View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by Elsevier - Publisher Connector

Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Computer Science 18 (2013) 2521 – 2524

Procedia Computer Science

International Conference on Computational Science, ICCS 2013

Deterministic Routing with HoL-Blocking-Awareness for Direct Topologies

R. Peñaranda[a], C. Gómez[b], M.E. Gómez[a], P. López[a], J. Duato[a]

[a]*GAP, UPV, Valencia, España*
[b]*DSI, UCLM, Albacete, España*

Abstract

Routing is a key design factor to obtain the maximum performance out of interconnection networks. Depending on the number of routing options that packets may use, routing algorithms are classified into two categories. If the packet can only use a single predetermined path, routing is deterministic, whereas if several paths are available, it is adaptive. It is well-known that adaptive routing usually outperforms deterministic routing. However, adaptive routers are more complex and introduces out-of-order delivery of packets. In this paper, we take up the challenge of developing a deterministic routing algorithm for direct topologies that can obtain a similar performance than adaptive routing, while providing the inherent advantages of deterministic routing such as in-order delivery of packets and implementation simplicity. The proposed deterministic routing algorithm is aware of the HoL-blocking effect, and it is designed to reduce it, which, as known, it is a key contributor to degrade interconnection network performance.

Keywords: Parallel computers; Interconnection networks; Direct topologies; Deterministic routing; Adaptive routing.

1. Introduction

The interconnection network performance strongly impacts on the performance of large parallel computers. Latency and throughput are the key performance metrics of interconnection networks [1, 2]. To achieve the required performance level, the designer manipulates three main parameters [1, 2]: topology, routing and switching. The switching mechanism decides how resources are allocated to the messages while they advance through the network. Topology usually adopts a regular structure that simplifies routing, implementation and expansion capability. Among the different taxonomies of regular topologies, the most commonly one divides them into direct and indirect topologies. Taking into account that many of the very large machines of the top500 list [3] adopt a direct network topology, in this paper, we will focus on direct networks, although its conclusions could be extrapolated to indirect networks. Routing is a critical design issue of interconnection networks [1]. Routing algorithms can be deterministic or adaptive. In deterministic routing schemes, an injected packet traverses a unique, fixed, predetermined path between source and destination, while in adaptive routing schemes, several paths are available. However, as several routes are possible, a choice or selection of the path that will be finally used is required, which makes routing operation more complex. In addition, several concerns about deadlock-freedom must be taking into account. Moreover, adaptive routing introduces the problem of out-of-order delivery of packets, which

*Corresponding author. email: ropeaceb@gap.upv.es

1877-0509 © 2013 The Authors. Published by Elsevier B.V. Open access under CC BY-NC-ND license.
Selection and peer review under responsibility of the organizers of the 2013 International Conference on Computational Science
doi:10.1016/j.procs.2013.05.432

occurs when a packet sent from a given source arrives to a given destination before another one sent previously from the same source to the same destination. In-order packet delivery is important for cache coherence protocols and communication libraries. While there are solutions to this problem (for instance by using a reordering buffer at destinations [4]), they are not simple, as they require the use of storage resources and control packets. Instead, deterministic routing guarantees in-order packet delivery by design. Another issue to consider when designing routing algorithms is to avoid interference among packets destined to different nodes [5, 6], since the Head-Of-Line (HoL) blocking effect may limit the throughput of the switch up to 58% of its peak value [7, 8, 9].

In this paper, we explore the behavior of both deterministic and adaptive routing on direct topologies, also proposing a new deterministic routing algorithm that takes advantage of virtual channels to reduce the HoL blocking effect.

The rest of the paper is organized as follows. Section 2 introduces some background. In Section 3, we present the new HoL-blocking-aware deterministic routing algorithm with virtual channels. Section 4 evaluates different topologies (torus and mesh) with different routing algorithms. Finally, some conclusions are drawn.

2. Direct topologies

The most important regular direct topology is the k–ary n–cube, which has k nodes in each of its n dimensions, connected in a ring fashion. In this topology, nodes are labeled by an identifier with as many components as dimensions in the topology $\langle p_{n-1}, ..., p_0 \rangle$, and the value of the component associated to each dimension ranges from 0 to $k-1$ (i.e., nodes are numbered from $\langle 0, 0, ..., 0 \rangle$ to $\langle k-1, k-1, ..., k-1 \rangle$). This topology is popularly known as torus. A mesh is a particular case where the k nodes of each dimension are connected by a linear array (i.e. without wraparound links).

Deterministic routing in meshes and tori can be implemented with the DOR (dimension-order routing) routing algorithm [1]. This algorithm routes packets by crossing dimensions in strictly increasing (or decreasing) order. Despite that DOR is deadlock-free in meshes, it is not in tori due to the wraparound links. Channel dependence graph cycles has to be broken by splitting each physical channel into two VCs [10]. Another technique used to avoid deadlocks in tori with deterministic routing is the bubble flow control mechanism [11].

Adaptive routing can be achieved in meshes and tori by allowing packets to cross dimensions in any order. According to Duato's theory [12], we add a set of VCs that may be used to cross network dimensions in any order. Deadlock freedom is achieved by providing a *escape path* to packets, by means of using a deadlock-free routing algorithm (for instance, DOR) in other virtual channel.

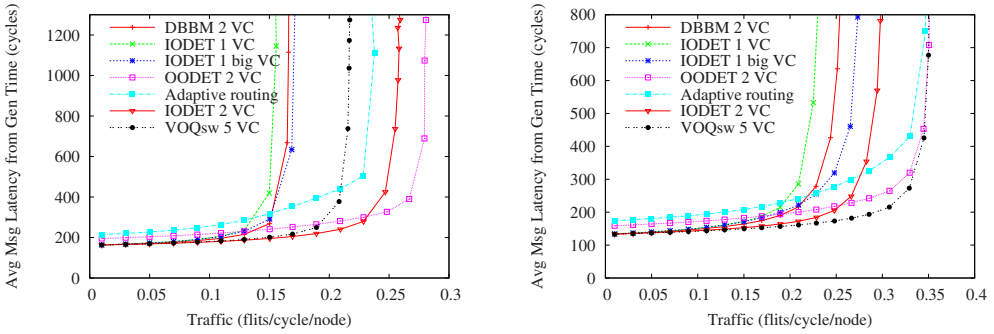
3. A HoL-blocking-aware Deterministic Routing Algorithm

Adaptive routing requires more resources (i.e. VCs) than deterministic routing. Indeed, more VCs imply not only more buffers but also increasing the crossbar size and the routing algorithm complexity. We will analyze alternatives to improve network performance also based on the use on VCs but trying to reduce this complexity.

A first approach is to use deterministic routing with several VCs, which offer as many routing options as VCs, also requiring a selection function as adaptive routing does. Although this routing algorithm improves network performance over the baseline deterministic routing (see Section 4), switch complexity and therefore routing time are still increased. Moreover, deterministic routing with several VCs may also introduce out-of-order delivery of packets. For this reason, we will refer to this mechanism as OODET (Out-of-Order DETerministic routing).

What is actually needed is a method to assign packets to VCs. We propose to classify them depending on their destinations. If a packet destined to a given node is only assigned to one VC, always the same, the result is a deterministic routing algorithm, with all its advantages (simpler, faster and in-order delivery of packets). In particular, VCs are assigned to packets according to the component of its destination node in the dimension in which the packet is being routed, modulo the number of VCs. As a consequence, the proposed mechanism classifies packets among the VCs, thus contributing to reduce the interference among packets, and, therefore, to reduce the HoL-blocking effect. A nice property of this mechanism is, that, as only one routing option is provided for each destination, it preserves, by design, in-order delivery of packets. On the other hand, VC selection is easily done, as only the LSBs of a component from destination identifier are required. In contrast to OODET, we will refer to this mechanism as IODET (In-Order DETerministic routing).

In [14], a similar mechanism (DBBM) was proposed, but considering the whole destination identifier modulo the number of VCs to assign packets to VCs. The fact of only considering the LSBs of the destination provides



(a) 8×8 Mesh (b) 8×8 Torus

Fig. 1. Average packet latency vs. accepted traffic for uniform traffic.

Table 1. Comparison of the number of switching elements for the torus topology

Dimensions	Virtual channels	OODET	Adaptive	IODET	Dimensions	Virtual channels	OODET	Adaptive	IODET
2	2	48	64	40	2	6	336	480	216
3	2	96	144	84	3	6	720	1152	540
4	2	160	256	144	4	6	1248	2112	1008
6	2	336	576	312	6	6	2736	4896	2376
2	4	160	224	112	2	8	576	832	352
3	4	336	528	264	3	8	1248	2016	912
4	4	576	960	480	4	8	2176	3712	1728
6	4	1248	2208	1104	6	8	4800	8640	4128

lower opportunities of classifying packets among VCs (see Section 4). DBBM is a simplification of VOQnet [15], which needs as many VCs as nodes in the network, and associates each VC to a different destination. Another scheme is VOQsw [16], in which VCs are selected according to the next output port the packet will use. VOQsw and VOQnet are not scalable as the number of VCs they require depends on the system or switch size, respectively.

4. Experimental Evaluation

In this section, we compare by simulation adaptive versus deterministic routing in meshes and tori. Each node has a switch based on a full crossbar with queues of two packets both at their input and output ports. Switch and link bandwidth is one flit per clock cycle, and link fly time is 1 clock cycle. We also assume that each node require 20 clock cycles to apply the routing algorithm in the baseline deterministic routing (with one VC). To consider the increased complexity of adaptive routing, we have assumed a higher routing time. In particular, we have used Chien's model ([13, 17]). This model states that routing time depends on the degree of freedom (i.e., the number of routing options) of the routing algorithm. It is the sum of a constant time (i.e. the one required to apply the routing function) plus a component that grows logarithmically with the number of routing options (i.e. the selection function delay). If this fact is taken into account, OODET and adaptive routing would take more clock cycles to route a packet than IODET, DBBM, and VOQsw.

In Figure 1.(a) we can see the behavior of 2D mesh with 64 nodes. As expected, adaptive routing outperforms the baseline deterministic routing, with one VC. But adaptive routing uses two VCs, each one with space for two packets. For this reason, we have also included in the comparison a deterministic routing with one VC but with double size (4 packets, big VC in the Figure). Although it improves the baseline deterministic routing, it does not reach the performance of adaptive routing. On the other hand, IODET and OODET with 2 VCs outperform adaptive routing. This is due to the ability of IODET to classify packets and the increased routing delay of adaptive routing. Notice that DBBM does not improve very much the performance of deterministic routing. The reason is that, as it uses the modulo of the whole packet destination identifier, packets may not classified in all dimensions. In fact, in some dimensions, all packets may use the same VC, wasting the other VCs. Finally, regarding VOQsw, in spite of using 5 VCs, it obtains a worse performance than adaptive routing.

Figure 1.(b) shows the behavior of a 2D torus with 64 nodes. In this case, adaptive routing works better than in mesh. This is because the mesh is not a true regular topology and the adaptive routing algorithm concentrates the

traffic in the center of the network, lowering performance. We can see that OODET, VOQsw and adaptive routing outperform IODET. However, IODET has a lower cost, because the number of switching elements required is strongly reduced if the restrictions introduced by the routing algorithm are considered in the design of the switch. This can be seen in Table 1. Notice that VOQsw needs the same number of switching elements as OODET if the same number of VCs are used, because in a given dimension a packet in VOQsw can change the VC at each hop, like OODET. As it can be seen, IODET requires the lowest number of switching elements in all analyzed cases.

5. Conclusions

This paper proposes a new HoL-blocking-aware deterministic routing algorithm (IODET) for direct regular topologies. It uses virtual channel flow-control, and assigns packets to VCs according to a subset of bits of the destination identifier (i.e., the component that corresponds to the dimension the packet is traversing). The result is a deterministic routing algorithm which exhibits its well-known advantages over adaptive routing: simplicity, low routing times and in-order delivery of packets. If routing times are scaled according to the number of routing options of each routing algorithm, IODET is able to outperform adaptive routing in meshes, while it reaches a performance half-way between the baseline deterministic and adaptive routing algorithms in torus. In addition, IODET also simplifies switch design. This combination of a moderate improvement in performance with a simple implementation makes IODET an interesting alternative to consider when selecting the routing algorithm.

Acknowledgment

This work was supported by the Spanish Ministerio de Economía y Competitividad (MINECO) and Plan E funds, under Grants TIN2009-14475-C04 and TIN2012-38341-C04, and by Ayudas para Primeros Proyectos de Investigación from Universitat Politècnica de València under grant ref. 2370.

References

[1] J. Duato, S. Yalamanchili, N. Lionel, Interconnection Networks: An Engineering Approach, Morgan Kaufmann Publishers Inc., USA, 2002.
[2] W. Dally, B. Towles, Principles and practices of interconnection networks, Morgan Kaufmann, 2004.
[3] TOP500 supercomputer site, http://www.top500.org.
[4] J. C. Martínez, J. Flich, A. Robles, P. López, J. Duato, In-order packet delivery in interconnection networks using adaptive routing, in: IEEE International Parallel and Distributed Processing Symp, 2005.
[5] C. Gómez, F. Gilabert, M. Gómez, P. López, J. Duato, Deterministic versus adaptive routing in fat-trees, in: Parallel and Distributed Processing Symposium, 2007. IPDPS 2007. IEEE International, 2007, pp. 1 –8. doi:10.1109/IPDPS.2007.370482.
[6] X.-Y. Lin, Y.-C. Chung, T.-Y. Huang, A multiple lid routing scheme for fat-tree-based infiniband networks, in: IPDPS, 2004.
[7] M. Karol, M. Hluchyj, S. Morgan, Input versus output queueing on a space-division packet switch, Communications, IEEE Transactions on 35 (12) (1987) 1347 – 1356. doi:10.1109/TCOM.1987.1096719.
[8] J. Bennett, C. Partridge, N. Shectman, Packet reordering is not pathological network behavior, Networking, IEEE/ACM Transactions on 7 (6) (1999) 789 –798. doi:10.1109/90.811445.
[9] N. McKeown, V. Anantharam, J. Walrand, Achieving 100input-queued switch, in: INFOCOM '96. Fifteenth Annual Joint Conference of the IEEE Computer Societies. Networking the Next Generation. Proceedings IEEE, Vol. 1, 1996, pp. 296 –302 vol.1. doi:10.1109/INFCOM.1996.497906.
[10] W. Dally, C. Seitz, Deadlock-free message routing in multiprocessor interconnection networks, Computers, IEEE Transactions on C-36 (5) (1987) 547 –553. doi:10.1109/TC.1987.1676939.
[11] V. Puente, R. Beivide, J. Gregorio, J. Prellezo, J. Duato, C. Izu, Adaptive bubble router: a design to improve performance in torus networks, in: Parallel Processing, 1999. Proceedings. 1999 International Conference on, 1999, pp. 58 –67. doi:10.1109/ICPP.1999.797388.
[12] J. Duato, A necessary and sufficient condition for deadlock-free routing in cut-through and store-and-forward networks, IEEE Transactions on Parallel and Distributed Systems 7 (1996) 841–854. doi:http://doi.ieeecomputersociety.org/10.1109/71.532115.
[13] A. Chein, A cost and speed model for k-ary n-cube wormhole routers, Parallel and Distributed Systems, IEEE Transactions on 9 (2) (1998) 150 –162. doi:10.1109/71.663877.
[14] T. Nachiondo, J. Flich, J. Duato, Buffer management strategies to reduce hol blocking, IEEE Transactions on Parallel and Distributed Systems 21 (2010) 739–753. doi:http://doi.ieeecomputersociety.org/10.1109/TPDS.2009.63.
[15] L. D. W.J. Dally, P. Carvey, Architecture of the avici terabit switch/router, in:Proceedings of Hot Interconnects 6.
[16] T. E. Anderson, S. S. Owicki, J. B. Saxe, C. P. Thacker, High-speed switch scheduling for local-area networks, ACM Trans. Comput. Syst. 11 (4) (1993) 319–352. doi:10.1145/161541.161736.
URL http://doi.acm.org/10.1145/161541.161736
[17] J. Duato, P. López, Performance evaluation of adaptive routing algorithms for k-ary n-cubes, in: K. Bolding, L. Snyder (Eds.), Parallel Computer Routing and Communication, Vol. 853 of Lecture Notes in Computer Science, Springer Berlin, Heidelberg, 1994, pp. 45–59, 10.1007/3-540-58429-3_27.