A Comparison of Router Architectures for Virtual Cut-Through and Wormhole Switching in a NOW Environment [*]

J. Duato, A. Robles, F. Silla
Universidad Politécnica de Valencia
Camino de Vera, 14
46071 - Valencia, SPAIN
e-mail: jduato@gap.upv.es

R. Beivide
Universidad de Cantabria
Avda. Los Castros s/n
Santander, SPAIN
e-mail: mon@atc.unican.es

Abstract

Most commercial routers designed for networks of workstations (NOWs) implement wormhole switching. However, wormhole switching is not well suited for NOWs. The long wires required in this environment lead to large buffers to prevent buffer overflow during flow control signaling. Moreover, wire length is limited by buffer size.

Virtual cut-through (VCT) achieves a higher throughput than wormhole switching. Moreover, the traditional disadvantages of VCT switching, as buffer requirements and packetizing overhead, disappear in NOWs.

In this paper, we show that VCT routers can be simpler than wormhole ones, while still achieving the advantages of using virtual channels and adaptive routing. We also propose a fully adaptive routing algorithm for VCT switching in NOWs. Moreover, we show that VCT routers outperform wormhole routers in a NOW environment at a lower cost.

1. Introduction

Wormhole switching was introduced as a suitable switching technique for the design of compact and fast routers. Wormhole switching pipelines message transmission across multiple routers and requires very small buffers, leading to single chip routers even with the technology available in 1985 [5]. When the channels requested by a message are busy, the message is blocked in place. Flow control signals exchanged between routers prevent buffer overflow. Therefore, buffers only need to provide storage for a few flow control digits, or flits.

Virtual cut-through switching was initially proposed for computer networks. It also pipelines message transmission across multiple routers, behaving in the same way as wormhole switching in the absence of contention. However, message flow control is performed at the level of packets, therefore requiring buffers with capacity for one or more packets to store blocked packets. Thus, when a packet blocks, it is removed from the network and stored in a single buffer. Although removing blocked packets from the network reduces contention considerably, the large storage capacity required for virtual cut-through routers led manufacturers to prefer wormhole switching. Most current routers implement this switching technique [3, 20] or variants of it [25].

As technology improved, more transistors became available and some researchers began to consider virtual cut-through as a suitable switching technique for single-chip routers [13]. The most noticeable implementation is the Chaos router [11]. This single-chip router implemented fully adaptive non-minimal routing for 2-D tori, using input, output, and central buffers with capacity for 32-flit packets.

Several researchers compared the performance of virtual cut-through and wormhole switching, assuming the simplest configuration for each router: deterministic routing and no virtual channels [1, 17]. Assuming that clock frequency is the same for both routers, the results show that both switching techniques achieve the same latency for low loads. However, wormhole switching quickly saturates as load increases due to blocked packets remaining in the network and keeping the resources previously reserved. Virtual cut-through achieves a much higher throughput at the cost of increased buffer capacity [17].

The situation changes when virtual channels are used. In a wormhole router, virtual channels allow messages to pass blocked messages, leading to better utilization of link bandwidth and increased throughput [6]. Adaptive routing also reduces contention by allowing messages to follow alternative paths. This does not come without a cost. Virtual channel arbitration and multiplexing introduce additional delays. Also, switches become larger and slower [4], therefore recommending a small number of virtual channels. Adaptive routing requires a more complex arbitration scheme to select the output channel, also increasing delay [4]. Despite the additional delay and complexity, several current routers incorporate virtual channels [3, 20] and even fully adaptive routing [20], leading to network throughput comparable to

[*] This work was supported by the Spanish CICYT under Grants TIC97–0897–C04–01 and TIC98-1162-C02-01

that achieved by virtual cut-through routers [8].

Moreover, as the number of transistors per chip increased, buffer size also increased in wormhole networks, leading to flit buffers with capacity for one or more packets. For example, in the Cray T3E router, deterministic channels have capacity for one packet and adaptive channels have capacity for two packets [20]. Also, channel pipelining [19] has been proposed to reduce the drawbacks of using long wires to interconnect different routers. This technique requires buffers with capacity proportional to wire length to avoid buffer overflow during the propagation of flow control signals. Several current routers have buffer capacity in the order of 4 Kbytes [20, 25].

Networks of workstations (NOWs) are rapidly emerging as a cost-effective alternative to parallel computers. Switch-based interconnects with irregular topologies allow the wiring flexibility, scalability, and incremental expansion capability required in this environment. Several switch-based interconnects like Autonet [18], Myrinet [2], and ServerNet [9] have been proposed to build networks of workstations for cost-effective parallel computing. The widespread use of wormhole switching led manufacturers of network hardware to select this switching technique. Similarly to multicomputer networks, the use of virtual channels and adaptive routing increases performance. However, this increase is much higher in a network of workstations than in a multicomputer [21, 22]. The reason is that irregular network topologies lead to poor and unbalanced link utilization when the routing algorithm is not fully adaptive.

2. Motivation

Despite its widespread use, wormhole switching has serious drawbacks when implemented in a NOW. The distance between routers is much longer in a network of workstations than in a multicomputer. As a consequence, buffers must be very large in order to support channel pipelining [2]. Moreover, as clock frequency increases, buffer requirements also increase. Additionally, wire length is limited by buffer capacity [2]. Clearly, this situation is not desirable. On the other hand, in virtual cut-through buffer capacity is determined by packet size, being independent of wire length. Therefore, the use of virtual cut-through would allow a higher wiring flexibility as well as much longer links without having to re-design the router. Moreover, flow control is relatively complex in wormhole networks with channel pipelining and virtual channels [23]. Flow control is simplified when virtual cut-through is used because the unit of flow control is a packet instead of a flit.

Despite these advantages, virtual cut-through has some potential disadvantages. The traditional drawbacks of virtual cut-through with respect to wormhole switching are large buffer size and packetizing overhead. However, when used in a NOW environment, wormhole routers also require large buffers, as mentioned above. With a suitable design, virtual cut-through routers can be made at least as fast as wormhole routers. Regarding the overhead of splitting messages into packets, some messaging layers for NOWs increase performance by splitting messages into fixed-length packets [14]. Performance is increased because packets are pipelined through the network interface card, being concurrently transmitted from main memory to the interface card, and from there to the network. If messages were not packetized, message transmission could not start until the whole message is stored in the interface card memory. It should be noted that a large percentage of communication latency is due to the overhead in the messaging layer and the interface card [10, 12]. Therefore, the need for message packetization in NOWs removes the only remaining drawback of virtual cut-through switching in this environment.

In this paper, we show that virtual cut-through routers can be simpler than wormhole routers, while still achieving the advantages of virtual channels and adaptive routing. We evaluate the performance of different architectural decisions, concluding that similar performance can be achieved by both switching techniques in a NOW environment. However, VCT switches are much simpler than wormhole switches, requiring less silicon area and imposing less constraints on wire length.

The rest of the paper is organized as follows. In Section 3 we review the NOW environment. A new router design for VCT switching is proposed in Section 4. Also, a fully adaptive routing algorithm for VCT switching in a NOW environment is proposed in Section 5. In Section 6, the performance of the new routing strategy when implemented on both wormhole and VCT switches is evaluated. Finally, some conclusions are drawn.

3. Networks of Workstations

Routing decisions in irregular networks can be based on source routing or on distributed routing. In both cases, some network mapping algorithm must be executed in order to fill the routing tables before routing can be performed. Regardless of where decisions are taken, a routing algorithm must determine the path to be followed. Several deadlock-free routing algorithms [18, 2, 9, 16] and two general methodologies for the design of adaptive routing algorithms for irregular networks [21, 22] have been proposed. The routing algorithm for wormhole switching used in the performance evaluation section results from the application of one of these methodologies. In order to make the paper self-contained, we briefly present the routing algorithm upon which the methodology is applied in this paper, and also the resulting routing algorithm.

3.1. Up*/Down* Routing

Up*/down* routing [18] is based on an assignment of direction to the operational links; first, a spanning tree is computed. Then the "up" end of each link is defined as:

(1) the end whose switch is closer to the root in the spanning tree; (2) the end whose switch has the lower ID, if both ends are at switches at the same tree level. To avoid deadlocks this routing algorithm uses the following up*/down* rule: a legal route must traverse zero or more links in the "up" direction followed by zero or more links in the "down" direction. Thus, cyclic dependencies between channels are avoided because a packet cannot traverse a link in the "up" direction after having traversed one in the "down" direction. Such routing not only prevents deadlocks but also provides partial adaptivity. However, it is not always able to provide a minimal path between every pair of workstations [21].

3.2. Adaptive Routing

Recently, a general methodology for the design of adaptive routing algorithms for networks with irregular topology has been proposed in [21, 22]. Physical links are split into two virtual channels called the original and the new channels. Applying this methodology to the Autonet routing algorithm results in the following routing scheme: newly injected packets can only leave the source switch using new channels belonging to minimal paths. When a packet arrives at a switch through a new channel, the routing function gives a higher priority to the new channels belonging to minimal paths. If all of them are busy, then the up*/down* routing algorithm is used, selecting an original channel belonging to a minimal path (if any). To ensure deadlock-freedom, if none of the original channels supplied provides minimal routing, then the one that provides the shortest path will be used. Once a packet reserves an original channel, it will be routed using only original channels according to the up*/down* routing function until delivered. This routing algorithm will be referred to as MA-2vc.

4. Design Considerations

Typically, a wormhole router consists of a switch, flit buffers and link controllers associated with input and output links, and a routing and arbitration unit [8]. Virtual channels are usually implemented by using a separate flit buffer for each virtual channel. A virtual channel controller arbitrates among requesting packets and multiplexes virtual channels over the corresponding link on a flit-by-flit basis. In order to efficiently support flit-by-flit multiplexing, the switch should have as many ports as virtual channels. Thus, a non-multiplexed crossbar should be implemented. Figure 1 shows the organization of a wormhole router with virtual channels. The percentage of silicon area occupied by the crossbar becomes very high when a non-multiplexed crossbar is implemented, even for a relatively small number of virtual channels. As a consequence, some commercial WH routers implement a multiplexed crossbar [3], thus reducing silicon area at the expense of a complex arbitration scheme and higher propagation delay.

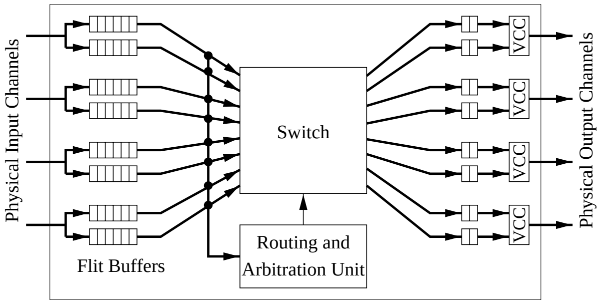


Figure 1. Organization of a wormhole router with virtual channels.

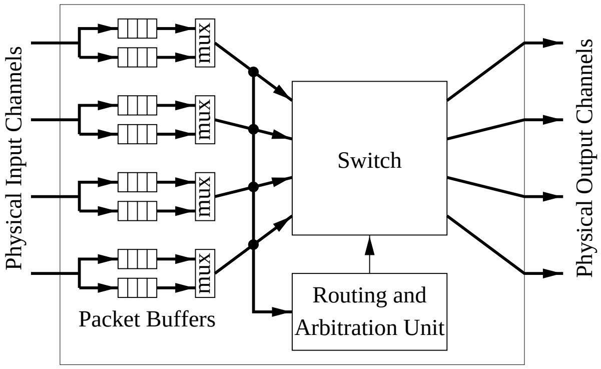


Figure 2. Organization of a virtual cut-through router with multiplexed crossbar.

The architecture of a virtual cut-through router could be very similar to that of a wormhole router. The mandatory differences are that buffers must be able to store one or more packets, and that flow control signals refer to packets instead of flits. However, it is possible to exploit the fact that buffers can store full packets to simplify virtual cut-through routers considerably.

We propose two router organizations for VCT. The main difference between them is the type of crossbar used [6]. Figure 2 shows a VCT router based on a multiplexed crossbar. It has the advantage of using a simpler crossbar with a lower number of ports, which may allow us to reduce clock cycle time. However, this router architecture introduces some contention among packets requesting the same switch input port, and may achieve a lower throughput than a WH router. Figure 3 shows a VCT router based on a partially multiplexed crossbar. Unlike for the VCT router in Figure 2, the throughput achieved by the VCT router with partially multiplexed crossbar is similar to the one for a WH router with non-multiplexed crossbar. Moreover, the crossbar is simpler than in the case of WH. As a consequence, VCT routers can be implemented using a considerably smaller

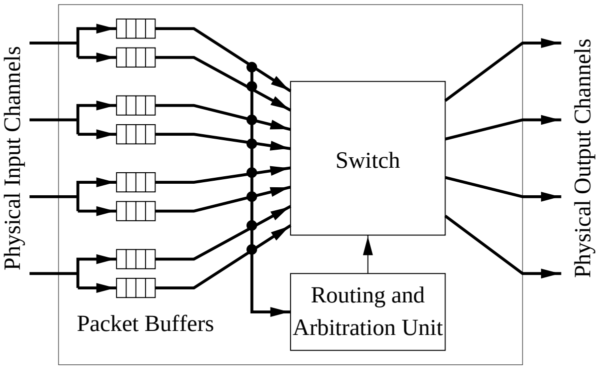


Figure 3. Organization of a virtual cut-through router with partially multiplexed crossbar.

silicon area than a WH router with similar performance. Additionally, propagation delay through a VCT router is lower than for a WH router because the virtual channel controllers associated with output links have been eliminated, thus reducing packet latency.

We have designed both WH and VCT routers using *Synopsis v1997.08* synthesis tool. The details of the designs are beyond the scope of this paper. The interested reader can refer to [15] for details on the designs, required silicon area, and propagation delays obtained.

5. A new routing algorithm for VCT

In VCT switching, the MA-2vc routing algorithm proposed for wormhole switching can be improved by increasing adaptivity, exploiting the fact that buffers can store full packets, which simplifies deadlock avoidance. Initially, we assume that the new routing strategy is implemented on a VCT router. Like the MA-2vc routing algorithm, the new routing scheme requires two queues associated with each physical link, referred to as new and original, respectively. Each queue can store one or more packets. From the point of view of routing, queues can be used as virtual channels in wormhole. Moreover, the MA-2vc algorithm could be implemented on a VCT router by making use of queues instead of virtual channels. The main difference between the new routing algorithm and the MA-2vc scheme is that the new routing algorithm allows new queues to be reserved after reserving original ones, which is forbidden in the MA-2vc algorithm. As a consequence, when a packet is being routed through original queues it can be stored in any queue, either new or original, associated with the incoming link at the next switch. Note that in the MA-2vc algorithm, these packets must be stored in original queues. This increased routing freedom is possible in VCT because packets are fully stored in buffers at intermediate switches when they are blocked, thus releasing the network resources occupied at previous switches. It can be proved that the new routing algorithm is deadlock-free by using the design methodology proposed in [7]. The new routing scheme will allow packets to follow minimal paths with higher probability than with the MA-2vc routing algorithm because, unlike the latter, the new algorithm continues providing fully adaptive minimal routing after having reserved an original queue.

Taking into account that the new routing algorithm allows packets to pass from new to original queues, and that routing on original queues may be non-minimal, livelocks may arise. However, the routing algorithm gives preference to minimal paths over non-minimal ones. This guarantees that as time goes to infinity, the probability of livelock goes to zero. A similar strategy was used in [11] to prove that the Chaos routing algorithm, which allows packets to follow both minimal and non-minimal paths, is livelock-free.

The proposed routing algorithm is fully adaptive because there is no restriction to pass from new queues to original ones and vice versa, thus allowing most packets to follow minimal paths. This algorithm is deadlock- and livelock-free. The new routing scheme can be also used on WH routers if buffer size is increased so that buffers can store an entire packet. In this case, queues are replaced by input buffers associated with virtual channels.

6. Performance Evaluation

In this section we evaluate the performance of the routing algorithm for VCT switching proposed in Section 5. We will refer to this routing algorithm as FA-2q. Each queue will have capacity for a single packet. Also, we will evaluate this new routing strategy when implemented on a wormhole router. We will refer to it as BFA-2vc. In this case, the buffer associated with each virtual channel has been enlarged so that it is be able to store an entire packet.

For comparison purposes, we have also evaluated the MA-2vc routing algorithm for a wormhole-switched network. We have also evaluated the MA-2vc routing strategy with a buffer size large enough to store an entire packet. We will refer to it as BMA-2vc.

6.1. Switch and Network Model

Although wormhole and VCT switches have some different features, some switch modules are similar in both architectures. In particular, each switch has a routing control unit that can only process one packet header at a time. It is assigned to waiting packets in a demand-slotted round-robin fashion. A crossbar inside the switch allows simultaneous multiple packet traversal. Table look-up routing is used.

In the switch model, we have assumed that it takes one clock cycle to compute the routing algorithm. Also, it takes one clock cycle to transmit one flit across the internal crossbar. On the other hand, data are injected into the physical link, which is pipelined, at a rate of one flit per cycle. The fly time is 4 cycles for each flit.

In the case for the wormhole switch, we assume a non-multiplexed crossbar and the flow control presented in [23]. According to [23], input buffer size has been set to 27 flits. With respect to the output buffer size, it will be 4 flits.

In the case for the VCT switch, we assumed that each queue can store a single packet. For the analysis presented in this paper, we have chosen a VCT switch based on a partially multiplexed crossbar.

The network is composed of a set of switches. Network topology is completely irregular and has been generated randomly. Only results for one random topology are presented. In [24] is shown that the randomness of the topology does not affect the relative behavior of the routing algorithms. For the sake of simplicity, we assumed first, that there are exactly 4 workstations connected to each switch. Second, two neighboring switches are connected by a single link, and finally, all the switches in the network have the same size. We assumed 8-port switches, thus leaving 4 ports available to connect to other switches. We have evaluated networks with a size of 16, 32, and 64 switches.

For each simulation run, we considered that packet generation rate is constant and the same for all the workstations. We have evaluated the full range of traffic, from low load to saturation. On the other hand, we have considered that packet destination is randomly chosen among all the workstations in the network. For packet length, 16-flit and 128-flit packets were considered.

6.2. Simulation Results

In this section, we analyze the performance of the proposed routing algorithm when implemented on wormhole switches with large buffers, comparing it with previously proposed routing algorithms. Then, we compare the performance of the new routing algorithm on wormhole and VCT switches when implemented with the same buffer capacity.

6.2.1 Performance of the new routing algorithm on wormhole switches

Figure 4 shows the average packet latency versus traffic for networks with 16, 32, and 64 switches, respectively. Packet length is 128 flits. As can be seen, the BFA-2vc algorithm reduces latency and increases throughput significantly with respect to the MA-2vc scheme in all cases. The improvement is due to the additional routing flexibility and larger buffer size provided by the BFA-2vc algorithm with respect to MA-2vc. In general, observing the behavior of the BMA-2vc strategy we note that for low and medium network loads, improvement on performance of BFA-2vc with respect to MA-2vc is mainly due to the increase in buffer size. As network load increases, the benefits of using a routing algorithm with higher adaptivity, such as BFA-2vc, become more noticeable. With respect to the influence of network size, one can see that as network size increases, the improvement with respect to MA-2vc achieved by the BFA-2vc routing strategy also increases. Moreover, the benefits of using a fully adaptive routing algorithm will be greater as network size increases. In the case for large networks (64 switches), BFA-2vc routing improves throughput by a factor of 2.2 with respect to MA-2vc, and 30% of this improvement is due to the increased adaptivity. In networks with 32 and 16 switches, BFA-2vc achieves 70% and 20% improvement in throughput, respectively, with respect to MA-2vc.

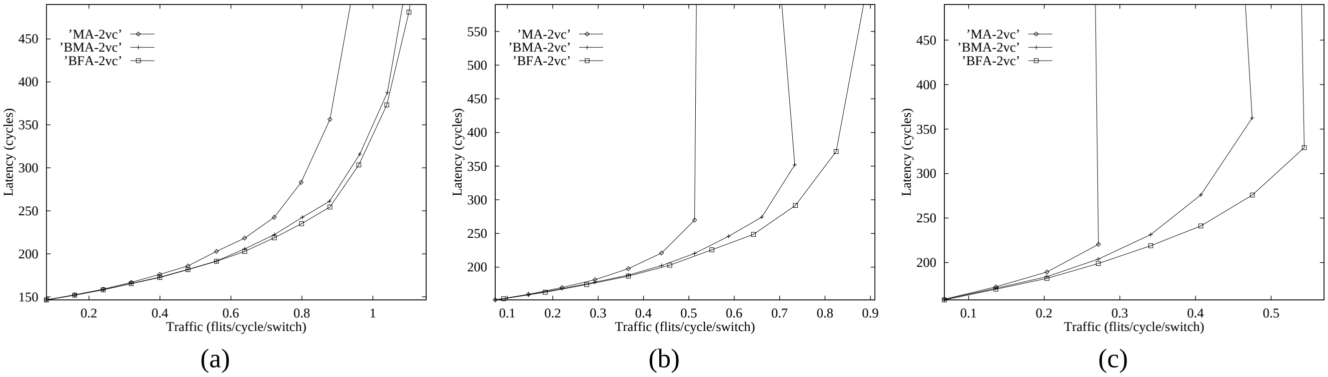
Analyzing the results, one can see that in small networks the benefits of using virtual cut-through switching are mainly due to the larger buffer size. However, in large networks, where distance between switches is longer, a routing algorithm that provides higher adaptivity and higher probability for routing through minimal paths, such as BFA-2vc, will strongly contribute to improve performance. In fact, in medium and large networks, approximately 30% of throughput improvement is due to the higher adaptivity.

Figure 5 shows some simulation results when packet length is 16 flits. With short packets, the increase in buffer size does not contribute to improve performance because the previous buffer size already was large enough to store an entire packet. In fact, the behavior of MA-2vc and BMA-2vc are similar. Thus, the improvement in performance achieved by the BFA-2vc routing algorithm is only due to the increase in adaptivity with respect to the MA-2vc algorithm. However, the improvement in performance is lower than in the case for long packets. Obviously, as network size increases, the advantages of using a fully adaptive routing strategy, such as BFA-2vc, also increase. The reason for this improvement is that the latency of short packets is very sensitive to the distance between switches. Thus, a routing algorithm that provides more minimal paths will strongly contribute to improve network performance. In the case for small networks (16 switches), the BFA-2vc scheme hardly contributes to increase performance with regard to MA-2vc. However, in networks with 32 and 64 switches, BFA-2vc achieves 20% and 35% improvement in throughput, respectively, with respect to MA-2vc.

6.2.2 Performance of the new routing algorithm on VCT switches

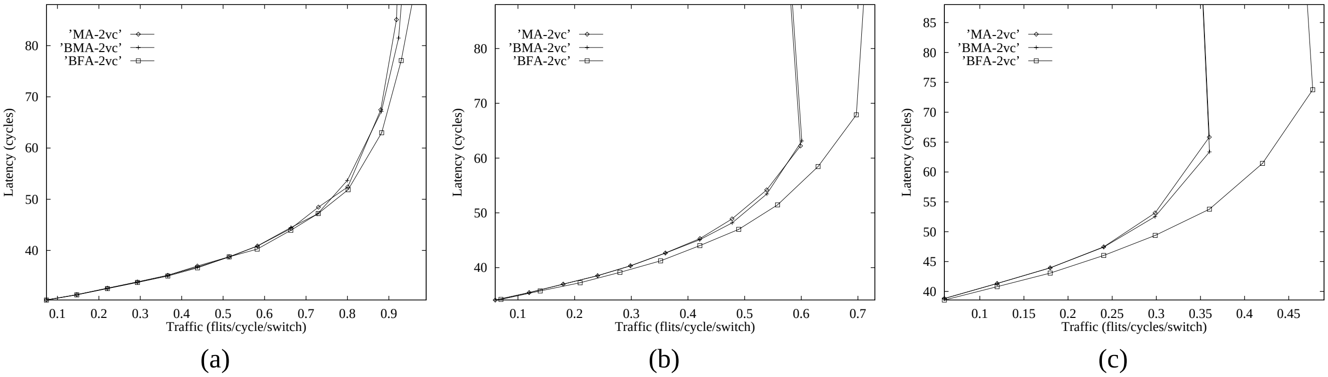
In this section, we analyze the behavior of the new routing algorithm proposed in Section 5 when implemented on the virtual cut-through switch with partially multiplexed crossbar that was proposed in Section 4. We compare it with the behavior of the same routing strategy implemented on a wormhole switch with the same buffer capacity. Thus, the difference in performance between FA-2q and BFA-2vc routing strategies will only be due to router complexity.

For comparison purposes, we have also analyzed the up*/down* routing algorithm on both switches. This algorithm will be referred to as UD-2q routing when implemented on VCT switches with two packet queues per physical link and BUD-2vc routing when implemented on worm-



(a) (b) (c)

Figure 4. Average packet latency versus traffic for an irregular network composed of (a) 16 switches, (b) 32 switches, and (c) 64 switches. MA, BMA, and BFA routing strategies are plotted. Packet length is 128 flits. Fly time is 4 cycles.



(a) (b) (c)

Figure 5. Average packet latency versus traffic for an irregular network composed of (a) 16 switches, (b) 32 switches, and (c) 64 switches. MA, BMA, and BFA routing strategies are plotted. Packet length is 16 flits. Fly time is 4 cycles.

hole switches with two virtual channels per physical link and buffers large enough to store an entire packet.

Figure 6 shows the average packet latency versus traffic for FA-2q and BFA-2vc routing strategies for networks with 16, 32, and 64 switches when packet length is 128 flits. As can be observed, both routing strategies achieve approximately the same message latency for the whole range of traffic. Nevertheless, the FA-2q strategy achieves a slightly higher throughput. The improvement in throughput increases as network size increases, achieving a 15% in networks with 64 switches.

This behavior is due to the fact that in VCT routers, when the requested physical output link is busy the router tries other possible physical output links even if there are empty buffers at the other side of the busy link. However, in wormhole, if the requested physical link has a free virtual channel then that channel is reserved by the packet even if other packets occupy virtual channels in the same physical link. If the physical link is busy at that time due to channel multiplexing, the packet must wait. In NOWs, channels are block-multiplexed instead of being multiplexed in a flit-by-flit basis [23]. As a result, routed packets may be waiting at the output buffers for long periods of time if packets are long. When traffic is high, the above situation is more frequent, causing a decrease in throughput for BFA-2vc with respect to FA-2q.

On the other hand, in the case for the FA-2q strategy, as routing is repeated each time the requested physical link is busy, the probability of routing through minimal paths is reduced, especially in small networks, due to the fact that minimal paths are tried first. Routing across non-minimal paths will increase packet latency. This will have a significant influence when packets are short, which justifies the behavior of the FA-2q strategy with respect to BFA-2vc.

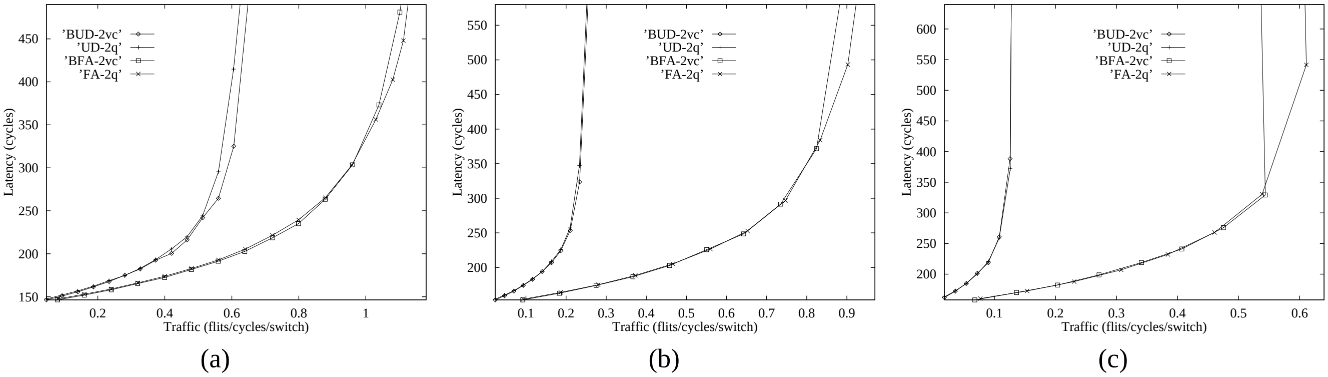


Figure 6. Average packet latency versus traffic for an irregular network composed of (a) 16 switches, (b) 32 switches, and (c) 64 switches. BUD, UD, BFA, and FA routing strategies are plotted. Packet length is 128 flits. Fly time is 4 cycles.

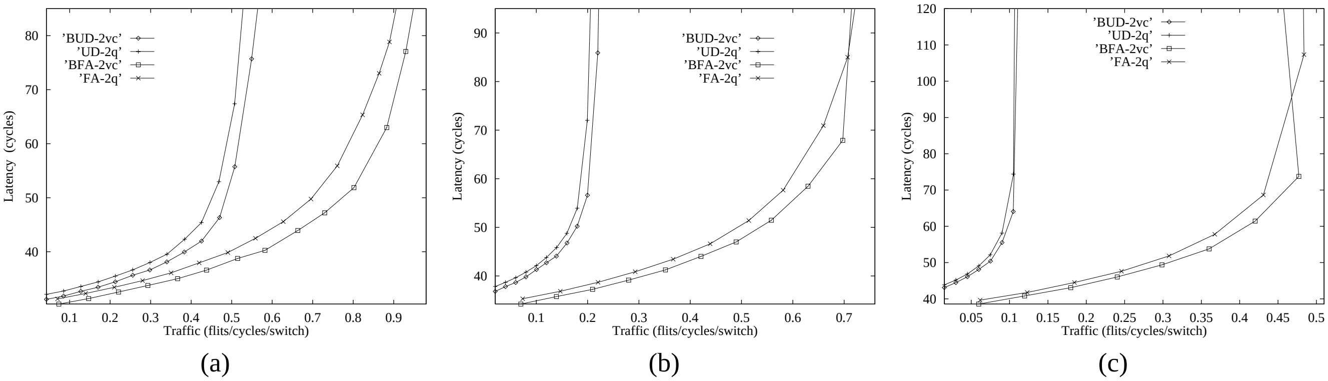


Figure 7. Average packet latency versus traffic for an irregular network composed of (a) 16 switches, (b) 32 switches, and (c) 64 switches. BUD, UD, BFA, and FA routing strategies are plotted. Packet length is 16 flits. Fly time is 4 cycles.

Figure 7 shows the results when packet length is 16 flits. In this case, the FA-2q strategy behaves worse than BFA-2vc for the whole range of traffic. It is specially noticeable for small networks and low traffic.

From the results above, we can conclude that differences in performance between the FA-2q and BFA-2vc strategies are not very significant. This is an interesting result because they can be implemented on a simpler and cheaper router, keeping the same clock cycle. Moreover, when packets are long, the FA-2q routing strategy achieves higher throughput than the BFA-2vc strategy, especially in large networks.

Also, we have analyzed the influence of varying the number of queues and buffers on the performance of the FA routing algorithm. We have observed that it is better to increase the number of queues instead of adding more buffers per queue. In particular, the FA routing algorithm with 3 queues and 1 buffer per queue achieves the best behavior from the point of view of cost/performance ratio.

7. Conclusions

In this paper, we compared the performance of wormhole and virtual cut-through (VCT) switching in a NOW environment. In order to perform such a comparison, we proposed a new fully adaptive routing algorithm for networks with irregular topology as well as a router architecture that exploits the features of VCT switching in a NOW environment in order to reduce silicon area. The comparison result suggests that, contrary to the current commercial trends, VCT is better suited than wormhole for NOWs, because they perform as efficiently as current solutions while requiring less resources and imposing less constraints on wire length. This claim is supported by the following observations:

- Buffer size limits wire length in wormhole networks. This is a serious drawback in NOWs. However, wire length is independent of buffer size in VCT networks. Buffers in VCT must be large enough to store full packets. Therefore, packets must be split into fixed-length packets, thus incurring packetization and depacketization overhead. However, some messaging layers for NOWs, like Fast Messages, increase performance by splitting messages into fixed-length packets, thus eliminating the drawbacks of packetization.

- When adaptive routing is used, VCT allows a higher routing flexibility than wormhole switching while preventing deadlock, because blocked packets are buffered and removed from the network. In this paper, we have proposed a new fully adaptive routing algorithm for networks with irregular topology. As adaptive buffers are used by following minimal paths, the proposed routing algorithm also increases the probability of routing through minimal paths, thus reducing packet latency. As shown in Section 6, the proposed fully adaptive routing algorithm outperforms previously proposed algorithms for wormhole switching when implemented in the same switch architecture.

- We have proposed a simple switch architecture for VCT switching that considerably reduces silicon area with respect to a comparable switch architecture for wormhole networks with the same buffer size. When implemented in the VCT switch, the proposed routing algorithm performs almost as efficiently as when implemented on a wormhole switch, despite the smaller silicon area required for its implementation. Nevertheless, the VCT switch outperforms the wormhole switch when packets are long, which is the case for FM.

As future work we plan to consider the impact of switch complexity on clock frequency in order to study whether virtual cut-through switches can be clocked at a higher frequency than wormhole switches, making virtual cut-through the switching technique of choice for general-purpose networks of workstations.

References

[1] A. Arruabarrena, et al., "A performance evaluation of adaptive routing in bidimensional cut-through networks," *Parallel Processing Letters*, vol. 3, no. 4, pp. 469–489, 1993.

[2] N. J. Boden, et al., "Myrinet - A gigabit per second local area network," *IEEE Micro*, pp. 29–36, February 1995.

[3] J. Carbonaro, et al., "Cavallino: The teraflops router and NIC," *Proc. of Hot Interconnects Symp. IV*, Aug. 1996.

[4] A. A. Chien, "A cost and speed model for k-ary n-cube wormhole routers," in *Proceedings of Hot Interconnects'93*, August 1993.

[5] W. J. Dally and C. L. Seitz, "The torus routing chip," *J. of Distributed Comp.*, vol. 1, no. 3, pp. 187–196, Oct. 1986.

[6] W. J. Dally, "Virtual-channel flow control," *IEEE Transactions on Parallel and Distributed Systems*, vol. 3, no. 2, pp. 194–205, March 1992.

[7] 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*, vol. 7, no. 8, pp. 841–854, August 1996.

[8] J. Duato, et al., *Interconnection Networks: An Engineering Approach*. IEEE Computer Society Press, 1997.

[9] R. Horst, "ServerNet deadlock avoidance and fractahedral topologies," *Proceedings of the 10th International Parallel Processing Symposium*, pp. 274–280, April 1996.

[10] V. Karamcheti and A. A. Chien, "Software overhead in messaging layers: Where does the time go?," *Proc. of the 6th Int, Conference on Architectural Support for Programming Languages and Operating Systems*, pp. 51–60, Oct. 1994.

[11] S. Konstantinidou and L. Snyder, "Chaos router: Architecture and performance," *Proc. of the 18th Int. Symposium on Computer Architecture*, pp. 79–88, June 1991.

[12] J. Miguel, et al., "Assessing the performance of the new IBM SP2 communication subsystem," in *IEEE Parallel and Distributed Technology*, pp. 12–22, Winter 1996.

[13] J. Y. Ngai and C. L. Seitz, "A framework for adaptive routing in multicomputer networks," *Proc. of the 1st Annual ACM Symp. on Par. Alg. and Architectures*, pp. 1–9, June 1989.

[14] S. Pakin, et al., "High performance messaging on workstations: Illinois fast messages on Myrinet," *Proceedings of Supercomputing 95*, November 1995.

[15] V. Puente, et al., "Bubble Router: A Fully Adaptive Virtual Cut-Through Switch for k-ary n-cube Networks," Submitted for publication to HPCA-5.

[16] W. Qiao and L. M. Ni, "Adaptive routing in irregular networks using cut-through switches," in *Proc. of the 1996 Int. Conf. on Parallel Processing*, Aug. 1996.

[17] J. Rexford and K. G. Shin, "Support for multiple classes of traffic in multicomputer routers," *Proceedings of the PCRCW'94*, pp. 116–130, May 1994.

[18] M. D. Schroeder et al., "Autonet: A high-speed, self-configuring local area network using point-to-point links," Technical Report SRC research report 59, DEC, April 1990.

[19] S.L. Scott and J.R. Goodman, "The impact of pipelined channels on k-ary n-cube networks," *IEEE Tran. on Parallel and Dist. Syst.*, vol. 5, no. 1, pp. 2–16, Jan. 1994.

[20] S. L. Scott and G. Thorson, "The Cray T3E network: adaptive routing in a high performance 3D torus," *Proceedings of Hot Interconnects Symposium IV*, August 1996.

[21] F. Silla, et al., "Efficient adaptive routing in networks of workstations with irregular topology," *Proceedings of CANPC'97*, pp. 46–60, February 1997.

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

[23] F. Silla and J. Duato, "On the use of virtual channels in networks of workstations with irregular topology," *Proceedings of PCRCW'97*, pp. 165–178, June 1997.

[24] F. Silla, *Routing and Flow Control in Networks of Workstations*, PhD. Thesis, Dept. of Information System and Computer Architecture, Universidad Politecnica de Valencia, Spain. Available at http://www.gap.upv.es

[25] C. B. Stunkel, et al., "The SP2 high-performance switch," *IBM Syst. Journal*, vol. 34, no. 2, pp. 185–204, Feb. 1995.