con un número elevado de terminales conectados, aumentar el grado de los conmutadores resulta en una disminución en el número de conmutadores y enlaces de red. Puesto que el coste de la red es proporcional al número de conmutadores, es evidente que el coste se reduce mediante el uso de conmutadores de mayor grado. Por otra parte, el consumo total de la red también se reduce considerablemente, ya que es directamente proporcional al número de conmutadores en la red.

Con respecto al rendimiento, por ejemplo, es evidente que en términos de latencia el uso de conmutadores con más puertos implica una reducción en el tiempo medio para transferir datos a través de la red. En particular, con menos conmutadores para conectar el mismo número de terminales se reduce el número de saltos y el número de posibles colisiones de paquetes en la red, y por lo tanto el tiempo que ellos emplean para llegar a sus destinos. Además, con menos conmutadores, el tiempo de procesamiento total de los paquetes en los conmutadores a lo largo de sus caminos también se reduce.

Por lo tanto, el diseño de conmutadores con un elevado número de puertos es una opción atractiva para mejorar el rendimiento y reducir el coste de la red de interconexión, especialmente para sistemas multiprocesador de gran tamaño. Sin embargo, hay algunos problemas en el diseño de éstos. Uno de ellos está relacionado con la complejidad de su lógica pues se vuelve más compleja a medida que aumenta el grado [2]. El equilibrio entre el coste y la eficiencia no es fácil de tratar, requiriendo un estudio profundo sobre esta disyuntiva. Por un lado, el tamaño de algunas estructuras del conmutador crece cuadráticamente con el grado. Por ejemplo, tal es el caso del espacio de los *buffers* [3], o de los planificadores [4]. Por otra parte, las políticas tradicionales de control de flujo también se ven afectadas por el grado del conmutador en dos aspectos [5]: el tiempo *roundtrip* aumenta drásticamente, y las memorias para el almacenamiento de créditos del control de flujo son linealmente dependientes del *roundtrip*. Por otro lado, el número de pines del chip (*pincount*) aumentará lentamente en la próxima década según el ITRS [6], y por tanto el número de puertos aumentará ligeramente. Por otra parte, existen dificultades para aplicar algunas técnicas cuando el número de puertos es alto. Así, *Virtual Output Queuing* (VOQ) [7] es inviable para los conmutadores con un grado alto. Para superar estos problemas, se han propuesto diferentes soluciones pero

¹Departamento de Sistemas Informáticos. Universidad de Castilla-La Mancha. Albacete. España. e-mail: {juanan,fandujar,falfaro,jsanchez}@dsi.uclm.es

²Departamento de Informática de Sistemas y Computadores. Universidad Politécnica de Valencia. Valencia. España. e-mail: jduato@disca.upv.es

en realidad, éstas están posponiendo el problema para las generaciones venideras de conmutadores.

En cualquier caso, las restricciones de tamaño del conmutador están determinadas principalmente por la escala de integración actual y número de pines del chip. Para ir más allá de los límites de la escala de integración, una solución alternativa para la construcción de conmutadores *high-radix* es la combinación de varios conmutadores de grado menor.

La idea principal es implementar conmutadores de m' puertos a partir de varios conmutadores más pequeños de m puertos. Por ejemplo, un conmutador de m' puertos compuesto por dos conmutadores idénticos de m puertos ($m'/2 < m < m'$), interconectados internamente por medio de $m - m'/2$ puertos, empleando los puertos restantes para las conexiones con el exterior (figura 1a). Nótese que esta estrategia seguirá siendo válida conforme la tecnología de integración continúe evolucionando.

Una consecuencia destacable es que el conmutador resultante ya no es homogéneo. Su rendimiento dependerá de la configuración interna. Así, la interconexión de los conmutadores internos puede convertirse en un cuello de botella si éstos tienen que soportar la mayoría del tráfico manejado por el conmutador. Por lo tanto, es esencial minimizar el impacto de este cuello de botella, de lo contrario la latencia de la red aumentará. Así, el patrón de conexión a nivel de conmutador¹ (SCP) se convierte en una decisión de diseño importante en la construcción de este tipo de conmutadores. Un patrón arbitrario probablemente producirá una degradación significativa de prestaciones, y por ello, es necesario determinar el patrón más conveniente para poder extraer el mayor rendimiento del conmutador.

En este artículo, se describe y evalúa esta alternativa para obtener conmutadores *high-radix*. También se discuten cuestiones clave que determinan su rendimiento. De hecho, se mostrará que el SCP y el ancho de banda de comunicación entre los conmutadores internos son cruciales para el comportamiento del conmutador. Se debe alcanzar un compromiso entre ambos aspectos para obtener diseños de conmutadores eficientes.

Este artículo está organizado como sigue: la sección II repasa brevemente las propuestas existentes sobre conmutadores *high-radix*. Tras ello, en la sección III se dan detalles sobre la alternativa propuesta para la construcción de conmutadores *high-radix*, y en la sección IV se incluyen los resultados de la evaluación realizada. Finalmente, se aportan algunas conclusiones en la sección V.

II. TRABAJO RELACIONADO

En esta sección se revisan las propuestas existentes de conmutadores *high-radix* que se han centrado

¹En adelante, se diferenciará entre *patrón de conexión a nivel de red* y *patrón de conexión a nivel de conmutador*. El primero es el patrón de interconexión tradicional utilizado en redes basadas en conmutadores (por ejemplo, la permutación *butterfly* utilizada para conectar los conmutadores en las redes multietapa); y el segundo patrón hace referencia a cómo los puertos externos de un conmutador *high-radix* se mapean en los puertos de los conmutadores internos.

principalmente en resolver problemas con los diseños tradicionales.

El conmutador YARC [8] es el conmutador *high-radix* utilizado por el Cray BlackWidow [9]. Trata de incrementar el número de puertos considerando enlaces más delgados en lugar de enlaces anchos. Los diseños de conmutadores tradicionales con pocos puertos no pueden adaptarse a conmutadores *high-radix* porque los diseños tradicionales emplean una organización centralizada que no escala apropiadamente.

Por otra parte, *Partitioned Crossbar Input Queued* [10] es más reciente y propone una organización interna de conmutadores *high-radix*, y trata con una de las restricciones principales en el diseño de conmutadores *high-radix*: los excesivos requerimientos de memoria.

Con respecto a la alternativa de construcción de conmutadores *high-radix* combinados por dos conmutadores *low-radix*, el conmutador Sun Blade 6048 Infiniband QDR Switched Network Express Module (NEM) [11] ya implementa esta estrategia que permite conectar hasta 12 *blades* duales en un único *shelf*. Cada NEM suministra 12 conexiones por cada uno de los dos conmutadores InfiniScale IV de 36 puertos. En total, proporciona 24 conexiones para comunicarse con los dos nodos por cada *blade*, y utiliza 9 puertos para que los dos conmutadores internos se comuniquen entre ellos. Los 30 puertos restantes (15 por InfiniScale IV) se usan como enlaces con otros NEMs, o conmutadores externos.

Por otra parte, la topología Dragonfly [12] utiliza un grupo de conmutadores como *router* virtual para incrementar el grado efectivo de la red. Lamentablemente, no se aporta ningún análisis formal del tráfico que cruza el *router* virtual.

En resumidas cuentas, y hasta donde tenemos conocimiento, no hay estudios formales publicados sobre la obtención del SCP óptimo.

III. CONMUTADORES HIGH-RADIX MEDIANTE CONMUTADORES LOW-RADIX

Como se ha mencionado, es posible construir conmutadores *high-radix* combinando varios conmutadores *low-radix*. Esta estrategia hace posible adelantarse a la tecnología de integración y acortar drásticamente el *time-to-market*. Nótese que esta estrategia seguirá siendo válida conforme la tecnología de integración continúe evolucionando.

Esta estrategia abre una serie de nuevos problemas que deben estudiarse. En las siguientes secciones, se repasan brevemente estos problemas que han sido analizados en [13] de un modo más formal. No obstante, en este artículo también se realiza una cuantificación experimental de su influencia en el rendimiento de la red.

A. Conmutadores Combinados

En esta sección se formalizan los conmutadores combinados mediante una definición general. Tras ella, la atención se centra en una subclase particular de conmutadores combinados la cual se usará para mostrar las características de estos conmutadores y

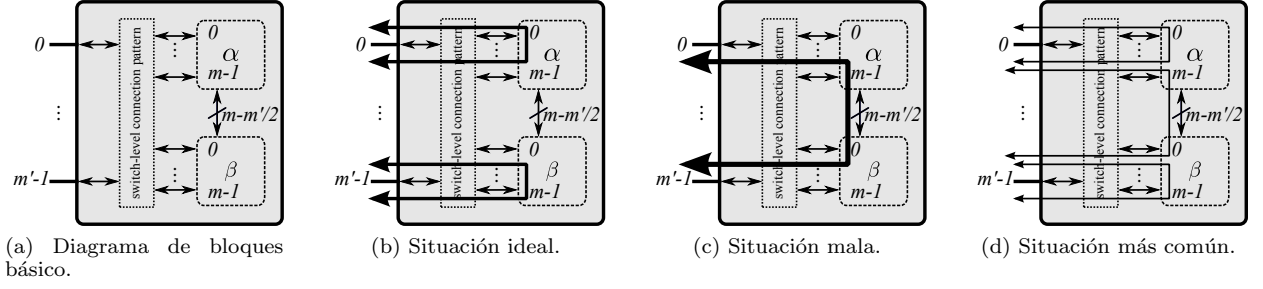


Fig. 1: T -switch: diagrama de bloques y posibles situaciones.

para evaluar el rendimiento de las redes construidas con estos conmutadores.

Definición 1: Un *conmutador combinado*, o simplemente *C-switch*, es un conmutador formado por varios conmutadores (conmutadores internos) más pequeños interconectados. Los puertos ofrecidos por el *C-switch* se obtienen a partir de los puertos libres de los conmutadores internos después de que se hayan interconectado entre sí.

Esta definición es muy genérica porque no especifica ni la cantidad de conmutadores internos ni cuál es su grado. Por tanto, cualquier conmutador obtenido por combinación de otros conmutadores de menor grado cae dentro de esta categoría. Sin embargo, existen varias dificultades para construir *C-switches* que tengan varios conmutadores internos y un grado alto de heterogeneidad.

Para que la latencia interna sea baja, es preferible utilizar una topología completamente conectada para la interconexión de todos los conmutadores internos. Conforme el número de conmutadores internos crezca, el número de puertos dedicados a las conexiones entre los conmutadores internos crecerá tan rápidamente como el número de puertos destinados a la interconexión exterior decrecerá, de modo que esta forma de construir conmutadores *high-radix* perderá interés. Por tanto, parece razonable que el número de conmutadores internos no sea muy alto.

Por otra parte, un diseño interno sencillo del *C-switch* se puede alcanzar cuando todos los conmutadores internos son idénticos. Aunque este aspecto no es tan restrictivo como el número de conmutadores internos, es recomendable que todos los conmutadores internos tengan el mismo grado.

Un caso interesante es aquel donde los *C-switches* se construyen a partir de dos conmutadores internos idénticos. Esta subclase de *C-switches* todavía ofrece un incremento importante en el número de puertos mientras la topología entre los conmutadores internos es la más simple.

Definición 2: Un *conmutador gemelo*, o simplemente *T-switch* (*Twin*), es un conmutador formado por dos conmutadores internos idénticos de menor grado interconectados entre sí. Los puertos obtenidos por el *T-switch* se obtienen a partir de los puertos libres de sus dos conmutadores internos después de que éstos se hayan interconectado entre sí.

Considerando que los dos conmutadores internos y el *T-switch* tienen respectivamente m y m' puertos, la figura 1a muestra un diagrama básico de bloques

de un T -switch, donde los conmutadores internos se nombran como α y β . Aunque los T -switches parecen simples, hay retos significativos en su diseño. Dos de ellos destacan especialmente: (1) obtener el SCP apropiado para la estructura interna del T -switch, (2) determinar el número adecuado de puertos para interconectar los conmutadores internos α y β .

El SCP tiene una influencia importante en la latencia de los paquetes. El tiempo empleado por los paquetes para cruzar un T -switch será mínimo cuando sólo crucen uno de los conmutadores internos (α o β). La figura 1b muestra esta situación. Las flechas representan los caminos seguidos por los paquetes y el ancho de la flecha es proporcional a la cantidad de caminos. El peor caso es aquel en el que todos los caminos cruzan los dos conmutadores internos en el T -switch (figura 1c). Obtener el mejor caso no es trivial y requiere un estudio en profundidad donde deben considerarse varios factores como por ejemplo, la topología de la red, el algoritmo de encaminamiento y el patrón de tráfico. En [13] se muestra formalmente como se obtiene el patrón de conexión óptimo considerando dichos factores.

Respecto al segundo reto, el número de puertos internos debe ser tal que evite la aparición de un cuello de botella entre los conmutadores internos α y β . Obviamente, el número de puertos y el SCP tiene una clara interdependencia.

Las situaciones mostradas en las figuras 1b y 1c son apropiadas para mostrar los retos de diseño de los T -switches, pero la situación más común es la mostrada en la figura 1d, donde los casos anteriores coexisten. Es esta situación, el objetivo principal cuando se diseña un SCP para los T -switches es minimizar el uso de los puertos que interconectan los conmutadores internos.

Por lo tanto, puesto que en muchos casos la comunicación necesitará usar ambos conmutadores internos (eso es, un camino que cruce un T -switch pasará por los conmutadores α y β), se tiene que evitar que la interconexión entre los conmutadores α y β se convierta en un cuello de botella. Además, se debe determinar el número de puertos adecuado para cada conmutador interno. Está claro que cuanto mayor sea el número de puertos usados para interconectar los conmutadores internos menor será la probabilidad de que aparezca un cuello de botella en estos puertos. Sin embargo, y tal como se ha mencionado anteriormente, conforme el número de puertos dedicados a interconectar los conmutadores α y β aumenta, el

grado del T -switch disminuye. Se deberá encontrar un compromiso entre ambos aspectos. Nótese que se está asumiendo que todos los puertos proveen el mismo ancho de banda (de otra forma en lugar de usar el número de puertos se debería considerar el ancho de banda total agregado).

En resumen, la configuración interna de los T -switches se convierte en un aspecto clave en el diseño de esta clase de conmutadores *high-radix*. La configuración óptima de un T -switch depende de las condiciones bajo las que esté funcionando, es decir: tipo de red, topología de red, algoritmo de encaminamiento, patrón de tráfico, etc. A continuación se esboza una metodología para obtener la mejor configuración de esta clase de conmutadores cuando se usan en redes de interconexión de altas prestaciones. El propósito de esta metodología es determinar el SCP óptimo para los T -switches.

B. Metodología para Configurar Conmutadores Combinados

Nuestra metodología para determinar el SCP óptimo para C -switches consiste en los siguientes pasos:

- Análisis de los caminos en la red. El propósito de este paso es determinar las conexiones necesarias en cada C -switch a nivel de red y la cantidad de veces que se utilizan todas ellas teniendo en cuenta todos los posibles caminos usados por los paquetes.
- Clasificación del conmutador. Dependiendo de las características y carga de la red, se pueden obtener una o varias configuraciones diferentes para los C -switches. En este paso, los C -switches se agrupan según los requisitos de conexión, y por tanto, se distinguirán varios tipos de C -switches.

Como resultado del paso anterior, puede suceder que varias de las posibles conexiones en el C -switch soporten uno o más caminos, y sin embargo, al mismo tiempo existan conexiones que nunca se establezcan.

En la topología *fat-tree*, por ejemplo, los C -switches de diferentes etapas pueden requerir distintos SCPs, y lo mismo puede ocurrir con los C -switches de la misma etapa. Cuando se dé un patrón de tráfico simple y se emplee un algoritmo de encaminamiento que balancee la carga, es probable que todos los C -switches de la red puedan configurarse con el mismo SCP.

- Configuración del conmutador. A partir de los requisitos de conexión y dado un número de conmutadores internos de un C -switch, este último paso consiste en encontrar el SCP óptimo para cada tipo de C -switch, intentando minimizar el uso de los enlaces que interconectan los conmutadores internos.

En [13] se ha aplicado dicha metodología a un caso particular: topología k -ary n -tree, algoritmo de encaminamiento determinista, y patrón de tráfico uniforme. Como resultado, se ha obtenido la configuración óptima para todos sus C -switches.

IV. EVALUACIÓN DE RENDIMIENTO

En esta sección se evalúa el potencial de los C -switches estudiando los T -switches. La evaluación se ha realizado por simulación y se ha centrado en los dos aspectos clave que se discutieron anteriormente: el SCP, y el ancho de banda disponible entre los conmutadores internos.

En las siguientes subsecciones, en primer lugar se describe el modelo de red simulado, y a continuación se establecen las diferentes configuraciones de red y los SCPs empleados en las simulaciones. Finalmente, se aportan y analizan en detalle los resultados de las simulaciones obtenidos.

A. Modelo Simulado

Para llevar a cabo la evaluación se ha utilizado un simulador detallado conducido por eventos que es capaz de modelar distintos tipos de redes basadas en conmutadores, y en particular, T -switches.

Se ha elegido la topología k -ary n -tree que es una subclase particular de los *fat-trees*. Se ha seleccionado porque actualmente es una de las más usadas entre los supercomputadores del Top500.

Se ha implementado el algoritmo determinista DESTRO [14] a nivel de red. Se ha considerado este algoritmo por sus características: una implementación en hardware sencilla, tiempo de encaminamiento reducido y reparto de paquetes en orden. Además, DESTRO es capaz de balancear uniformemente el tráfico en la red y de reducir la contención en la red. Como carga de red se ha asumido un patrón de tráfico uniforme porque es uno de los patrones clásicos utilizados en este tipo de evaluaciones.

Con respecto a los conmutadores, según lo mencionado en la sección III, se han simulado T -switches. En este estudio de evaluación, debido a los fuertes requisitos temporales sólo se han simulado T -switches de 32 puertos. Estos conmutadores se forman con dos conmutadores internos de 24 puertos cada uno, de los cuales 16 puertos se utilizan para la comunicación del T -switch con el exterior y 8 puertos se destinan para la comunicación de los dos conmutadores internos ($m = 24$ y $m' = 32$ en la figura 1). Se han considerado conmutadores internos de 24 puertos porque están disponibles en el mercado (por ejemplo, el chip InfiniScale III [15]). No obstante, actualmente la herramienta de simulación se está mejorando para que soporte conmutadores de mayor grado.

Se han evaluado redes de distinto tamaño. Debido a la limitación de espacio, aquí sólo se han incluido los resultados para las redes de 4096 terminales.

Otras suposiciones son que los conmutadores soportan *virtual output queuing* a nivel de conmutador; todos los enlaces son bidireccionales y tienen un ancho de banda de 1 GByte/s; existe un planificador *round-robin* por puerto de salida; el tamaño de paquete es de 2 KBytes; todos los *buffers* tienen una capacidad para almacenar 64 paquetes, entre otros.

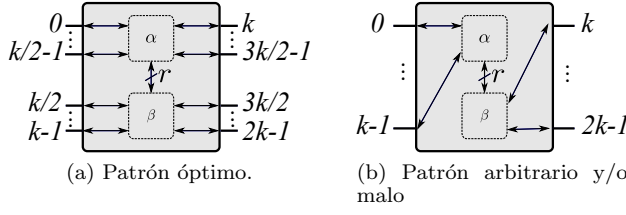


Fig. 2: Patrones de conexión a nivel de conmutador.

B. Casos de Estudio

A fin de evaluar la viabilidad y potencial de los conmutadores combinados, se ha estimado el rendimiento de la red descrita en la sección IV-A, considerando T -switches con el SCP que se ha demostrado siguiendo la metodología propuesta en la sección III-B que es el óptimo² para los T -switches bajo las condiciones de simulación (esta red se etiqueta en las figuras como O-BMIN). Tras ello, el rendimiento de esta red se compara con el rendimiento alcanzado por una red equivalente que emplea conmutadores en un chip del mismo grado que los T -switches. Dicha red será referenciada como U-BMIN y representa una cota superior de rendimiento.

Además, para poner de manifiesto la importancia del SCP también se ha considerado otro caso de estudio que representa a un patrón arbitrario y/o malo, referenciado como B-BMIN en la figuras. Las redes de los casos O-BMIN y B-BMIN usan T -switches con los patrones de conexión a nivel de conmutador OSCP y BSCP, respectivamente. Ambos patrones pueden verse en la figura 2 para un T -switch arbitrario de $k \times k$ puertos y r puertos internos entre los conmutadores internos. También se incluye un cuarto caso de estudio: una red (L-BMIN) que conecta la misma cantidad de terminales, pero con conmutadores en chip de menor grado que también son viables con la escala de integración actual.

Otro propósito de esta evaluación es saber si la diferencia de rendimiento entre los casos de estudio es significativa, y en caso afirmativo, medirla.

Obsérvese que aunque en todos los casos la red interconecta 4096 terminales, los casos U-BMIN, O-BMIN y B-BMIN, son redes 16-ary 3-tree formadas por un total de 768 conmutadores de 32 puertos. Sin embargo, la red del caso L-BMIN es una 8-ary 4-tree con 2048 conmutadores de 16 puertos cada uno.

C. Resultados de Simulación

Se han realizado dos tests distintos. El primero sirve para medir la diferencia de rendimiento entre los distintos casos de estudio mencionados anteriormente. Recuérdese que el objetivo global de esta evaluación es identificar problemas potenciales que limitan la eficiencia de los T -switches (y en general de los C -switches).

La figura 3 muestra los valores medios de la productividad y latencia con los intervalos de confianza al 95 % de 30 simulaciones para cada nivel de carga. Se pueden observar algunos detalles:

²Debido a la falta de espacio, aquí no se han incluido las correspondientes demostraciones, que está disponibles en [13].

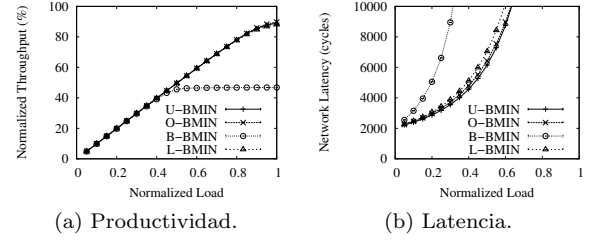


Fig. 3: Resultados del primer test para todos los casos de estudio.

- Cuando se ha considerado el SCP óptimo, la red basada en T -switches (O-BMIN) es capaz de ofrecer un rendimiento similar al que ofrece la red (U-BMIN) cuyos conmutadores *high-radix* se integran en un único chip. Por tanto, los conmutadores combinados se confirman como una buena alternativa para la construcción de conmutadores *high-radix*.
- Sin embargo, la red B-BMIN alcanza el punto de saturación con sólo el 40 % de la carga. El rendimiento dispar obtenido es debido al hecho de que los T -switches configurados con el patrón BSCP producen que la longitud del camino de la mayoría de los paquetes se incremente en un salto extra. De tal modo, también se confirma que es esencial la determinación en cada caso del SCP óptimo para cada tipo de conmutador combinado.
- Cuando se comparan los casos U-BMIN y L-BMIN, se ha observado que la latencia en el caso L-BMIN es superior a la latencia en el caso U-BMIN (por ejemplo, un 14 % al 50 % de carga). Este incremento es causado porque la red L-BMIN tiene una o más etapas que la red U-BMIN, por lo que los caminos son más largos.
- Además, se puede ver que el número de puertos destinados para la interconexión entre los conmutadores internos α y β no producen ningún tipo de degradación en el rendimiento de la red.

En el primer test se han empleado 8 puertos de cada conmutador interno para la interconexión entre ellos. Sin embargo, se necesita asignar un número de puertos suficiente para disponer de ancho de banda entre los conmutadores internos, pero al mismo tiempo evitar sobredimensionar esta cantidad. De no lograrse, es probable que se produzca una degradación de prestaciones en la comunicación entre el interior y el exterior del T -switch. Para encontrar el mismo óptimo de puertos, se ha realizado un segundo test que estima cómo dicha cantidad de puertos influye en el rendimiento global de la red.

En la figura 4 se pueden ver los resultados de productividad de la red cuando la carga de la red está al 80 % para todos los casos de estudio excepto para L-BMIN. A fin de realizar el segundo test, se ha configurado el simulador para que varíe el número de puertos entre los conmutadores internos entre 1 y 16, mientras que el número de puertos al exterior se mantiene constante.

De modo similar al primer test, se pueden resaltar

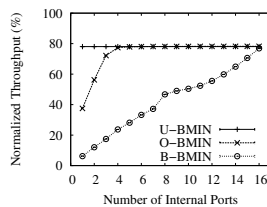


Fig. 4: Productividad de la red en función del número de puertos entre los conmutadores internos.

algunos detalles importantes en este segundo test:

- Se puede determinar el número mínimo de puertos entre los conmutadores internos para evitar la creación de un cuello de botella entre dichos conmutadores. Se puede observar que el SCP es el factor clave que influye en el rendimiento global y en cambio, el número concreto de puertos no es tan relevante.
- Cuando el número de puertos entre los conmutadores internos pasa de 4, O-BMIN alcanza el mismo rendimiento que U-BMIN. Sin embargo, cuando se considera B-BMIN, el cuello de botella no desaparece hasta que el número de puertos entre los conmutadores internos es igual a la aridad del T -switch ($k = 16$).
- Es importante remarcar que en los resultados mostrados en la figura 3 el número de puertos entre los conmutadores internos es 8, pero cuando el T -switch se configura con el SCP óptimo sólo son necesarios 4 puertos para alcanzar las prestaciones máximas. En tal caso, estos 4 puertos extra podrían emplearse en la comunicación con el exterior en lugar de hacerlo para la comunicación entre conmutadores internos, y por tanto, el grado del T -switch se incrementará de un modo más efectivo.

V. CONCLUSIONES

En este artículo se ha descrito una alternativa para la construcción de conmutadores *high-radix* consistente en combinar varios conmutadores *low-radix*. Aunque aparentemente es simple, esta estrategia plantea unos retos de diseño claves a fin de que estos conmutadores *high-radix* alcancen su mejor rendimiento. En este sentido, se han discutido aspectos importantes relacionados con la estructura interna de esta clase de conmutadores.

Los puertos ofrecidos por estos conmutadores se obtienen a partir de los puertos de sus conmutadores internos con los que se construye el conmutador. Las conexiones internas entre ambos grupos de puertos determina el patrón de conexión que se convierte en crucial para el comportamiento del conmutador *high-radix*. Por otra parte, también es esencial poder determinar la interconexión más apropiada para evitar la aparición de un cuello de botella entre los conmutadores internos.

Para comprobar el potencial de esta alternativa se han evaluado varias redes bajo ciertas condiciones para medir su rendimiento. Son redes construidas con conmutadores de un único chip y también se han considerado redes con T -switches. Los resultados

de las redes con T -switches y el patrón de conexión óptimo, se han visto que son idénticos que las redes con conmutadores construidos con conmutadores en un único chip. Además, los resultados han puesto de manifiesto la importancia de seleccionar un número adecuado de puertos para la interconexión entre los conmutadores internos.

AGRADECIMIENTOS

Este trabajo ha sido cofinanciado por el MEC y MICINN de España, fondos FEDER de la Comisión Europea, con subvenciones “Consolider Ingenio-2010 CSD2006-00046” y “TIN2009-14475-C04”; y la Junta de Comunidades de Castilla-La Mancha con proyectos “PEII 11-0229-2343” y “POII 10-0289-3724”.

REFERENCIAS

- [1] J.J. Dongarra, “TOP500 Supercomputer Sites,” 10/2010.
- [2] H. Wang, L.-S. Peh, and S. Malik, “Power-driven design of router microarchitectures in on-chip networks,” in *Proc. of the 36th annual IEEE/ACM International Symposium on Microarchitecture*, Washington, DC, USA, 2003.
- [3] M. Gusat, F. Abel, F. Gramsamer, et al., “Stability degree of switches with finite buffers and non-negligible round-trip time,” *International Conference on Computer, Communication and Networking*, vol. 27, no. 5–6, 2003.
- [4] C. Minkenberg, F. Abel, P. Muller, et al., “Control path implementation for a low-latency optical HPC switch,” in *Proc. of the 13th Symposium on High Performance Interconnects*, Washington, DC, USA, 2005, HOTI’05, pp. 29–35, IEEE Computer Society.
- [5] C. Minkenberg and M. Gusat, “Speculative flow control for high-radix datacenter interconnect routers,” *Parallel and Distributed Processing Symposium, International*, vol. 0, pp. 1–10, 2007.
- [6] “International Technology Roadmap for Semiconductors. 2010,” www.itrs.net/Links/2010ITRS/Home2010.htm.
- [7] W.J. Dally, “Virtual-channel flow control,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 3, no. 2, pp. 194–205, mar 1992.
- [8] J. Kim, W.J. Dally, B. Towles, and A.K. Gupta, “Micro-architecture of a high-radix router,” *SIGARCH Comput. Archit. News*, vol. 33, no. 2, pp. 420–431, 2005.
- [9] S. Scott, D. Abts, J. Kim, and W.J. Dally, “The BlackWidow high-radix Clos network,” *SIGARCH Comput. Archit. News*, vol. 34, no. 2, pp. 16–28, 2006.
- [10] G. Mora, J. Flich, J. Duato, et al., “Towards an efficient switch architecture for high-radix switches,” in *Proc. of the 2006 ACM/IEEE symposium on Architecture for networking and communications systems*, New York, NY, USA, 2006, pp. 11–20, ACM.
- [11] “Sun datacenter Infiniband switch 36, Sun datacenter Infiniband switch 72, Sun datacenter Infiniband switch 648: Architecture and deployment,” April 2010.
- [12] J. Kim, W.J. Dally, S. Scott, and D. Abts, “Technology-driven, highly-scalable Dragonfly topology,” in *ISCA ’08: Proc. of the 35th Annual International Symposium on Computer Architecture*, Washington, DC, USA, 2008, pp. 77–88, IEEE Computer Society.
- [13] J.A. Villar, F.J. Andújar, F.J. Alfaro, J.L. Sánchez, and J. Duato, “An alternative for building high-radix switches: Formalization and configuration methodology,” Tech. Rep. DIAB-11-02-1, Dpt. of Computing Systems. University of Castilla-La Mancha, 2011, www.dsi.uclm.es/descargas/technicalreports/DIAB-11-02-1/diab-11-02-1.pdf.
- [14] C. Gómez, F. Gilabert, M.E. Gómez, P. Lopez, and J. Duato, “Deterministic versus adaptive routing in fat-trees,” Los Alamitos, CA, USA, 2007, IEEE Computer Society.
- [15] Mellanox Technologies Inc., “Infiniscale III 3rd generation infiniband switch architecture,” www.mellanox.com/related-docs/prod_silicon/PB_InfiniScale_III.pdf.