Análisis del Riesgo de Interbloqueo durante la Exploración de Redes Myrinet

P.J. García, F.J. Quiles, F.J. Alfaro, J.L. Sánchez y J. Duato

Resumen— **Las redes de interconexión de altas prestaciones suelen implementar algún tipo de mecanismo de autoconfiguración, de manera que la red es capaz por sí misma de crear y actualizar las tablas de encaminamiento de sus nodos. En aquellas redes que pueden adoptar topologías irregulares, dicha generación de tablas de encaminamiento requiere un conocimiento previo y explícito de la topología. En consecuencia, la mayoría de redes actuales incorporan mecanismos de autodetección, que permiten obtener información topológica detallada sin intervención externa. Uno de estos mecanismos, implementado en la red Myrinet, se basa en la exploración de toda la red de forma centralizada desde un sólo terminal mediante "mensajes exploradores".**

En el presente trabajo se analiza el riesgo de que se produzca interbloqueo (*deadlock*) debido al uso de tal mecanismo, evaluando tanto la probabilidad de interbloqueo durante la exploración como su impacto en las prestaciones de la red. Hemos encontrado que el interbloqueo puede darse durante exploraciones con una probabilidad que aumenta al hacerlo la carga de tráfico en la red y el tamaño de la misma. Además, si se usan técnicas de recuperación para eliminar los interbloqueos, como sucede en Myrinet, puede provocarse un dramático incremento en la latencia media de los mensajes e incluso suceder que el mecanismo de exploración no detecte correctamente la topología.

Palabras clave— **Redes de interconexión, Myrinet, Autodetección, Interbloqueo, Recuperación.**

I. Introducción

UNA de las principales características de la mayoría de las actuales redes de altas prestaciones es su capacidad para autoconfigurarse mediante la implementación de mecanismos adecuados para tal fin. Dichos mecanismos permiten a la red crear las tablas de encaminamiento de sus nodos de forma automática y, en caso de cambios en la red, actualizar dichas tablas para adaptarse a la nueva situación. Puesto que estas redes suelen ser redes conmutadas basadas en enlaces punto a punto (a diferencia de las redes locales tradicionales basadas en medio compartido), la generación de rutas que garanticen el correcto tránsito de los mensajes entre origen y destino requiere el uso de algún algoritmo de encaminamiento [1], [2], [3], [4]. Ahora bien, la aplicación de cualquiera de estos algoritmos a una red conmutada de topología irregular requiere, a diferencia del caso de redes conmutadas regulares, un conocimiento previo y exacto de la topología concreta adoptada por la red. Por tanto, la primera fase de cualquier mecanismo de autoconfiguración debe consistir en detectar de algún modo la topología actual. Es decir, aquellas redes que se desea sean autoconfigurables [5], [6], [7] deben incorporar mecanismos de "autodetección" de la topología.

En concreto, la conocida red Myrinet [6], [8] implementa un mecanismo de autodetección que, con mínimas modificaciones, sería válido para cualquier red de las mismas características: encaminamiento fuente, conmutación *wormhole*[9], conmutadores sin ID propio ni capacidad de procesamiento, etc.. Se trata de un mecanismo de exploración centralizada, en el que un único terminal envía mensajes especiales (mensajes de exploración) siguiendo cierta pauta y, basándose en la recepción o no de respuesta a dichos mensajes, acaba construyendo un "mapa" de la red donde se refleja la topología adoptada por la misma. El análisis de dicho mecanismo nos permite afirmar que su uso puede originar situaciones de interbloqueo, ya que los mensajes de exploración no están sujetos a las restricciones impuestas al tráfico de usuario [9], [10] por el algoritmo de encaminamiento libre de bloqueos usado en la red (*Up*/Down**)[2].

En el presente trabajo se explica cómo pueden llegarse a dar situaciones de interbloqueo cuando en una red con las características de Myrinet se ejecuta el mencionado mecanismo de exploración. Además, presentamos resultados de simulación que muestran, para distintas cargas de tráfico y tamaños de red, la probabilidad de interbloqueo durante la exploración. Con la intención de valorar el impacto de los interbloqueos en el funcionamiento global de la red, y asumiendo que ésta usa técnicas de recuperación para eliminar los interbloqueos, presentamos también resultados que muestran la latencia media de los mensajes durante la exploración y la probabilidad de que ésta no detecte la topología correctamente. El estudio descrito brevemente en la presente sección se detalla en las siguientes con arreglo al siguiente esquema: en la Sección 2 se explican los fundamentos del mecanismo de exploración centralizada basado en mensajes; en la Sección 3 se razona por qué las exploraciones pueden provocar situaciones de interbloqueo y se analizan sus posibles consecuencias; la metodología y resultados de las distintas simulaciones se incluyen en la Sección 4; finalmente, presentamos nuestras conclusiones en la Sección 5.

II. Exploración centralizada basada en mensajes

El mecanismo de exploración implementado en Myrinet se realiza desde un único terminal de la red,

Departamento de Informática, Escuela Politécnica Superior, Universidad de Castilla-La Mancha, 02071 - Albacete (España). {pgarcia, paco, falfaro, jsanchez}@info-ab.uclm.es

Departamento de Informática de Sistemas y Computadores, Universidad Politécnica de Valencia, 46071 - Valencia (España). jduato@gap.upv.es

Este trabajo ha sido parcialmente financiado por la CICYT con el proyecto TIC2000-1151-C07 y por la Junta de Comunidades de Castilla-La Mancha con el proyecto PBC-02-008.

denominado "*mapper*" ya que su tarea consiste en elaborar un "mapa" de la misma. Para adaptarse a posibles cambios en la red, las exploraciones deben repetirse periódicamente. De este modo los cambios pueden detectarse comparando el "viejo" mapa con el recién elaborado. Es por ello que en cada exploración el nuevo mapa se construye desde cero, sin tener en cuenta los mapas anteriores.

Básicamente, la elaboración de un mapa consiste en detectar qué componente (si es que hay alguno) se encuentra conectado a cada uno de los puertos de la red. Para llevar a cabo esta detección, el *mapper* envía mensajes de exploración por cada puerto y procesa las posibles réplicas a los mismos.

Para explorar un puerto, el *mapper* envía a dicho puerto, en primer lugar, un mensaje de detección de terminal. Si el componente "desconocido" es realmente un terminal, éste enviará una réplica al *mapper*. Los mensajes de detección de terminal incluyen la ruta que debe seguir la réplica para llegar al *mapper*, por lo que el terminal detectado no necesita consultar su tabla de encaminamiento. En la réplica, el terminal detectado incluye su identificador único. Al recibir una réplica a un mensaje de detección de terminal, el *mapper* añadirá el terminal detectado al mapa, donde el identificador único indicado permitirá distinguirlo de otros terminales. El *mapper* esperará la llegada de la réplica a cada mensaje durante cierto tiempo, pasado el cual reenviará el mensaje. Mientras no reciba réplica, el *mapper* reenviará cada mensaje hasta 2 veces. Si tras esto sigue sin recibirse réplica, el *mapper* asumirá que el componente "desconocido" no es un terminal, y enviará al puerto un mensaje de detección de conmutador.

Los mensajes de detección de conmutador son distintos a los usados para detectar terminales debido a que un conmutador, en una red con las características de Myrinet, no puede generar mensajes y por tanto no puede enviar réplicas al *mapper*. Para solucionar esto, el *mapper* asigna a los mensajes de detección de conmutador una ruta de ida y vuelta. Siguiendo esta ruta, el mensaje sale del *mapper*, llega hasta el puerto a explorar y, sólo si existe un conmutador conectado a dicho puerto, el mensaje entrará al conmutador, saldrá de él por el mismo puerto de entrada, y regresará al *mapper* recorriendo en dirección opuesta el camino de llegada. De este modo, el mismo mensaje puede ejercer de su propia réplica al seguir una ruta "circular". Si no existiera un conmutador conectado al puerto, el mensaje se descartaría. Tras la recepción de un mensaje de detección de conmutador se añadirá el conmutador detectado al mapa, donde será identificado mediante un número asignado por el *mapper*. Los mensajes de detección de conmutador enviados por el *mapper* que no "regresan" tras cierto tiempo también son reenviados hasta 2 veces. Si tras esto sigue sin detectarse un conmutador, el *mapper* considerará que el puerto explorado está desconectado, y así lo reflejará en el mapa.

En definitiva, si existe un componente conectado a un determinado puerto se detectará por la recepción en el *mapper* de algún tipo de réplica, y la ausencia de las mismas permitirá considerar al puerto desconectado. En cualquier caso, una vez determinado el estado de la conexión, se dejará de explorar el puerto.

Téngase en cuenta que cada nuevo mapa se elabora sin considerar los anteriores y, por tanto, al principio del proceso sólo existe un puerto por explorar: el propio puerto del *mapper*. Al añadir cualquier conmutador al mapa, se añaden sus puertos a una lista de puertos por explorar[1]. Los puertos de un mismo conmutador son explorados simultáneamente. Una vez explorados todos los puertos de un conmutador, se pasa a explorar los puertos de otro. La exploración acaba cuando no existen en el mapa puertos conectados a componentes desconocidos.

III. Situaciones de interbloqueo durante la exploración de la red

Una vez descritas las principales características del mecanismo de exploración, nos ocuparemos ahora de la problemática de los interbloqueos durante la ejecución de dicho mecanismo. En primer lugar, explicaremos cómo puede llegarse a situaciones de interbloqueo durante la exploración, para a continuación analizar el posible impacto de tales situaciones en el funcionamiento global de la red.

A. Interbloqueos causados por la exploración

El problema que suponen las situaciones de interbloqueo en una red ha sido ampliamente estudiado, y para solucionarlo se han propuesto distintas técnicas [1]. La solución más habitual consiste en evitar dichas situaciones usando un algoritmo de encaminamiento libre de bloqueos. Estos algoritmos garantizan la ausencia de interbloqueos durante el funcionamiento normal de la red, a costa de imponer restricciones a las rutas que puede seguir un mensaje [9], [10]. O sea, de todas las rutas posibles entre un origen y un destino, sólo algunas serán "legales".

Myrinet, en concreto, emplea el conocido algoritmo *Up*/Down**[2] para generar las rutas que seguirá el tráfico de usuario. Puesto que *Up*/Down** garantiza la ausencia de interbloqueos, éstos no pueden darse mientras en la red sólo existan mensajes convencionales. Ahora bien, durante las exploraciones, el *mapper* envía los mensajes de exploración sin las restricciones que *Up*/Down** impone a los mensajes de usuario[2]. Por tanto, no puede garantizarse que los mensajes de exploración sigan rutas legales. En definitiva, la coexistencia de mensajes que siguen rutas legales y otros que pueden seguir rutas ilegales impide que durante la exploración pueda asegurarse la ausencia de interbloqueos[3].

Por ejemplo, pueden darse interbloqueos durante la exploración de la red cuyo diagrama se muestra en

[1] Salvo el puerto por donde el mensaje de exploración entró y salió del conmutador, que ya se considera explorado.

[2] De hecho, ya que cada exploración se realiza sin considerar el mapa anterior, tampoco se considera su grafo *Up*/Down**.

[3] Asumiendo que la topología de la red contiene, al menos, un conjunto de canales donde un interbloqueo pueda suceder (o sea, canales en disposición cíclica).

la Figura 1. Un posible grafo *Up*/Down** para dicha red puede verse en la Figura 2. Teniendo en cuenta la regla principal[4] de *Up*/Down**, ningún mensaje de usuario puede cruzar el conmutador D desde el puerto P_0 hacia el puerto P_1 (y viceversa). Pero los mensajes de exploración sí que podrían hacerlo si fuera necesario para explorar la red.

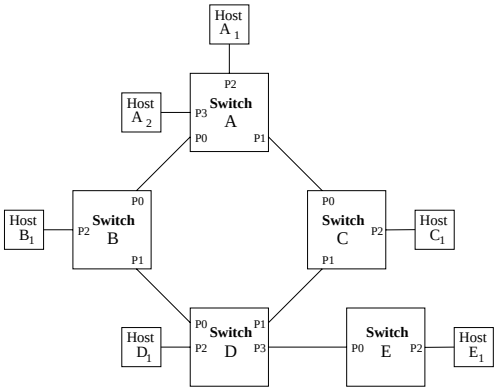

Fig. 1. Diagrama de la red a explorar.

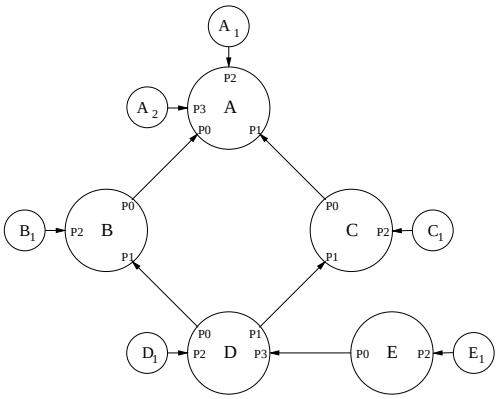

Fig. 2. Grafo *Up*/Down** para la red de la Figura 1.

La Figura 3 refleja una situación de interbloqueo que puede producirse durante la exploración de la red mostrada en la Figura 1. Se ha representado cada enlace dividido en dos canales opuestos, y cada *buffer* a la entrada de un puerto mediante un pequeño rectángulo. Los *buffers* sombreados contienen mensajes ($M_1, M_2, ...$) que siguen las rutas indicadas en la tabla adjunta. Las flechas de puntos indican el puerto de salida solicitado por cada mensaje. El terminal A_1 es el *mapper*, y M_1 es el único mensaje de exploración en la red. El resto ($M_2, M_3, ...$) son mensajes de usuario. Se asume que cada mensaje llena el *buffer* que lo contiene, de manera que el control de flujo impide la llegada de más *flits* desde el correspondiente canal de entrada.

En la Figura 3, la ruta seguida por M_1 indica que es un mensaje enviado por el *mapper* para detectar un posible terminal conectado al puerto P_1 del conmutador D. También podría ser un mensaje de detección de un conmutador conectado a dicho puerto,

[4]Ningún mensaje puede cruzar un enlace en dirección *Up* (el sentido de las flechas del grafo) tras haber cruzado un enlace en dirección *Down* (el sentido opuesto a las flechas del grafo).

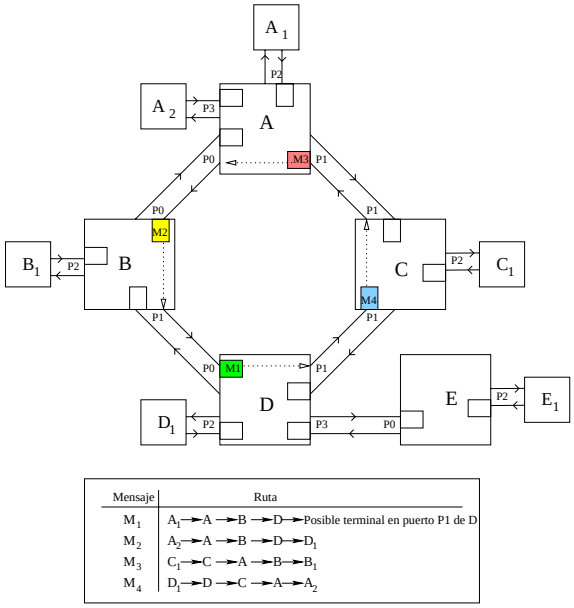

Fig. 3. Situación de interbloqueo en la red debida a un mensaje de exploración.

ya que en este caso la ruta seguida por el mensaje sería la misma hasta su salida por el puerto explorado. En cualquier caso, M_1 sigue una ruta "ilegal", lo cual acaba produciendo interbloqueo.

Téngase en cuenta que para que puedan darse situaciones de interbloqueo durante la exploración es imprescindible que exista en la red al menos un mensaje de exploración siguiendo una ruta ilegal. Ahora bien, esta condición es necesaria, pero no suficiente, para que el interbloqueo se produzca. Además, no todos los mensajes de exploración siguen rutas ilegales. En definitiva, la presencia de mensajes de exploración en la red hace posible, pero no garantiza, la existencia de situaciones de interbloqueo.

B. *Impacto de los interbloqueos en el funcionamiento de la red*

Una vez justificado el riesgo de situaciones de interbloqueo durante la exploración, cabe preguntarse si dichas situaciones pueden degradar las prestaciones de la red. Es obvio que un interbloqueo permanente tendría consecuencias catastróficas para la circulación de mensajes. Afortunadamente, Myrinet implementa un mecanismo que impide que un mensaje permanezca bloqueado indefinidamente [6]. Este mecanismo establece que si el tiempo que un mensaje permanece almacenado en un *buffer* sobrepasa cierto límite (un *timeout* fijo), se descartará la información contenida en dicho *buffer*. Por tanto, ante un interbloqueo, algunos de los mensajes implicados en el mismo serán descartados antes o después y el interbloqueo desaparecerá. Esta es la filosofía de las técnicas de recuperación de bloqueos: éstos pueden ocurrir, pero un mecanismo los detecta y elimina [1].

Sin embargo, el empleo de este mecanismo basado en *timeouts* no convierte en inocuas las situaciones de interbloqueo. Esto es debido a que dichas situaciones, que pueden seguir produciéndose, durarán

hasta que sean detectadas (o sea, hasta que los *timeouts* expiren). Durante este tiempo, parte de la red estará inaccesible al tráfico [5], lo que afectará negativamente a la latencia de los mensajes. Cabría pensar en reducir el *timeout*, pero esto complicaría el distinguir un interbloqueo de una congestión en la red.

Otro inconveniente de este mecanismo es que deben descartarse mensajes para eliminar el interbloqueo. En general, esto implicará que los mensajes descartados deban ser reenviados, lo que consumirá tiempo de procesamiento y aumentará la contención en la red. Pero incluso más grave parece la posibilidad de que los propios mensajes de exploración sean descartados. Para el *mapper*, dicho descarte significará una ausencia segura de réplica. Aunque está previsto que el *mapper* reenvíe los mensajes de exploración sin respuesta, sólo lo hará un escaso número de veces. En el peor de los casos, un mismo mensaje de exploración puede ser descartado cada vez que el *mapper* lo envíe, y esto puede provocar que un elemento presente en un puerto no sea detectado.

En definitiva, la presencia de interbloqueos durante la exploración puede degradar distintos aspectos del funcionamiento de la red, incluso aunque se use un mecanismo para eliminarlos.

IV. Evaluación de la degradación de prestaciones durante la exploración

Con intención de cuantificar la degradación de las prestaciones de la red durante la exploración, hemos evaluado los siguientes parámetros:

- Probabilidad de situaciones de interbloqueo.
- Probabilidad de fallo en la detección de la topología.
- Latencia media de los mensajes de usuario.

Para realizar esta evaluación hemos usado un simulador a nivel de *flit*, basado en eventos, que refleja el comportamiento de una red Myrinet. Antes de presentar los resultados obtenidos, detallaremos los modelos de red y tráfico empleados en las simulaciones y los criterios seguidos en la evaluación.

A. Modelos de red y tráfico

En todas las pruebas, la red se compone de conmutadores de 8 puertos y terminales interconectados mediante enlaces punto a punto. La topología es irregular, con la única restricción de que cada conmutador se conecta a otros 4 y a 4 terminales. Este modelo es bastante realista y ha sido empleado ya en otros estudios [11], [12], [4], [13]. Hemos realizado simulaciones con redes de 8, 16 y 32 conmutadores.

Los enlaces se han modelado para representar cables LAN de Myrinet de 10 metros, con un ancho de banda de 160 MB/s y un retardo de 4.92 ns/m. Un *flit* equivale a un byte, y cada canal físico puede transmitir un *flit* en paralelo. El tamaño de los *buffers* de cada puerto es de 80 bytes, garantizando el protocolo de control de flujo *Stop&Go*[6] que no se desbordarán. El *timeout* empleado para descartar mensajes bloqueados en un *buffer* es 25 ms. Todos estos parámetros corresponden a componentes Myrinet de segunda generación (Myrinet-1280).

El simulador incluye el mecanismo de exploración descrito previamente. El *mapper* espera la réplica a los mensajes de exploración durante 5 ms, y reenvía los mensajes sin réplica 2 veces como máximo.

En cuanto al tráfico, hemos empleado una distribución uniforme de destinos. En cada simulación, la tasa de generación de mensajes es constante e idéntica para todos los terminales. Para obtener resultados en función del tráfico, hemos variado dicha tasa en distintas simulaciones. El tamaño de los mensajes es variable debido a que el tamaño de la cabecera varía en función de la ruta, pero la carga es fija para los mensajes de usuario (64 bytes) y para los de exploración (60 bytes para los de detección de terminal y 12 para los de detección de conmutador).

B. Criterios de evaluación

Para cada tamaño de red y tasa de generación de tráfico hemos realizado 100 simulaciones, variando la semilla aleatoria usada para generar tráfico. En cada simulación, la exploración no comienza hasta transcurrido un intervalo que permita al tráfico alcanzar un estado estable. Las simulaciones finalizan cuando el proceso de exploración concluye.

La probabilidad de interbloqueo para cada caso se ha evaluado como el porcentaje de simulaciones en las que se produce al menos un interbloqueo durante la exploración. Un interbloqueo no provoca el fin de una simulación, ya que el mecanismo de *timeouts* descartará mensajes para desbloquear la red. Si dicho descarte provoca que un componente no sea detectado, la simulación finaliza. El número de simulaciones concluidas así ha permitido estimar, para cada caso, la probabilidad de que la topología se detecte incorrectamente. Además, en cada simulación hemos medido la latencia media de los mensajes desde su generación considerando sólo aquellos mensajes recibidos tras el inicio de la exploración. Para cada red y tasa de tráfico hemos obtenido un único valor de latencia media como la media de los resultados de las correspondientes 100 simulaciones.

C. Resultados de probabilidad de interbloqueo

La Figura 4 muestra, para distintos tamaños de red, la probabilidad de interbloqueo durante la exploración en función de la tasa de generación de tráfico.

Independientemente del tamaño de la red, la probabilidad de interbloqueo crece al aumentar la carga de tráfico, debido a que cuantos más mensajes haya circulando, más fácil será que se llenen los *buffers* en aquellas zonas de la red donde pueden darse interbloqueos (o sea, en los "ciclos" de canales).

Nótese también que la probabilidad de interbloqueo crece con el tamaño de la red, debido a que cuanto mayor es ésta, mayor es el número de mensajes de exploración necesarios para descubrir su

[5] Esta parte será como mínimo la correspondiente a los canales directamente implicados en el interbloqueo, pero en general, poco después de que se produzca éste, todos los canales de la red se bloquearán.

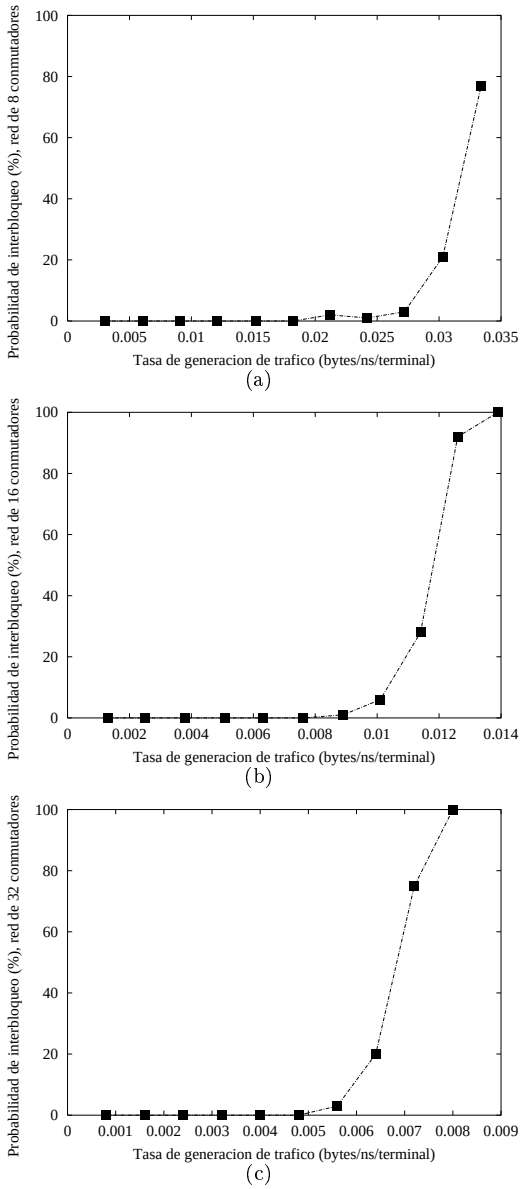

Fig. 4. Probabilidad de interbloqueo para redes de (a) 8, (b) 16, y (c) 32 conmutadores.

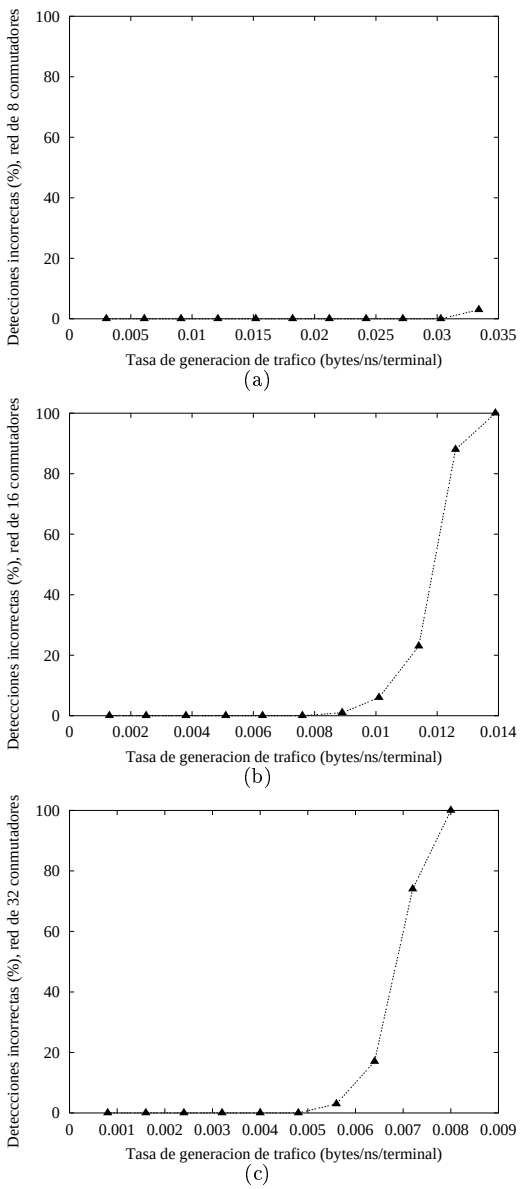

Fig. 5. Probabilidad de detección incorrecta de la topología para redes de (a) 8, (b) 16, y (c) 32 conmutadores.

topología y, por tanto, más mensajes cruzarán rutas ilegales. Además, cuanto mayor es una red, más fácil es que contenga canales en disposición cíclica.

D. Resultados de probabilidad de detección incorrecta de la topología

La Figura 5 muestra, para cada tamaño de red, y en función de la tasa de generación de tráfico, la probabilidad de que el mecanismo de exploración no acabe detectando la auténtica topología de la red.

Puede verse en las Figuras 4 y 5 que la probabilidad de detección incorrecta para un determinado tamaño de red y tasa de tráfico es siempre menor o igual que la probabilidad de interbloqueo para el mismo caso. Esto es debido a que las exploraciones que produzcan interbloqueos no siempre provocarán que la topología sea detectada incorrectamente. Este hecho es más notable en la red de 8 conmutadores (Figura 5(a)), donde la máxima probabilidad de detección incorrecta es muy baja. La razón de este resultado es el bajo número de mensajes de exploración necesarios para explorar una red pequeña, que hace poco probable que uno de ellos sea descartado 3 veces y provoque que un componente no sea detectado.

E. Resultados de latencia media

En la Figura 6 se compara la latencia media de los mensajes medida durante la exploración con la obtenida durante la operación normal (sin exploración) de la red. Ambas se han medido para los tamaños de red y tasas de tráfico usados en las medidas expuestas previamente, y siempre midiendo la latencia de los mensajes desde su generación.

En todos los casos de la Figura 6 se muestra claramente que, independientemente del tamaño de la red, la latencia media durante la exploración crece dramáticamente para tasas de tráfico mucho menores que la tasa de saturación en funcionamiento normal.

Comparando los resultados expuestos en esta sección y en las previas se aprecia la relación entre

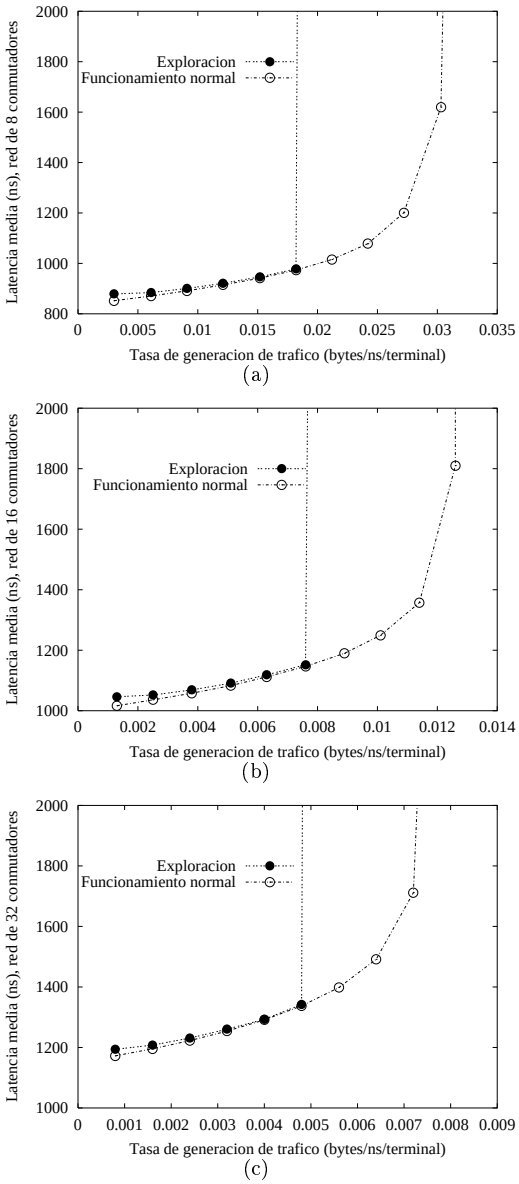

Fig. 6. Latencia mediade los mensajes durante la exploración y durante el funcionamiento normal de la red para redes de (a) 8, (b) 16, y (c) 32 conmutadores.

latencia y probabilidad de interbloqueo. Para tasas en que esta probabilidad es cero, la latencia media durante la exploración es casi idéntica a la obtenida en funcionamiento normal[6]. Si dicha probabilidad no es nula, la latencia media se dispara, debido a que los interbloqueos desaparecen sólo tras vencer un *timeout*, durante el cual gran parte de la red no estará disponible al tráfico.

V. Conclusiones

Hemos detectado y justificado que la exploración centralizada basada en mensajes puede ser causa de que existan interbloqueos cuando se emplea en una red de las características de Myrinet. También hemos analizado el impacto de tales interbloqueos en el funcionamiento normal de la red.

En este artículo se ha expuesto cómo se generan los interbloqueos durante la exploración debido a que los mensajes empleados en la misma no están sujetos a las restricciones impuestas por un algoritmo de encaminimiento libre de bloqueos. Por tanto, no pueden darse garantías de ausencia de interbloqueos durante la exploración de la red mientras en ella exista al menos un ciclo de canales.

Además, se han presentado resultados de simulación que muestran que la probabilidad de que se produzcan tales interbloqueos crece al aumentar el tamaño y la carga de tráfico de la red. También se han presentado resultados que muestran la degradación de las prestaciones de la red cuando, para solucionar los interbloqueos, se emplea un mecanismo de recuperación que descarta mensajes basándose en *timeouts*. Esta degradación se refleja en un incremento de la latencia media de los mensajes durante la exploración y en la posibilidad de que la topología no se detecte correctamente. De acuerdo con los resultados presentados, un 100% de exploraciones provocarán interbloqueo y no detectarán correctamente la topología en redes de medio o gran tamaño al acercarse a la tasa de tráfico para la cual satura la red.

[6]Esto indica que la sobrecarga de los mensajes de exploración casi no afecta a la latencia media de los mensajes.

Referencias

[1] J. Duato, S. Yalamanchili, and L. Ni, *Interconnection networks. An engineering approach*, Morgan Kaufmann Publishers, 2002.

[2] M.D. Schroeder, A.D. Birrell, M. Burrows, H. Murray, R.M. Needham, T.L. Rodeheffer, E.H. Satterthwate, and C.P. Thacker, "Autonet: A high-speed, self-configuring local area network using point-to-point links," *IEEE Journal on Selected Areas in Communications*, vol. 9, no. 8, pp. 1318–1334, Oct. 1991.

[3] L. Cherkasova, V. Kotov, and T. Rokicki, "Fibre channel fabrics: Evaluation and design," in *29th Int. Conf. on System Sciences*, Feb. 1995.

[4] J.C. Sancho, A. Robles, and J. Duato, "A new methodology to compute deadlock-free routing tables for irregular networks," in *Proceedings of the Workshop on Communication, Architecture and Applications for Network-based Parallel Computing*, Jan. 2000.

[5] T.L. Rodeheffer and M.D. Schroeder, "Automatic reconfiguration in Autonet," Tech. Rep. 77, Systems Research Center of Digital Equipment Corporation, Sept. 1991.

[6] N.J. Boden, D. Cohen, and R.E. Felderman, "Myrinet – a gigabit per second local area network," *IEEE Micro*, pp. 29–36, Feb. 1995.

[7] J. Duato, R. Casado, F.J. Quiles, and J.L. Sánchez, *Dependable Network Computing*, chapter Dynamic reconfiguration in high speed local area networks, Kluwer Academic Publishers, 1999.

[8] http://www.myri.com, "Myricom home page," Tech. Rep., Myricom, Inc., 2001.

[9] W.J. Dally and C.L. Seitz, "Deadlock-free message routing in multiprocessor interconnection networks," *IEEE Transactions on Computers*, vol. C-36, no. 5, pp. 547–553, May 1987.

[10] J. Duato, "A new theory of deadlock-free adaptive routing in wormhole networks," *IEEE Transactions on Parallel and Distributed Systems*, vol. 4, no. 12, pp. 1320–1331, Dec. 1993.

[11] F. Silla and J. Duato, "Improving the efficiency of adaptive routing in networks with irregular topology," in *Proceedings of the 1997 Int. Conference on High Performance Computing*, Dec. 1997.

[12] J. Flich, M.P. Malumbres, P. López, and J. Duato, "Improving routing performance in Myrinet networks," in *Proceedings of International Parallel & Distributed Processing Symposium (IPDPS 2000)*, May 2000.

[13] P.J. García, M.D. Mora, F.J. Alfaro, J.L. Sánchez, and J. Flich, "Evaluation of alternative arbitration policies for Myrinet switches," in *Proceedings of Workshop on Communication Architecture for Clusters*, April 2002.