On the Optimal Network for Multicomputers: Torus or Hypercube?

Mohamed Ould-Khaoua

Department of Computer Science, University of Strathclyde, Glasgow G1 1XH, UK.

Abstract. This paper examines the relative performance merits of the torus and hypercube when adaptive routing is used. The comparative analysis takes into account channel bandwidth constraints imposed by VLSI and multiple-chip technology. This study concludes that it is the hypercube which exhibits the superior performance, and therefore is a better candidate as a high-performance network for future multicomputers with adaptive routing.

1 Introduction

The hypercube and torus are the most common instances of k-ary n-cubes [5]. The former has been used in early multicomputers [3, 10] while the latter has become popular in recent systems [4, 8]. This move towards the torus has been mainly influenced by Dally's study [5]. When systems are laid out on a VLSI-chip, Dally has shown that under the constant wiring density constraint, the 2 and 3-D torus outperform the hypercube due to their higher bandwidth channels.

Abraham [1] and Agrawal [2] have argued that the wiring density argument is applicable where a network is implemented on a VLSI-chip, but not in situations where it is partitioned over many chips. In such circumstances, they have identified that the most critical bandwidth constraint is imposed by the chip pin-out. Both authors have concluded that it is the hypercube which exhibits better performance under this new constraint.

Wormhole routing [11] has also promoted the use of high-diameter networks, like the torus, as it makes latency independent of the message distance in the absence of blocking. In wormhole routing, a message is broken into flits for transmission and flow control. The header flit governs the route, and the remaining data flits follow in a pipeline. If the header is blocked, the data flits are blocked in situ.

Most previous comparative analyses of the torus and hypercube [1, 2, 5, 6] have used deterministic routing, where a message always uses the same network path between a given pair of nodes. Deterministic routing has been widely adopted in practice [3, 8, 10] because it is simple and deadlock-free. However, messages cannot use alternative paths to avoid congested channels. Fully-adaptive routing has often been suggested to overcome this limitation by enabling messages to explore all the available paths in the network. Duato [6] has recently proposed a fully-adaptive routing, which achieves deadlock-freedom with minimal hardware requirement. The Cray T3E [4] is an example of a recent machine that uses Duato's routing algorithm.

The torus continues to be a popular topology even in multicomputers which employ adaptive routing. However, before adaptive routing can be widely adopted in practical systems, it is necessary to determine which of the competing topologies are able to fully exploit its performance benefits. To this end, this paper re-assesses the relative performance merits of the torus and hypercube in the context of adaptive routing. The study compares the performance of the 2 and 3-D torus to that of the hypercube. The analysis uses Duato's fully-adaptive routing [6]. The present study uses queueing models developed in [9] to examine network performance under uniform traffic. Results presented in the next section reveal that it is the hypercube which provides the optimal performance under both the constant wiring density and pin-out constraints, and thus is the best candidate as a high-performance network for future multicomputers with fully-adaptive routing.

2 Performance Comparison

The torus has higher bandwidth channels than its hypercube counterpart under the constant wiring density and pin-out constraints. The detailed derivation of the exact relationship between the channel width of the torus in terms of that of the hypercube under both constraints can be found in [1, 2, 5].

The router's switch in the hypercube is larger than that in the torus due to its larger number of physical and virtual channels. As a consequence, the switching delay in the hypercube should be higher due to the additional complexity. Comparable switching delays in the two networks can be obtained if the routers have comparable switch sizes. This can be achieved by normalising the total number of channels in the routers of the two networks.

When mapped in the 2D plane, the hypercube ends up with longer wires, and thus higher wire delays than the torus. However, delays due to long wires can be reduced by using pipelined channels as suggested in [12]. The performance of the networks is examined below when both the wire delay is taken into account and when it is ignored. For illustration, network sizes of N=64 and 1024 nodes are examined. The channel width in the hypercube is one bit. The channel width in the torus is normalised to that of the hypercube. The message length (M) is 128 flits. A physical channel in the hypercube has V=2 virtual channels. The total number of virtual channels per router in the torus is normalised to that in the hypercube.

Figures 1-a and b depict latency results in the torus and hypercube under the constant wiring density constraint and when the effects of wire delays are taken into account in the 64 and 1024 node systems respectively. The figures reveal that the torus is able exploit its wider channels to provide a lower latency than the hypercube under light to moderate traffic. However, as traffic increases its performance degrades as message blocking rises, offsetting any advantage of

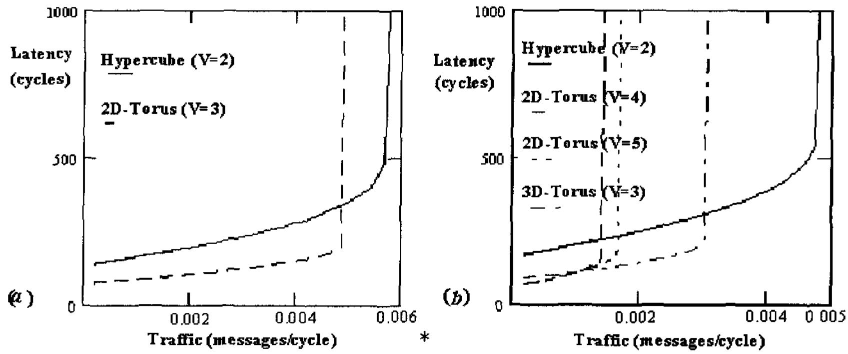

Fig. 1. The performance of the torus and hypercube including the effects of wiring delays. *(a) N=64, (b)N=1024.*

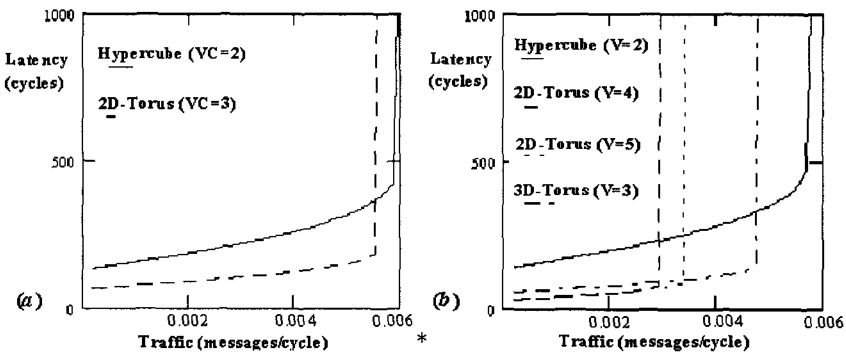

Fig. 2. The performance of the torus and hypercube ignoring the effects of wiring delays. *(a) N=64, (b)N=1024.*

having wider channels. Figures 2-a and b show latency when the effects of wire delays are ignored. The torus outperforms the hypercube under light to moderate traffic, but loses edge to the hypercube under heavy traffic. The difference in performance between the two networks increases in favour of the hypercube for larger network sizes. Figures 1 and 2 together reveal an important finding about the torus, and that is even though wires are longer in the hypercube, the torus is more sensitive to the effects of wire delays. This is because a message in the torus crosses, on average, a larger number of routers, and therefore require a longer service time to reach its destination. Since the ratio in channel width in the hypercube and torus decreases under the constant pin-out constraint, we can conclude that the hypercube is even more favourable when the networks are subjected to this condition.

3 Conclusion

This paper has compared the performance merits of the torus and hypercube in the context of adaptive routing. The results have revealed that the hypercube has superior performance characteristics to the torus, and therefore is a better candidate as a high-performance network for future multicomputers, that use adaptive routing.

References

1. S. Abraham, Issues in the architecture of direct interconnection networks schemes for multiprocessors, Ph.D. thesis, Univ. of Illinois at Urbana-Champaign(1991).
2. A. Agarwal., Limits on interconnection network performance, IEEE Trans. Parallel & Distributed Systems,Vol. 2(4), (1991) 398–412.
3. R. Arlanskas, iPSC/2 system: A second generation hypercube, Proc. 3rd ACM Conf. Hypercube Concurrent Computers and Applications,Vol. 1 (1988).
4. Cray Research Inc., The Cray T3E scalable parallel processing system, on Cray's web page at http://www.cray.com/PUBLIC/product-info/T3E/.
5. W.J. Dally, Performance analysis of k-ary n-cubes interconnection networks, IEEE Trans. Computers,Vol. 39(6) (1990) 775-785.
6. J. Duato, A New theory of deadlock-free adaptive routing in wormhole routing networks, IEEE Trans. Parallel & Distributed Systems, Vol. 4 (12) (1993) 320–331.
7. J. Duato, M.P. Malumbres, Optimal topology for distributed shared-memory multiprocessors: hypercubes again?, Proc. EuroPar'96, Lyon, France, (1996) 205–212.
8. R.E. Kessler, J.L. Schwarzmeier, CRAY T3D: A new dimension for Cray Research, in CompCon, Spring (1993) 176–182.
9. M. Ould-Khaoua, An Analytical Model of Duato's Fully-Adaptive Routing Algorithm in k-Ary n-Cubes, to appear in the Proc. 27th Int. Conf. Parallel Processing, 1998.
10. C.L. Seitz, The Cosmic Cube, CACM, Vol. 28 (1985) 22–33.
11. C.L. Seitz, The hypercube communication chip, Dep. Comp. Sci., CalTech, Display File 5182:DF:85 (1985).
12. S.L. Scott, J.R. Goodman, The impact of pipelined channel on k-ary n-cube net works, IEEE Trans. Parallel & Distributed Systems, Vol. 5(1) (1994) 2-16.