

CARM: Congestion Adaptive Routing Method for On Chip Networks

Manoj Kumar^{1,2}, Vijay Laxmi¹, Manoj Singh Gaur¹, Seok-Bum Ko² and Mark Zwolinski³

¹Malaviya National Institute of Technology, Jaipur, India

bohra.manoj1980@gmail.com, vlaxmi@mnit.ac.in, gaurms@mnit.ac.in

²University of Saskatchewan, Saskatoon, Canada

seokbum.ko@usask.ca

³University of Southampton, Southampton, United Kingdom

mz@ecs.soton.ac.uk

Abstract—Network-on-Chip (NoC) has emerged as a long-term and efficient on-chip communication solution for MC-SoC and CMP micro-architectures to overcome bottleneck of traditional bus-based interconnects. Performance of NoC is highly dependent on routing algorithm we choose. In this paper, we present a highly adaptive and deadlock free routing algorithm for 2D mesh topology to mitigate congestion. Proposed algorithm provides a high degree of adaptiveness by allowing cycles in channel dependency graph and using one additional virtual channel along the Y dimension only. It uses all available minimal/non-minimal paths between source and destination nodes. A packet is routed along the non-minimal path only when minimal paths get congested at the neighboring nodes. Results show that proposed congestion-aware routing algorithm improves network performance by routing packets through non-congested areas.

Keywords—Networks on Chip, routing, non-minimal paths, congestion, deadlock freedom.

I. INTRODUCTION

On-chip interconnection networks have been aggressively researched and discussed in recent past years as a dominant and promising communication infrastructure for complex Chip Multi Processors (CMPs) and Multi Core Systems on Chips (MCSocS) because of their increased predictability, reusability, scalability, energy-efficiency and reliability [1]. The choice of routing algorithm plays an important role in deciding performance and efficiency of NoCs. Higher adaptiveness of the routing algorithm reduces probability for a packet to get into faulty or congested region. In this work, we focus on improving performance of fully adaptive routing algorithm by providing high degree of adaptiveness and congestion awareness.

Network congestion may result into increased power consumption and transmission latency, and thus limits the performance of NoC. However, the performance can be improved enough by mis-routing packets through less congested paths and flattening the distribution of traffic over the network. Minimal routing methods guarantee the shortest route from source to destination. However, it is imprudent to ignore the promising performance benefits provided by non-minimal routing methods. For example, if all output channels corresponding to shortest routing paths are faulty;

mis-routing the packets along non-minimal route may be the only viable alternative. Thus, in this paper, we present congestion-aware, non-minimal and fully adaptive routing algorithm. It offers high degree of adaptiveness by permitting cycles in channel dependency graph. It is shown experimentally that the presented routing method assures significant performance improvement compared to existing routing methods for different traffic patterns.

II. RELATED WORK

The overall performance of on-chip network depends on many network parameters such as topology, flow control mechanism, routing method and switching technique. In this paper, we mainly focus on routing method. Most of the routing algorithms use turn model or virtual channel based methodologies to achieve deadlock freedom. Many routing algorithms [2]–[5] exist in literature that use turn models for deadlock avoidance. The e-cube routing (XY routing) prohibits turn from Y dimension to X dimension as shown in Figure 1a. Glass and Ni [2] proposed three routing algorithms namely west-first, north-last, and negative-first for n-dimensional mesh network. These routing algorithms provide deadlock freedom by restricting one turn in each abstract cycle (clockwise and counter-clockwise) as shown in Figure 1. Hu and Marculescu [3] proposed a routing al-

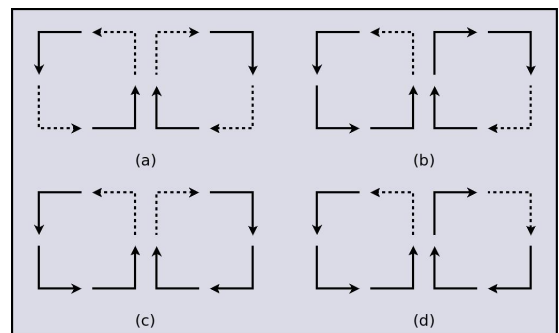


Figure 1: Turn models (a) XY (b) west-first (c) north-last (d) negative-first (solid lines for permitted turns and dash lines for prohibited turns)

gorithm called DyAD, that combines low latency advantage of deterministic routing at low loads and high throughput advantage of adaptive routing. When the network is not congested, DyAD router is operated under deterministic mode. It switches to adaptive mode as the network gets congested.

Another popular method of designing deadlock free routing algorithms with higher adaptiveness, is to add virtual channels to the network. Several minimal/non-minimal and fully adaptive routing algorithms, based on a small number of additional virtual channels, have been proposed in [6]–[13].

The routing algorithms presented in [6], [7] produce an equivalent minimal and fully adaptive routing algorithm for 2D mesh, called double-y. It requires two virtual channels in Y dimension and one virtual channel in X dimension. In [8], authors proposed maximally adaptive double-y routing algorithm (Mad-y) that is an improvement over double-y network based algorithms [6], [7] and makes better use of available resources (virtual channels) to increase adaptiveness. Ming Li et al. [9] introduced congestion-aware dynamic routing algorithm DyXY that determines output channel on the basis of congestion status of buffers of adjacent nodes. RCA [10], DAR [11], DBAR [12] and CATRA [13] are congestion-aware fully adaptive routing algorithms that use non-local congestion information to route the packet using extra hardware.

III. PROPOSED METHOD

In this section, we present minimal/non-minimal, deadlock-free and congestion-aware fully adaptive routing method CARM (Congestion Adaptive Routing Method) for 2D NoC mesh topology. Goal of CARM is to increase the capability of existing virtual channels in Mad-y to route (minimally) or misroute (non-minimally) packets around congested and hot-spot regions. It deploys double-y network that uses one virtual channel along X dimension and two virtual channels along Y dimension to achieve high degree of adaptiveness. CARM uses turn model, which is extension and improvement over turn model used by Mad-y algorithm.

A. Mad-y Algorithm

Turn model representation is an effective way to describe routing algorithm and its restrictions. Figure 3a shows turn model representation of Mad-y routing algorithm. It imposes following constraints on routing turns in order to avoid deadlocks:

- 1) It prohibits four 90-degree turns (E-N1, E-S1, N2-W and S2-W) as shown in Figures 3a(a) and 3a(b).
- 2) It prohibits two 0-degree turns (S2-S1 and N2-N1) as shown in Figure 3a(d).
- 3) It prohibits all 180-degree turns as it is a minimal routing algorithm.

We can deduce following restrictions from above constraints.

- 1) A packet is allowed to take 90-degree turns (N1-E and S1-E) only when it has not already routed to east. These routing turns are restricted only because a packet cannot use N1 or S1 after using east channel.
- 2) A packet is allowed to take 90-degree turns (W-N2 and W-S2) only when it does not need to take west turn further. These routing turns are restricted only because a packet cannot use west channel after using N2 or S2.
- 3) A packet is allowed to take 0-degree turns (N1-N2 and S1-S2) only when it does not need to take west turn further. These routing turns are restricted only because a packet cannot take west turn after using N2 or S2.
- 4) A packet is allowed to take 0-degree turns (N1-N2 and S1-S2) only when it has not already routed to east. These routing turns are restricted only because a packet cannot use N1 or S1 after using east channel.
- 5) If a packet needs to route west, it cannot use N2 or S2 at source node.

Mad-y routing algorithm is proved deadlock free on the basis of work of Dally and Seitz [14], who show that a routing algorithm is deadlock free if channels of network can be assigned numbers such that the algorithm routes each packet along channels with strictly increasing (or decreasing) numbers. A two-digit number (a, b) is assigned to each output channel of a router in $n \times m$ 2D mesh network as shown in Figure 2.

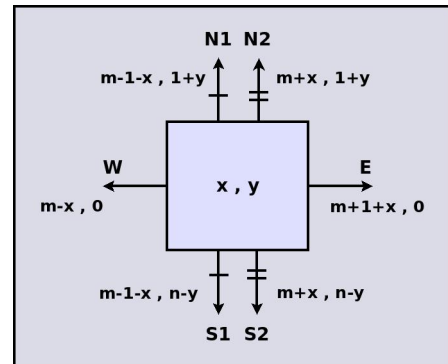


Figure 2: Numbering of the output channels leaving each router (x, y) of $n \times m$ mesh for Mad-y algorithm

B. Congestion Adaptive Routing Method (CARM)

An acyclic channel dependency graph requirement for deadlock avoidance imposes unnecessary restrictions on routing turns in a routing algorithm. Mad-y routing method is minimal and proved deadlock free using acyclic channel dependency graph. Thus, it cannot fully utilize all eligible turns to route packets through less congested regions. CARM imposes substantially fewer restrictions on routing turns, thus becomes more adaptive. Figure 3b shows turn

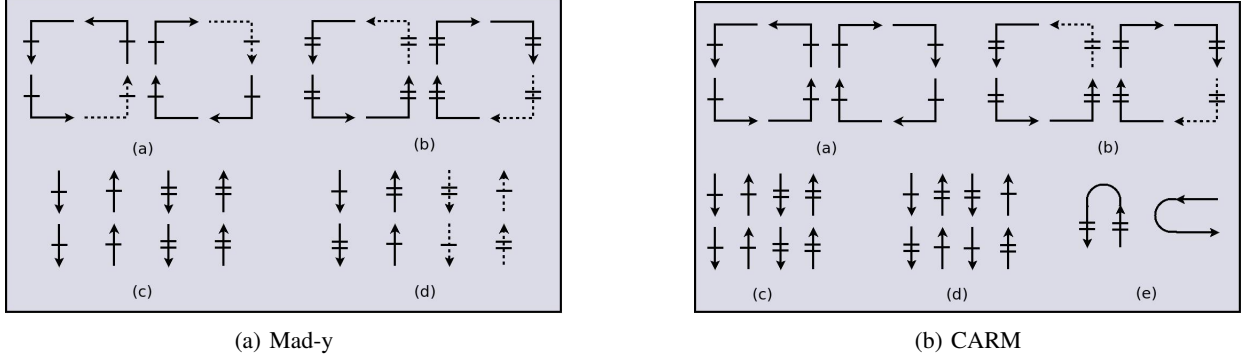


Figure 3: Turn models (solid lines for permitted turns and dash lines for prohibited turns)

model representation of CARM. It imposes following constraints on routing turns in order to avoid deadlocks:

- 1) It prohibits two 90-degree turns (N2-W and S2-W) as shown in Figure 3b(b).
- 2) Although CARM allows all 0-degree turns (N1-N2, S1-S2, N2-N1 and S2-S1), but it restricts them. It allows these restricted turns only when packet does not need to be forwarded further west.
- 3) It permits some 180-degree turns as shown in Figure 3b(e).

We can deduce following restrictions from above constraints.

- 1) A packet is allowed to take 90-degree turns (W-S2 and W-N2) only when it does not need to take west turn further. This restriction is because of prohibited 90-degree turns (N2-W and S2-W).
- 2) If a packet needs to route west, it cannot use N2 or S2 at source node.

Functionality of CARM is divided into two phases: route computation and output channel selection. Routing function computes a set of output channels based on the input channel (on which packet has arrived) and relative position of current and destination nodes. Table II shows the options of eligible output channels permitted by CARM routing function. The table contents are listed based on the relative position of destination and an input channel on which packet has arrived. Similarly, Table I shows the choices of eligible output channels permitted by Mad-y algorithm. We can see that CARM offers a high degree of adaptiveness to route packets as compared to Mad-y algorithm.

Selection function selects an output channel from set of channels computed by the routing function. CARM selection function first inspects eligible output channels corresponding to minimal paths and routes the packet along output channel in which corresponding neighbor router has its congestion flag set to zero (and possibly, which belongs to *non-escape channels*). If the congestion flags of all neighboring routers corresponding to minimal paths are set to one, the congestion status of each eligible non-minimal path is checked. If

there exist such non-minimal channels that are not congested, CARM selects one of them as an output channel to route the packet (and possibly, which belongs to *non-escape channels*). CARM gives preference to adaptive output channels over output channels which are used to escape from deadlocks, because it increases the probability of escape output channels being available when they are needed to escape from deadlock. The selection among adaptive output channels is done by using the strategy described above.

Table I: Eligible output channels for a packet according to the input channel and its destination for Mad-y

	N	S	E	W	NE	NW	SE	SW
N1	-	S1, S2	E	W	-	-	S1, S2, E	S1, W
N2	-	S2	E	-	-	-	S2, E	-
S1	N1, N2	-	E	W	N1, N2, E	N1, W	-	-
S2	N2	-	E	-	N2, E	-	-	-
E	N1, N2	S1, S2	-	W	-	N1, W	-	S1, W
W	N2	S2	E	-	N2, E	-	S2, E	-
L	N1, N2	S1, S2	E	W	N1, N2, E	N1, W	S1, S2, E	S1, W

C. Deadlock and Livelock Freedom of CARM

With deterministic routing, packets have a single output channel choice at each router. Thus, it is necessary to eliminate all cyclic dependencies between channels to avoid deadlocks. In the case of adaptive routing, packets often have several choices at each router. Thus, it is not necessary to remove all cyclic dependencies between channels, provided that every packet can always find a path towards its destination whose channels are not involved in cyclic dependencies. The channels of these acyclic paths are considered as *escape channels* from cycles.

Deadlock-freedom of CARM method can be proved by using Duato's theorem [15] stated as follows.

Theorem 1: (Duato's Theorem) For a given interconnection network I , a connected and adaptive routing function R is deadlock free if there exists a routing subfunction $R_1 \subseteq R$, that is connected and has acyclic extended channel dependency graph.

Table II: Eligible output channels for a packet according to the input channel and its destination for CARM

	N	S	E	W	NE	NW	SE	SW
N1	-	S1, S2	E, S1, S2	W	E, S1, S2	-	S1, S2, E	S1, W
N2	-	S1, S2	E, S1, S2	-	E, S1, S2	-	S1, S2, E	-
S1	N1, N2	S1, S2, N1, N2	E, N1, N2, S1, S2	W	N1, N2, E, S1, S2	N1, W	S1, S2, E, N1, N2	-
S2	N1, N2	S1, S2, N1, N2	E, N1, N2, S1, S2	-	N1, N2, E, S1, S2	-	S1, S2, E, N1, N2	-
E	N1, N2, W	S1, S2, W, N1, N2	E, S1, S2, N1, N2, W	W	N1, N2, E, S1, S2, W	N1, W	S1, S2, E, N1, N2	S1, W
W	N1, N2	S1, S2, N1, N2	E, S1, S2, N1, N2	-	N1, N2, E, S1, S2	-	S1, S2, E, N1, N2, W	-
L	N1, N2	S1, S2, N1, N2	E, S1, S2, N1, N2	W	N1, N2, E, S1, S2	N1, W	S1, S2, E, N1, N2	S1, W

Following Duato’s terminology, the routing function of CARM is denoted by R and the set of channels used by R is denoted by C . To prove deadlock freedom of CARM, we first identify the subset of channels $C_1 \subseteq C$, that defines a connected routing subfunction $R_1 \subseteq R$ and has an extended channel dependency graph (ECDG) with no cycles arising from direct, direct-cross, indirect and indirect-cross dependencies. For CARM, C_1 has all virtual channels except N1 and S1.

Lemma 1: The routing subfunction R_1 is connected.

Proof: R_1 routing function with channel set C_1 is non-minimal version of west-first routing algorithm. Since non-minimal west-first routing is connected, so R_1 is connected. ■

Lemma 2: Extended channel dependency graph of C_1 with additional channel introduced by R (N1 and S1), does not have any cyclic dependencies.

Proof: There is no direct-cross dependency in ECDG of C_1 as routing function R does not add any new routing capability between channels of C_1 directly. Although, routing function R adds new routing capability between channels of C_1 indirectly, but it causes no indirect-cross dependency. Additional channels of R can cause only indirect dependencies between west channels as a packet can use west channel and later can use west channel of different row and column. But this indirect dependency does not introduce any cycle in ECDG of C_1 . The ECDG for C_1 has no dependencies from a channel in the north, east or south directions to a channel in the west direction, so the west channels are always used before all other channels in C_1 . Hence, these indirect dependencies introduce new dependencies between only the west virtual channels and create no cycles using only the west virtual channels. Therefore, the ECDG of C_1 is acyclic. ■

Theorem 2: CARM routing algorithm is deadlock free.

Proof: From Lemma 1 & Lemma 2 and using Theorem 1, CARM routing algorithm is deadlock free. ■

Non-minimal routing algorithms are susceptible to livelock. CARM is proved livelock free using following theorem.

Theorem 3: CARM routing algorithm is livelock free.

Proof: From Tab. II, we can see that whenever a packet is routed in the east direction, it is not allowed

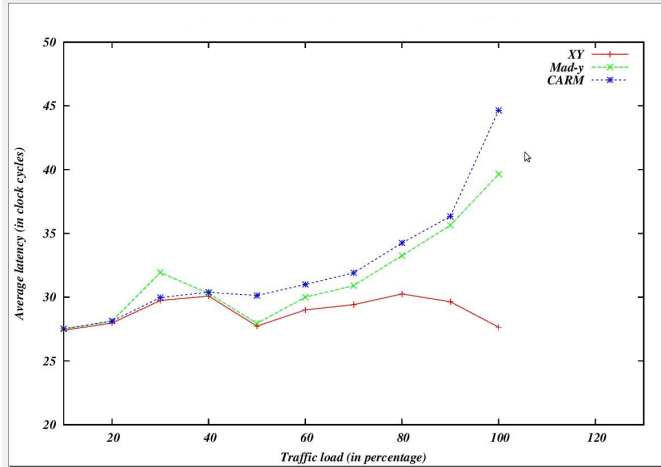
to route it back in the west direction. Therefore, in the worst case, packet may reach to the west most column then starts moving to destination column. In each column, only one 180-degree (N-S) turn is allowed. Therefore, a packet may reach at most top of the column and then starts to move toward destination. Therefore, after a limited number of hops, the packet reaches to its destination node. Thus, CARM routing algorithm is livelock free. ■

IV. EXPERIMENTAL SETUP AND RESULT ANALYSIS

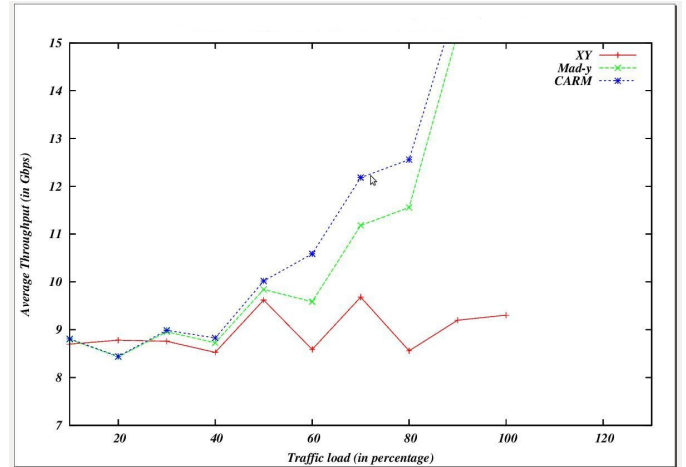
In this section, we evaluate CARM using a cycle accurate simulator NIRGAM [16], [17] (NoC Interconnect Routing and Application Modeling) developed using systemC for on-chip networks. All experiments are carried out using 7×7 mesh for wormhole switching. In all simulations, input port virtual channel size and packet size are set to 6 and 8 flits respectively. Congestion threshold value is set to 60% of input port virtual channel buffer size. Each simulation is executed for 20000 cycles with 16000 traffic generation cycles and 6000 network warm-up cycles. As performance metrics, we use communication latency, throughput and power consumption. Latency is defined as number of clock cycles between arrival and departure of a packet through a router. Throughput is defined as the amount of information delivered by the network per time unit. Latency and throughput are computed per channel per packet basis. We compare CARM with XY and Mad-y routing algorithms for uniform and hot spot traffic models.

A. Latency and Throughput Analysis

1) *Uniform Traffic Model:* With uniform traffic pattern, each NoC node generates packets according to a specific packet injection rate and sends them to every other node in the network with equal probability. Figures 4a and 4b show average latency and throughput per channel under uniform traffic. We can see that at low traffic loads, all algorithms perform similarly. But with increased packet injection rate at high loads, it is observed that the XY and Mad-y routing methods outperform CARM as expected. Since, CARM uses non-minimal paths to alleviate congestion, latencies are higher as compared to XY and Mad-y.



(a) Average Latency



(b) Average Throughput

Figure 4: Performance characteristics per channel per packet under uniform traffic

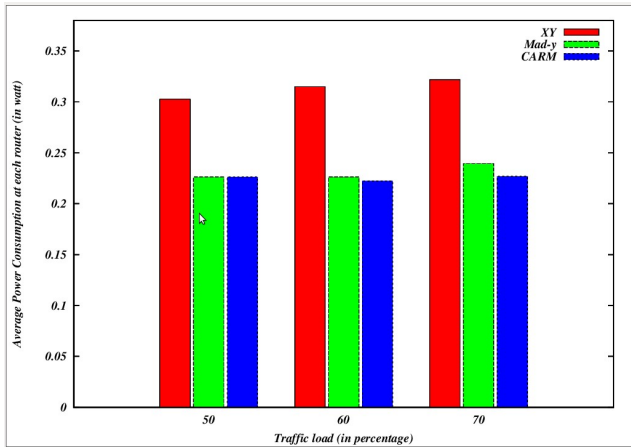


Figure 5: Power consumption results under hot spot traffic

2) *Hot Spot Traffic Model*: Hot spot traffic model is considered as a more realistic traffic pattern, in which a few hot spot nodes receive extra packets in addition to the regular uniform traffic. We set node 10 as hot spot with 0.4 probability of getting additional traffic. Figures 6a and 6b show average latency and throughput per channel under hot spot traffic. It can be observed that CARM method achieves better performance as compared to other schemes. Because of higher adaptiveness and congestion awareness of CARM, it is able to route around local congestion. The experimental results show that the non-minimal method along with congestion awareness can distribute the traffic efficiently.

B. Power Analysis

We deploy an existing NoC power estimation tool ORION [18], which is integrated with NIRGAM. It esti-

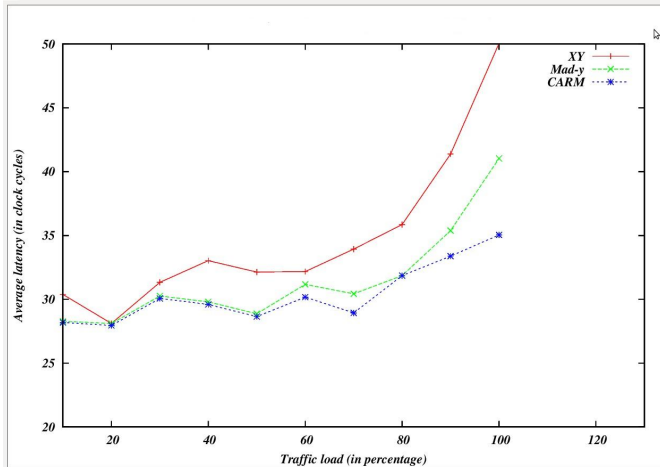
mates total power consumption of a router into various sub-components: input buffers, router control logic including arbiter and crossbar traversal and channels. Figure 5 illustrates average power consumption for hotspot traffic with different traffic loads. It can be observed that CARM consumes less power than other two routing methods due to exploitation of adaptiveness and distribution of traffic uniformly within network.

V. CONCLUSIONS

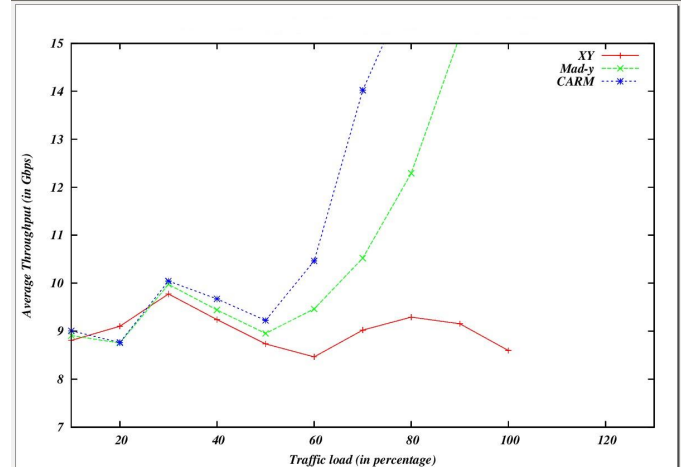
Acyclic channel dependency graph requirement for deadlock avoidance imposes unnecessary restrictions on routing turns, thus reduces degree of adaptiveness. At the same time, inappropriate selection of output channel may result into hot spots in the network causing congestion. In this paper, we have proposed a routing method, CARM to address aforesaid issues for two dimensional meshes. CARM allows cyclic dependencies in channel dependency graph providing higher degree of adaptiveness and still remains deadlock free. It uses congestion-aware channel selection policy that results into balanced distribution of traffic under hot spot traffic pattern. On the basis of simulation results, we argue that deadlock avoidance methodology adopted by CARM is also cost-efficient because it uses only one extra virtual channel along Y dimension to achieve deadlock freedom. Our future work is focused on incorporating global congestion awareness with addition hardware and extending proposed method for n-dimensional meshes.

ACKNOWLEDGMENT

This research is partially supported by Canadian Bureau for International Education, Canada under Canadian Commonwealth Scholarship Program and Ministry of Human Resource Development, India under Institute Assistantship, and is gratefully acknowledged. This work is also supported



(a) Average Latency



(b) Average Throughput

Figure 6: Performance characteristics per channel per packet under hot spot traffic

by UK India Education and Research Initiative grant for the collaborative project on HiPER NIRGAM (2011-2014).

REFERENCES

- [1] L. Benini and G. De Micheli, "Networks on chips: A new soc paradigm," 2002.
- [2] C. Glass and L. Ni, "The turn model for adaptive routing," in *Proceedings of 19th International Symposium on Computer Architecture*, pp. 278–287, 1992.
- [3] J. Hu and R. Marculescu, "DyAD: smart routing for networks-on-chip," in *Proceedings of 41st Design Automation Conference*, pp. 260–263, 2004.
- [4] Z. Zhang, A. Greiner, and S. Taktak, "A reconfigurable routing algorithm for a fault-tolerant 2d-mesh network-on-chip," in *Proceedings of 45th Design Automation Conference*, pp. 441–446, 2008.
- [5] B. Fu, Y. Han, J. Ma, H. Li, and X. Li, "An abacus turn model for time/space-efficient reconfigurable routing," in *Proceedings of 38th International Symposium on Computer Architecture*, pp. 259–270, 2011.
- [6] D. H. Linder and J. C. Harden, "An adaptive and fault tolerant wormhole routing strategy for k-ary n-cubes," *Computers, IEEE Transactions on*, vol. 40, no. 1, pp. 2–12, 1991.
- [7] A. Chien and J. Kim, "Planar-adaptive routing: Low-cost adaptive networks for multiprocessors," in *Proceedings of 19th International Symposium on Computer Architecture*, pp. 268–277, 1992.
- [8] C. J. Glass and L. M. Ni, "Maximally fully adaptive routing in 2d meshes," in *International Conference on Parallel Processing, volume 1*, pp. 101–104, 1992.
- [9] M. Li, Q.-A. Zeng, and W.-B. Jone, "DyXY - a proximity congestion-aware deadlock-free dynamic routing method for network on chip," in *Proceedings of 43rd Design Automation Conference*, pp. 849–852, 2006.
- [10] P. Gratz, B. Grot, and S. Keckler, "Regional congestion awareness for load balance in networks-on-chip," in *Proceedings of 14th International Symposium on High Performance Computer Architecture*, pp. 203–214, 2008.
- [11] R. Ramanujam and B. Lin, "Destination-based adaptive routing on 2D mesh networks," in *Proceedings of 6th ACM/IEEE Symposium on Architectures for Networking and Communications Systems*, pp. 1–12, 2010.
- [12] S. Ma, N. Jerger, and Z. Wang, "DBAR: An efficient routing algorithm to support multiple concurrent applications in networks-on-chip," in *Proceedings of 38th International Symposium on Computer Architecture*, pp. 413–424, 2011.
- [13] M. Ebrahimi, M. Daneshtalab, P. Liljeberg, J. Plosila, and H. Tenhunen, "CATRA- congestion aware trapezoid-based routing algorithm for on-chip networks," in *Proceedings of 15th Design, Automation and Test in Europe Conference Exhibition*, pp. 320–325, 2012.
- [14] W. J. Dally and C. L. Seitz, "Deadlock-free message routing in multiprocessor interconnection networks," *Computers, IEEE Transactions on*, vol. 100, no. 5, pp. 547–553, 1987.
- [15] J. Duato, "A necessary and sufficient condition for deadlock-free adaptive routing in wormhole networks," *Parallel and Distributed Systems, IEEE Transactions on*, vol. 6, no. 10, pp. 1055–1067, 1995.
- [16] L. Jain, B. Al-Hashimi, M. S. Gaur, V. Laxmi, and A. Narayanan, "Nirgam: A systemc based cycle accurate noc simulator," 2010.
- [17] "Nirgam:," <http://wiki.mnit.ac.in/mediawiki/index.php/Nirgam/>.
- [18] A. Kahng, B. Li, L.-S. Peh, and K. Samadi, "ORION 2.0: A fast and accurate noc power and area model for early-stage design space exploration," in *Proceedings of 12th Design, Automation and Test in Europe Conference Exhibition*, pp. 423–428, 2009.