Technical Report #2012-2-1, Department of Computer Science and Engineering, Texas A&M University

Early Transition for Fully Adaptive Routing Algorithms in On-Chip Interconnection Networks

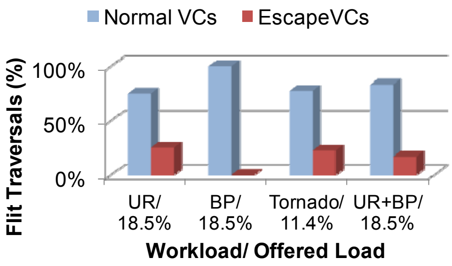
Minseon Ahn, Eun Jung Kim, and Ki Hwan Yum
Department of Computer Science and Engineering
Texas A&M University
College Station, TX 77843 USA
{msahn, ejkim, yum}@cse.tamu.edu

***Abstract*—Interconnection networks have become prevalent not only in massively parallel processing systems, but also in chip multiprocessors. Unlike off-chip interconnection networks, limited available resources in on-chip interconnection networks mandate to choose simple routing algorithms which, in turn, provide low throughput. Fully adaptive routing algorithms improve throughput, but need escape channels for a deadlock recovery technique resulting in low utilization. To improve utilization, we propose an early transition scheme where packets are transferred to the escape channels earlier, before the normal channels are full. Our evaluation results using a cycle-accurate network simulator show that our proposed scheme improves network throughput up to 12% in a concentrated mesh, compared to Duato's fully adaptive routing algorithm. Because the proposed scheme has better utilization in the escape channels, the early transition scheme is less sensitive to the ratio of the escape channels to the total channels.**

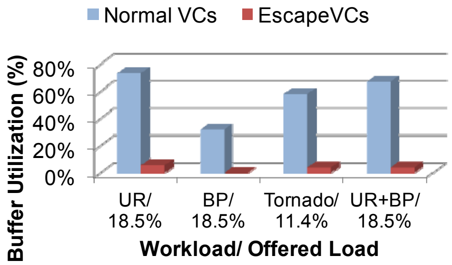
I. INTRODUCTION

Interconnection networks have been developed in massively parallel multiprocessor systems to connect processors, memories, and I/O devices. As the feature size keeps shrinking in CMOS process technology, chip multiprocessors (CMPs) are projected to have more than hundreds of processing cores in a single chip [1]. To support a large number of processing cores, interconnection networks replace shared buses due to the scalability problem. There are many studies on CMPs using on-chip interconnection networks, such as Intel 80-core Teraflop [10], Tilera 64-core [23], TRIPS [17], and RAW [21]. Compared to off-chip interconnection networks, on-chip interconnection networks have a limited area and power budget in a single chip. This constraint limits the number of hardware resources in the on-chip interconnection networks, such as the number of buffers and the number of virtual channels (VCs).

Because of the limited amount of hardware resources in on-chip interconnection networks, simple routing algorithms have been widely used, such as dimension order routing algorithms and O1TURN [18]. However, these routing algorithms suffer from poor network throughput, especially when hotspots exist. Theoretically, adaptive routing algorithms provide better performance than deterministic and oblivious routing algorithms because traffic is adaptively distributed around hotspots using network status information. To maximize adaptiveness, fully adaptive routing algorithms use all possible output ports as routing candidates without any restriction, thus resulting in better throughput.



(a) Flit Traversals



(b) Buffer Utilization

Figure 1. Traffic and Utilization in Duato's Fully Adaptive Routing Algorithm

To recover from a deadlock, the routing algorithms use escape channels [7], which occupy a small number VCs. These VCs are not utilized until a deadlock is detected. This condition causes low utilization of the escape channels because a deadlock can be detected only if the normal channels are full. Figure 1 shows that the normal channels have 58% utilization per VC, while the escape channels have less than 4% utilization per VC. If a network can provide many VCs, assigning a few VCs to the escape channels does not critically reduce the overall utilization. If, for example, off-chip interconnects can provide 16 VCs and 2 VCs are allocated to the escape channels, the average utilization is 51%, which is 7% lower than the utilization of the normal channels. Unlike off-chip interconnection networks, on-chip interconnection networks have limited numbers of VCs due to the limited resources. If there are only 4 VCs and the escape channels occupy 2 VCs, the average utilization can be 31%, which is around half of the utilization of the normal channels. Therefore, it is imperative to revisit the fully adaptive routing algorithm design in on-chip interconnection networks to improve buffer utilization.

In this paper, we propose an early transition scheme to increase the utilization of the escape channels. Our main idea is to use the escape channels not only for deadlock recovery but also for load-balancing. Packets are transferred to the escape channels if the queue occupancy of the normal channels is larger than the queue occupancy of the escape channels, which still guarantees that the routing algorithm is deadlock-free. By increasing the utilization of the escape channels, the early transition scheme improves network throughput.

Our evaluation results using a cycle-accurate network simulator show that the proposed early transition scheme increases network throughput by up to 12% compared to Duato's fully adaptive routing algorithm [7] under synthetic workload traffic in a concentrated mesh [2]. The escape channels support 13% ~ 24% more traffic and provide at least doubled buffer utilization. Performance is also improved up to around 8% in a flattened butterfly [12, 14] and a generic mesh. The proposed scheme no remarkable performance degradation even with the larger ratio of the escape channels to the total channels, showing that it is less sensitive to the number of the escape channels than Duato's fully adaptive routing algorithm.

The remainder of this paper is organized as follows. We briefly discuss related work in Section II. After providing background information on this work in Section III, we describe the proposed scheme in Section IV. In Section V, we illustrate the evaluation methodology, followed by presenting simulation results in Section VI. Finally, we conclude our work in Section VII.

II. RELATED WORK

To provide better throughput in the networks, there are many studies on deadlock prevention and deadlock-free routing algorithms in 2D mesh networks. The turn model [9] is introduced for deadlock avoidance of adaptive routing algorithms in mesh topologies. By limiting some turns, routing algorithms are completely free from deadlocks without any deadlock recovery scheme. Furthermore, PFNF (Positive First Negative First) [22] achieves better performance by extending adaptiveness with two turn models, one in each virtual network separately because it efficiently distributes all traffic symmetrically. Similarly, O1TURN [18] has two virtual networks each of which uses a different dimension order routing algorithm. These routing algorithms are either a partial adaptive routing algorithm or an oblivious routing algorithm, resulting in limited throughput improvement.

Deadlock-free fully adaptive routing algorithms have been studied to maximize adaptiveness. Duato [7] first proposes to use extra escape channels to prevent deadlocks. Basically, messages traverse through unrestricted normal channels until they are full. Thus, the escape channels suffer from low utilization. 3p routing algorithm [3] is the most recently proposed fully adaptive routing algorithm in 2D meshes, which divides VCs into two classes, waiting and non-waiting. It is deadlock-free because the waiting channels have two separate networks, positive and negative, and there is no cycle on channel dependency graphs in each network. However, this fully adaptive routing algorithm may have traffic imbalance because two separate networks are dependent on traffic directions.

There are several studies on fully adaptive routing algorithms in other topologies not applicable to on-chip interconnection networks. GOAL [20] and UGAL [19] are load-balanced fully adaptive routing algorithms in tori and hypercubes to achieve better network throughput by balancing the network workload into minimal and non-minimal paths. More recently, Kim et al. [13] propose adaptive routing algorithms in a folded-Clos network with high-radix routers [15]. These routing algorithms also provide load-balancing between multiple minimal paths in the network. They also propose the precision reduction of network information to minimize the hardware overhead of adaptive routing algorithms and pre-computation to minimize the impact on the router pipeline delay without significant performance degradation. Jiang et al. [11] propose several indirect ways of adaptive routing algorithms, such as credit round trip, progressive adaptive routing, piggyback routing, and reservation routing.

III. BACKGROUND

A. Baseline Router Microarchitecture

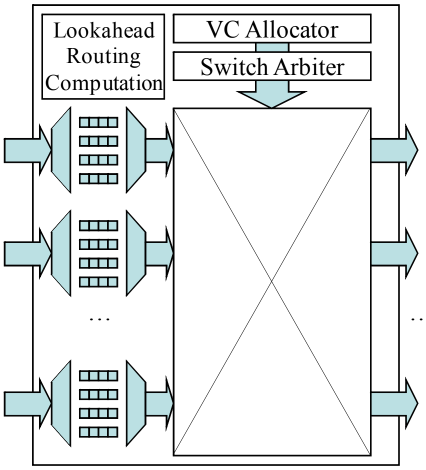


Figure 2. Baseline Router Microarchitecture

We use a state-of-the-art router architecture [16] shown in Figure 2. It has 2 pipelined stages in the routers performing virtual-channel allocation (VA) and switch arbitration (SA) at the first stage, and switch traversal (ST) at the second stage. Routing calculation (RC) is removed from the critical path by adopting lookahead routing [8] that generates routing information of the downstream router. SA is speculatively performed in parallel with VA. The VC allocator logic allocates one available VC at the input port of the downstream router. The switch arbiter logic arbitrates input and output ports of the crossbar. Once a flit is granted in SA stage, it enters the crossbar. After the flit traverses through the crossbar, it is sent to the downstream router during link traversal. We assume link traversal takes one cycle.

Each router has multiple VCs per input port to reduce head-of-line blocking by decoupling buffer resources from transmission resources. It uses flit-based wormhole switching [5] and credit-based VC flow control [4] for a small buffer cost to minimize the area cost in on-chip interconnection networks. This flow control provides back-pressure from downstream routers to upstream routers to avoid buffer overflow.

Communication messages are transmitted as packets. A sender network interface (NI) splits a packet into multiple flits to fit in the communication bandwidth for flow control and injects them serially. At the receiver NI, flits are reassembled to a packet after receiving all flits. The first flit of a packet is called a header flit, where routing information is stored, and the last flit is a tail flit. The other flits are called body flits. Once the header flit is routed to its destination according to its routing information, the remaining flits follow the header flit from source to destination.

B. *Deadlock Prevention*

A deadlock occurs when some packets in the network are blocked infinitely without any advance. If packets are waiting for network resources consumed by other packets, it makes a dependency. If the dependency becomes a cycle, it makes a deadlock. There are two techniques [6] to prevent deadlocks in routing algorithms. First, deadlock avoidance prohibits some turns to avoid cyclic dependency. Turn model [9] prevents some turns in routing algorithms to avoid circular dependency in mesh networks. The other turns are still allowed to the adaptive routing algorithms making routing algorithms partially adaptive.

The second technique is deadlock recovery which breaks the cyclic dependency when a deadlock is detected. Deadlock recovery techniques allow routing algorithms to make any turn to all possible directions until a possible deadlock is detected. Duato [7] proposes a simple and necessary condition to detect a deadlock. It simply checks if the normal channels are full. If so, there is a possibility of deadlocks. Otherwise, no deadlock happens. Once this conservative deadlock detection condition is satisfied, the packet is removed from the normal channels to the restricted escape channels to recover from a deadlock.

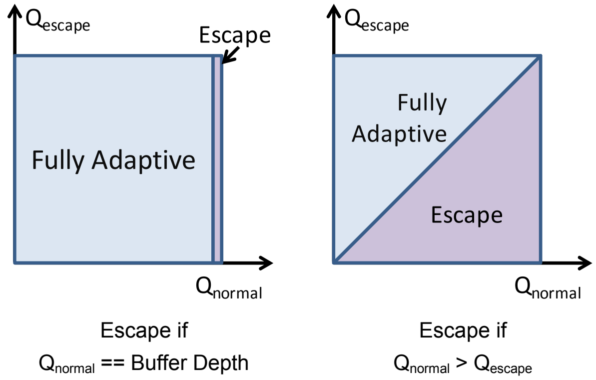
IV. EARLY TRANSITION FOR FULLY ADAPTIVE ROUTING ALGORITHMS

In this section, we describe an early transition scheme for fully adaptive routing algorithms in on-chip interconnection networks to improve network throughput.

A. *Increasing Escape Channel Utilization*

Virtual channels [4] are proposed to reduce head-of-line (HOL) blocking by virtually dividing one physical channel into several VCs. These virtual channels are also used to provide separate virtual networks for either deadlock avoidance (PFNF [22] or O1TURN [18]) or deadlock recovery (escape channels in Duato's fully adaptive routing algorithm [7]).

Fully adaptive routing algorithms generally use a deadlock recovery technique by adopting escape channels where routing algorithms must be deadlock-free [7], such as dimension order routing (DOR) algorithms. Generally, a deadlock could happen only if there is no available buffer space in the normal virtual channel. Therefore, the escape channels are not utilized until there is a possibility of a deadlock. In the previous deadlock recovery scheme, the escape channels have low utilization because packets are not traversing until the unrestricted normal channels for adaptive routing algorithms are full as shown in Figure 3 (a). We observe that this could cause low utilization in the escape channels, resulting in earlier network saturation.

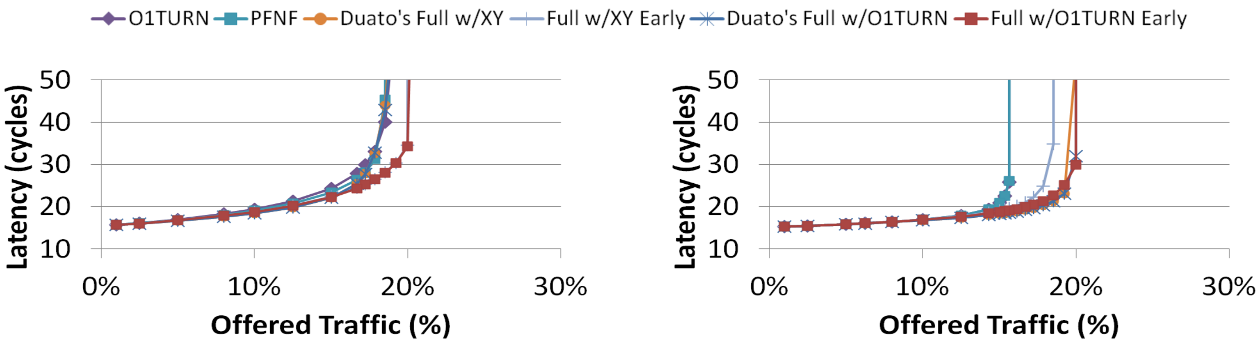


(a) Duato's Fully Adaptive Routing Algorithm

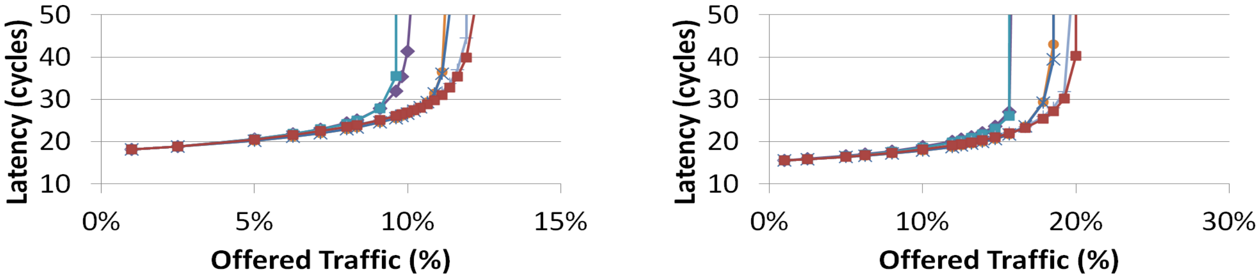
(b) Early Transition

Figure 3. Flit Transition to Escape Channels

To solve this early saturation problem, we propose an early transition scheme as shown in Figure 3 (b). To increase the utilization of the escape channels, it transfers packets to the escape VCs earlier, before the normal channels are full. Packet transfer is performed when the queue occupancy of the escape VCs is smaller than the queue occupancy of the unrestricted normal VCs.



(a) UR (b) BP



(c) Tornado (d) UR+BP

Figure 4. Overall Performance in a Concentrated Mesh

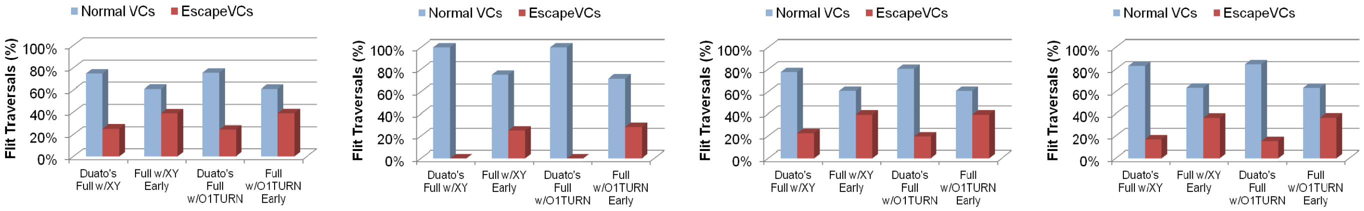
There are two conditions to make fully adaptive routing algorithm deadlock-free [7]. First, the routing algorithm used in the escape channels must be deadlock-free. Second, the deadlock detection condition necessarily detects deadlocks. The proposed early transition scheme still uses deterministic routing algorithms, which satisfies the first condition. The second condition is also satisfied since it transfers packets to the escape channels when the normal channels are full in the proposed scheme. Therefore, it makes the routing algorithms still deadlock-free.

To achieve further load-balancing in all directions, we use both DOR algorithms like O1TURN [18] in the escape VCs instead of one DOR algorithm. We observe that using a simple DOR algorithm causes non-uniformity in the network physical channels of the escape channels. Unlike the DOR algorithm, O1TURN achieves load-balancing by using two orthogonal DOR algorithms (XY and YX) together. To avoid deadlocks, it partitions VCs into two virtual networks and each DOR algorithm is assigned to each virtual network. To accommodate two DOR algorithms in the escape channels, all escape channels must be partitioned into two, each of which uses one DOR algorithm independently. When a packet enters the escape VCs, one of two virtual networks is randomly selected and the same DOR algorithm is used in the same virtual network until the packet arrives at the destination.

B. Reducing the Overhead of Adaptive Routing Algorithms

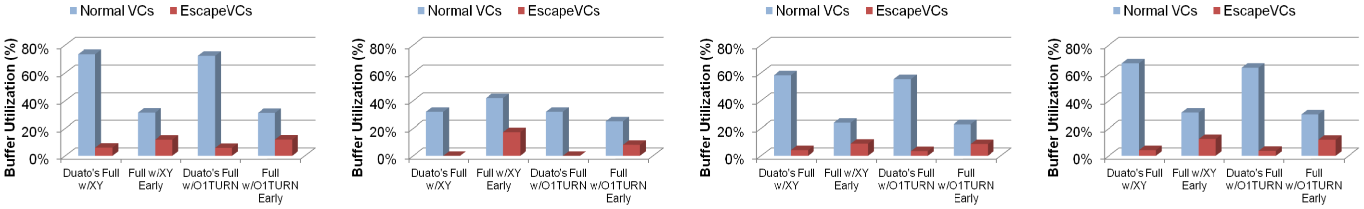
To make a fair comparison with other deterministic or oblivious routing algorithms, the computational overhead of adaptive routing algorithms is minimized by reducing the precision in credit information and pre-computing the allocation [13]. First, precision reduction in credit information has a little performance degradation compared to full precision. If each virtual channel has *n*-flit buffers, $log_2\ n$ bits are needed for credit information. Since the buffer depth in each VC is 4 in our experiment, the number of bits in the credit information is still minimal. In addition, the credit information is changing by 1 per cycle. There are not many changes within a couple of cycles. Therefore, applying adaptiveness in lookahead routing has a minimal hardware overhead. Another way to reduce the computation overhead is pre-computation. With the credit information of the previous cycle, the routing calculation can be pre-computed before a packet is arriving. Basically pre-computation loss is only minimal because the difference of the credit information is maximally just one per cycle. Kim et al. [13] show that reducing the precision in credit information and pre-computing minimize the computation costs without significant performance degradation compared to the full precision and no pre-computation.

Fully adaptive routing algorithms check a deadlock detection condition during VA stage. If the condition is satisfied, it moves packets to one of escape channels after



(a) UR 18.5% (b) BP 18.5% (c) Tornado 11.4% (d) UR+BP 18.5%

Figure 5. Flit Traversals on Workload and Offered Load



(a) UR 18.5% (b) BP 18.5% (c) Tornado 11.4% (d) UR+BP 18.5%

Figure 6. Buffer Utilization on Workload and Offered Load

performing escape RC/VA. To avoid performance overhead, escape RC/VA can be performed in parallel with VA. If the deadlock condition is not satisfied, escape RC/VA information is ignored and unused. Since the routing algorithm used in the escape channels is simple compared to the fully adaptive routing algorithm, the hardware overhead is minimal, compared to the fully adaptive routing algorithm.

V. EXPERIMENTAL METHODOLOGY

We evaluate performance using our cycle-accurate on-chip network simulator implementing pipelined routers with buffers, VCs, arbiters, and crossbars. The network simulator is configured with 4-flit buffer per each VC and 4 VCs per each input port. We assume that the bandwidth of a link is 128 bits with additional error correction bits. We use a concentrated mesh topology [18] for on-chip interconnection networks where each router connects 4 communication nodes and routers are connected as a 2D mesh topology. In this topology, the number of communication nodes is 64, and the number of routers is 16.

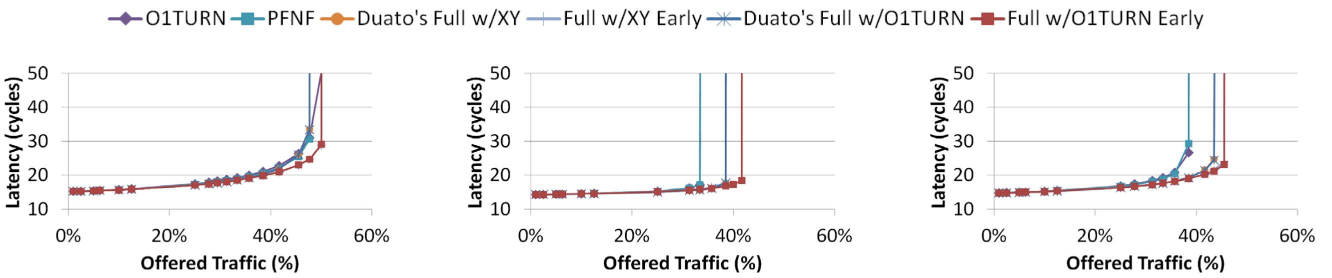
To compare with the most recently proposed routing algorithms in 2D meshes, we use O1TURN [18] and PFNF [22]. For fair comparison to O1TURN, we use minimal adaptive routing algorithms which select the best candidate among those in the shortest paths only. We also use dynamic VA policy to maximize network throughput and latency. It chooses an output VC where the number of available buffers is the greatest among all possible VCs in the downstream router. If there is no available buffer in all candidate output VCs, VA fails. To select the best VC, we use credit information. Thus, it has evenly distributed workload between VCs of the same kind. This allocation policy is generally used in on-chip interconnection networks.

To apply two DOR algorithms in the escape channels, we need at least two VCs in the escape channels. Since there are 4 VCs per input port, we assign 2 VCs to the normal channels and the other 2 VCs to the escape channels. To make a fair comparison, we assign the same number of VCs to the escape channels in every configuration of fully adaptive routing algorithms.

In this experiment, we use several different basic patterns of synthetic workload traffic. First, we use uniform random (**UR**) traffic, where the destination is uniformly distributed to all nodes in the network. Thus, it has equal chances to use all links. The second traffic is bit permutation (**BP**) whose communication pattern is generated based on matrix transpose. This traffic has only one destination for each source. Due to symmetry on patterns, it does not have any deadlock, but creates spatial hotspots in the diagonal line. The last traffic is tornado (**Tornado**), where all communications are going clockwise in 2D meshes and the destinations are 4-hop away, resulting in asymmetric usage of links. To see performance in complicated workload traffic, we also create one more pattern, mixing UR and BP together (**UP+BP**) to create complicated and dynamic hotspots in time and space. In the experiment with synthetic workload traffic, all packets are 5 flit long.

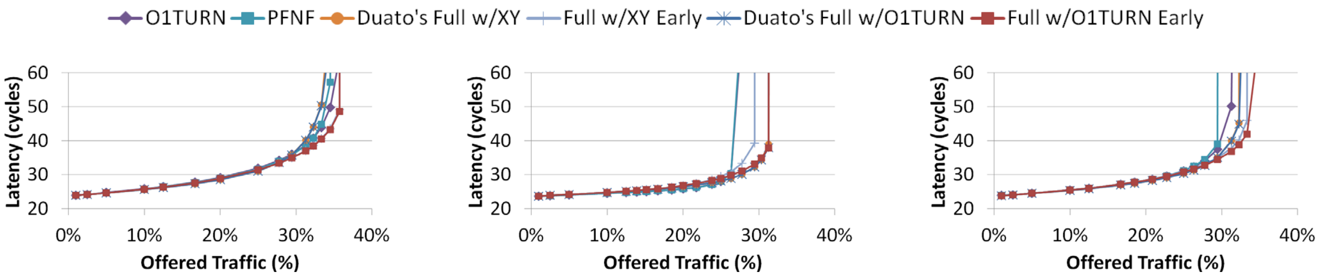
VI. PERFORMANCE EVALUATION

In this section, we evaluate the proposed load-balancing techniques. To compare performance with other routing algorithms, we also use two previously proposed routing algorithms, such as O1TURN [18] and PFNF [22]. In this section, the fully adaptive routing algorithm using XY in the escape channels is indicated by **Full with XY**, the fully adaptive routing algorithm using two DOR algorithms in the escape channels is indicated by **Full with O1TURN**.



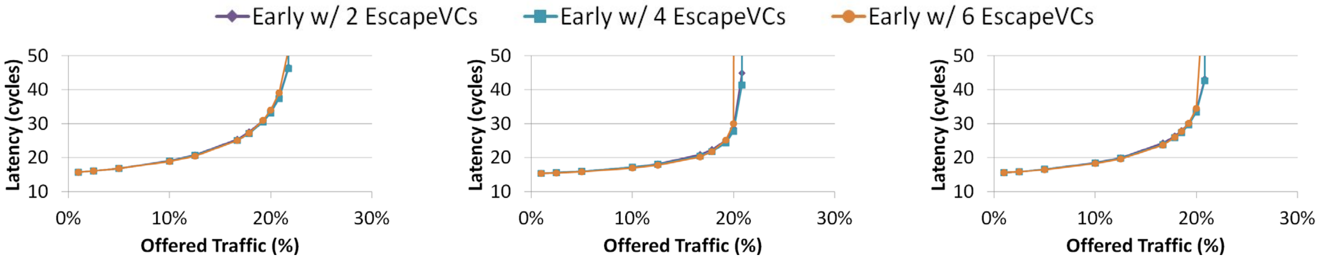
(a) UR (b) BP (c) UR+BP

Figure 7. Overall Performance in a Flattened Butterfly

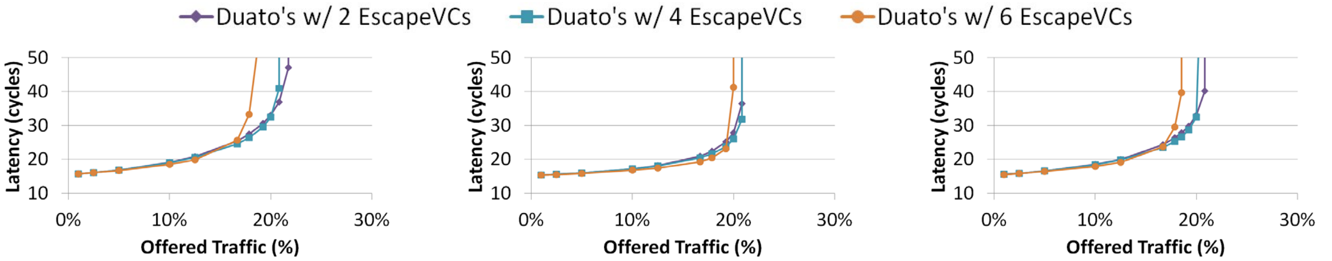


(a) UR (b) BP (c) UR+BP

Figure 8. Overall Performance in a Generic Mesh



(a) UR in Early Transition (b) BP in Early Transition (c) UR+BP in Early Traistion



(d) UR in Duato's (e) BP in Duato's (f) UR+BP in Duato's

Figure 9. Performance Sensitivity on the Number of Escape VCs

A. Performance Evaluation with Basic Synthetic Workload Traffic

We evaluate performance of the proposed early transition scheme in load-balanced fully adaptive routing algorithms in a concentrated mesh [2] as shown in Figure 4. Since UR has equally distributed traffic to all possible directions, there is no performance difference between **Full with XY** and **Full with O1TURN**. Figure 4 (a) shows that the proposed early transition scheme increases network throughput by 12% normalized to Duato's routing algorithm. We observe that UR has temporal hotspots randomly with a small amount of traffic burst. BP has hotspots in the diagonal line because all traffic is concentrated to those hotspot routers. Thus, **Full with XY** using the early transition scheme has performance degradation compared to **Full with XY** using Duato's routing algorithm. However, **Full with O1TURN** has better performance as shown in

Figure 4 (b) because O1TURN distributes traffic into both dimensions in the escape channels. The other two synthetic workload patterns show the performance difference in detail. Inherently, O1TURN has better performance than any other dimension order routing algorithm [18]. Consequently, the routing algorithm used in the escape channels affects network performance. Besides, the proposed early scheme increases the buffer utilization in the escape channels as shown in Figure 6. Therefore, **Full with O1TURN** using the early transition scheme has the best performance because of the additional improvement in the escape channels. Overall, it achieves 12% normalized network throughput improvement in Tornado and UR+BP, compared to **Full with XY** using Duato's routing algorithm.

Figure 5 shows the percentage of flit traversals in the normal channels and the escape channels. The proposed early transition scheme increases traffic in the escape channels 13% ~ 24% compared to Duato's routing algorithm. In other words, the early transition scheme tries to evenly distribute traffic workload to both channels. Figure 6 shows the buffer utilization, the percentage of buffer occupancy on average. The early transition, at least, doubles the buffer utilization in the escape channels. Since the proposed early transition scheme does not transfer packets to the escape channels if both queue occupancies are equal. Thus, the normal channels still have higher buffer utilization than the escape channels. Note that packets are still traversing in the normal channels if both queue occupancies are equal. Therefore, the escape channels still have less utilization than the normal channels.

B. *Performance Evaluation in Other Topologies*

We evaluate performance of the proposed early transition in load-balanced fully adaptive routing algorithms in other topologies, such as a flattened butterfly [12, 14] and a general mesh. The flattened butterfly has the same concentration as the concentrated mesh. However, it has additional express channels unlike the concentrated mesh. Since we assume that each link has the same bandwidth as in the concentrated mesh, this topology can accept more communication. Because the express channels can bypass hotspots without any adaptiveness, performance improvement in Flattened Butterfly is smaller than in the concentrated mesh with UR and UR+BP traffic. However, BP has up to 8.3% throughput enhancement, compared to Duato's routing algorithm, because the express channel can separate flows in this synthetic workload reducing hotspot traffic as shown in Figure 7.

To support the same number of cores, the mesh topology has 64 routers, building an 8x8 network. Thus, it has higher communication latency because of the larger average number of hops. Similarly in the concentrated mesh, **Full with XY** using the early transition scheme has performance degradation in BP, but **Full with O1TURN** using the early transition scheme still is as good as **Full with XY** using Duato's routing algorithm in this synthetic workload traffic. However, network throughput is improved with **Full with O1TURN** using the early transition scheme in UR and UR+BP by 7.14% and 3.35%, respectively, compared to **Full with XY** using Duato's routing algorithm as shown in Figure 8.

C. *Effect on Multiple Virtual Channels*

We also conduct an experiment with several different configurations on the number of VCs in the escape channels to see the sensitivity on the number of escape channels. In this experiment, we use fully adaptive routing algorithms in the normal channels and O1TURN in the escape channels, in a concentrated mesh. To make more configurations on the number of escape VCs, we increase the total number of virtual channels to 8 and generate 3 different configurations. The first configuration has 6 normal and 2 escape VCs. The second has 4 normal and 4 escape VCs. The last has 2 normal and 6 escape VCs. Figure 9 shows that the proposed early transition scheme has similar performance regardless of the number of the escape VCs whereas, Duato's routing algorithm has significant performance degradation when the number of the escape VCs is larger than that of the normal VCs. This is because the proposed scheme increases the utilization of the escape channels.

VII. CONCLUSIONS

On-chip interconnection networks have been used in chip multiprocessors as communication architecture. Unlike off-chip interconnection networks, on-chip interconnection networks have limited resources due to area and power budget in a single chip. With the limited resources, on-chip interconnection networks have a small number of VCs. Consequently, fully adaptive routing algorithms with a deadlock recovery technique have low utilization. To increase the utilization, we propose an early transition scheme for fully adaptive routing algorithms in on-chip interconnection networks. This scheme moves packets to escape channels earlier, before the normal channels are full. Our cycle-accurate network simulator reveals that the proposed scheme improves network throughput up to 12% in a concentrated mesh because of better utilization of the escape channel than Duato's fully adaptive routing algorithm. It increases traffic in the escape channels up to 13% ~ 24% because of doubled buffer utilization. The proposed early transition scheme improves performance by around 8% in a flattened butterfly and a generic mesh. It becomes less sensitive to the number of VCs in the escape channels because the escape channels have better utilization.

To achieve perfect load-balancing between the escape channels and the normal channels, we will improve the early transition scheme for future work. Furthermore, we will also develop the early transition scheme to apply to other adaptive routing algorithms, such as UGAL in a flattened butterfly.

REFERENCES

[1] The International Technology Roadmap for Semiconductors, http://www.itrs.net/

[2] J. Balfour and W. J. Dally, "Design Tradeoffs for Tiled Cmp on-Chip Networks," in *Proceedings of the 20th annual International Conference on Supercomputing (ICS)*, Cairns, Queensland, Australia, 2006.

[3] Y. M. Boura and C. R. Das, "Efficient Fully Adaptive Wormhole Routing in N-Dimensional Meshes," in *Proceedings of the 14th International Conference on Distributed Computing Systems (ICDC), 1994.*, 1994, pp. 589-596.

[4] W. J. Dally, "Virtual-Channel Flow Control," *IEEE Transactions on Parallel and Distributed Systems (TPDS),* vol. 3, pp. 194-205, 1992.

[5] W. J. Dally and C. L. Seitz, "Deadlock-Free Message Routing in Multiprocessor Interconnection Networks," *IEEE Transactions on Computers,* vol. C-36, pp. 547-553, 1987.

[6] W. J. Dally and B. Towles, *Principles and Practices of Interconnection Networks*: Morgan Kaufmann, 2003.

[7] J. Duato, "A New Theory of Deadlock-Free Adaptive Routing in Wormhole Networks," *IEEE Transactions on Parallel and Distributed Systems (TPDS),* vol. 4, pp. 1320-1331, 1993.

[8] M. Galles, "Scalable Pipelined Interconnect for Distributed Endpoint Routing: The Sgi Spider Chip," in *Proc. Hot Interconnects*, 1996, pp. 141-146.

[9] C. J. Glass and L. M. Ni, "The Turn Model for Adaptive Routing," in *Proceedings of the 19th annual International Symposium on Computer Architecture (ISCA)* Queensland, Australia: ACM, 1992.

[10] Y. Hoskote, S. Vangal, A. Singh, N. Borkar, and S. Borkar, "A 5-Ghz Mesh Interconnect for a Teraflops Processor," *IEEE Micro,* vol. 27, pp. 51-61, 2007.

[11] N. Jiang, J. Kim, and W. J. Dally, "Indirect Adaptive Routing on Large Scale Interconnection Networks," in *Proceedings of the 36th annual International Symposium on Computer Architecture (ISCA)* Austin, TX, USA: ACM, 2009, pp. 220-231.

[12] J. Kim, J. Balfour, and W. Dally, "Flattened Butterfly Topology for on-Chip Networks," in *Proceedings of the 40th Annual IEEE/ACM International Symposium on Microarchitecture*, 2007.

[13] J. Kim, W. J. Dally, and D. Abts, "Adaptive Routing in High-Radix Clos Network," in *Proceedings of the 2006 ACM/IEEE conference on Supercomputing* Tampa, Florida: ACM, 2006, p. 92.

[14] J. Kim, W. J. Dally, and D. Abts, "Flattened Butterfly: A Cost-Efficient Topology for High-Radix Networks," in *Proceedings of the 34th annual International Symposium on Computer Architecture (ISCA)* San Diego, California, USA: ACM, 2007, pp. 126-137.

[15] J. Kim, W. J. Dally, B. Towles, and A. K. Gupta, "Microarchitecture of a High Radix Router," in *Proceedings of the 32nd annual International Symposium on Computer Architecture (ISCA)*, 2005, pp. 420-431.

[16] L.-S. Peh and W. J. Dally, "A Delay Model and Speculative Architecture for Pipelined Routers," in *Proceedings of the 7th International Symposium on High-Performance Computer Architecture (HPCA)*, 2001.

[17] K. Sankaralingam, R. Nagarajan, L. Haiming, K. Changkyu, H. Jaehyuk, D. Burger, S. W. Keckler, and C. R. Moore, "Exploiting Ilp, Tlp, and Dlp with the Polymorphous Trips Architecture," in *Proceedings of the 30th Annual International Symposium on Computer Architecture (ISCA)*, 2003, pp. 422-433.

[18] D. Seo, A. Ali, W.-T. Lim, N. Rafique, and M. Thottethodi, "Near-Optimal Worst-Case Throughput Routing for Two-Dimensional Mesh Networks," in *Proceedings of the 32nd annual International Symposium on Computer Architecture (ISCA)*, 2005.

[19] A. Singh, "Load-Balanced Routing in Interconection Networks," *PhD Thesis, Stanford University,* 2005.

[20] A. Singh, W. J. Dally, A. K. Gupta, and B. Towles, "Goal: A Load-Balanced Adaptive Routing Algorithm for Torus Networks," in *Proceedings of the 30th annual International Symposium on Computer Architecture (ISCA)* San Diego, California: ACM, 2003, pp. 194-205.

[21] M. B. Taylor, J. Psota, A. Saraf, N. Shnidman, V. Strumpen, M. Frank, S. Amarasinghe, A. Agarwal, W. Lee, J. Miller, D. Wentzlaff, I. Bratt, B. Greenwald, H. Hoffmann, P. Johnson, and J. Kim, "Evaluation of the Raw Microprocessor: An Exposed-Wire-Delay Architecture for Ilp and Streams," in *Proceedings on the 31st Annual International Symposium on Computer Architecture (ISCA)*, 2004, pp. 2-13.

[22] J. H. Upadhyay, V. Varavithya, and P. Mohapatra, "Efficient and Balanced Adaptive Routing in Two-Dimensional Meshes," in *Proceedings of the First IEEE Symposium on High-Performance Computer Architecture (HPCA)*, 1995, pp. 112-121.

[23] D. Wentzlaff, P. Griffin, H. Hoffmann, B. Liewei, B. Edwards, C. Ramey, M. Mattina, M. Chyi-Chang, J. F. Brown, and A. Agarwal, "On-Chip Interconnection Architecture of the Tile Processor," *IEEE Micro,* vol. 27, pp. 15-31, 2007.