A Novel Highly Adaptive Routing for Networks-on-Chip

M. Kumar, M. S. Gaur, V. Laxmi, M. Daneshtalab, M. Zwolinski and S. Ko

The degree of adaptiveness has a major impact on the performance of an adaptive routing method. This research work presents a novel turn model based routing method that provides a high degree of adaptiveness for 2D mesh. The result is that the proposed method reduces restrictions on the routing turns significantly and hence can provide path diversity using additional routes (both minimal and non-minimal). Experimental results show that the proposed method provides better performance (average latency and throughput) in comparison with the recent routing methods.

Introduction and Background: The Networks-on-Chip (NoCs) model has become a viable communication paradigm for the SoCs, instead of dedicated wires or traditional shared buses. The overall NoC performance depends on many parameters such as topology, task-mapping, flow control mechanism, switching method and routing algorithm. In all cases, route computation function (one phase of routing algorithm) has major and strong effect on the performance of an adaptive routing method [1] and is the focal area of our research work.

In [2], the authors introduced a maximally adaptive double-y (Mad-y) routing algorithm that adds certain routing turns within existing virtual channels (VCs) to increase adaptivity. Minimal routing algorithms guarantee the shortest route between the source and destination, but it would be imprudent to neglect the performance improvements achievable by non-minimal routing. The non-minimal route can be a good (or the only) choice if the minimal routes are congested (or faulty). The degree of adaptiveness provided by the minimal routing algorithms is also low, even if they accurately detect the state of congestion. Ebrahimi *et al*. [3,4] proposed non-minimal routing schemes for a 2D mesh. These provide better adaptivity than [2] with the same VCs. The deadlock-freedom of these algorithms is proved using Dally's work [5]. In [6–8], authors presented non-minimal highly adaptive routing method to increase the adaptivity of route computation function. However, these non-minimal routing algorithms impose some unnecessary restrictions on routing turns, which could be removed to achieve high degree of adaptiveness. In this paper, we present a novel turn model based highly adaptive routing method with congestion awareness (CHARM) for a 2D mesh with wormhole flow control. CHARM offers a high degree of adaptiveness by allowing certain routing turns (prohibited in previous works [2–4, 6, 7]), thus improves network performance.

Proposed Work (CHARM): The motivation of the proposed routing algorithm is derived from the fact that a less restrictive routing algorithm offers a high degree of adaptiveness [1]. The routing algorithms [2–4] achieve deadlock-freedom by forcibly restricting certain routing turns so that the channel dependency graph (CDG) remains acyclic. This acyclic CDG requirement for the deadlock-freedom makes these algorithms more restrictive, thus reduces the degree of adaptiveness. The main focus of this research is to relax this requirement by allowing cycles in the CDG provided that Extended-CDG (ECDG) is acyclic (using DuatoâĂŹs theorem [9]). Since, the proposed turn model (CHARM) imposes fewer constraints on routing turns, it can provide a high degree of adaptiveness. We have explained our point by comparing CHARM with two recent algorithms LEAR [3] and HARAQ [4] (Table 1).

Table 1: Prohibited routing turns for different routing algorithms

Turns	Mad-y [2]	LEAR [3]	HARAQ [4]	CHARM
90-degree	E-N1, E-S1, S2-W, N2-W	E-N1, E-S1, S2-W, N2-W	E-N1, E-S1, S2-W, N2-W	S2-W, N2-W
0-degree	N2-N1, S2-S1	N2-N1, S2-S1	N2-N1, S2-S1	-
180-degree	ALL	ALL except N1-S2, S1-N2	ALL except W-E, N1-S2, S1-N2, S1-N1, S2-N2	ALL except W-E, S1-N1, S1-N2, S2-N1, S2-N2

Figure 1 shows the turn model representation of CHARM. For minimal routing, a packet is permitted to use the first VC ($N1$ and $S1$) at any time, as shown in Figure 1(i). It can use the second VC ($N2$ or $S2$) only if it has already been routed in the west, as shown in Figure 1(ii). All the prohibited routing turns for CHARM are shown in the Table 1. In contrast to the other related work [2–4], CHARM forbids only two 90-degree turns. CHARM permits all 0-degree turns. However, It should be noted that some 0-degree turns ($N2$-$N1$, $S2$-$S1$, $N1$-$N2$ and $S1$-$S2$) are permitted only if the destination is not in the west. The route computation

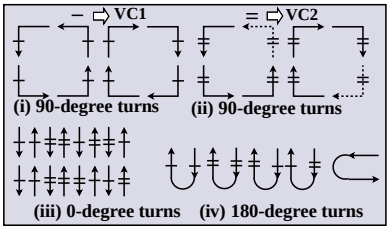

Fig. 1: CHARM Turn models (permitted (prohibited) turns are represented by solid (dash) lines)

function (*rfun*) of CHARM is described in the Table 2. The highlighted entries are part of the CHARM and discussed later in Theorem 3. The *rfun* produces output-channel vector for a packet using the packet's destination position (*des_pos*) and the packet's input channel (*i_ch*). The selection function (*sel*) of the CHARM selects one channel from output-channel vector. The *sel* first examines all the output channels on minimal paths and selects one of the output channels of minimal path in which the corresponding congestion flag (*cflag*) is set to zero. The CHARM routing algorithm prefers non-minimal routes if the minimal routes are congested (*cflag* set to 1). If the congestion flags of all the minimal paths are set to one, the non-minimal paths are checked. If there exist such non-minimal paths and the congestion flags are set to zero, *sel* picks one of the output channel on the non-minimal path to route the packet.

Table 2: route computation function (*rfun*) for CHARM

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

Deadlock and Livelock Freedom

The deadlock-freedom of CHARM is assured from Duato's theorem [9], stated as follows:

Theorem 1: (Duato's Theorem) For an 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 an acyclic ECDG (with no cycles because of direct, indirect, direct-cross and indirect-cross dependencies).

Following Duato's terminology, we represent the *rfun* of CHARM by R. The set C represents the channels used by R and contains all the VCs ($N1$, $S1$, $N2$, $S2$, E and W). To assure the deadlock-freedom of CHARM, we first identify the subset of channels $C_1 \subseteq C$ (escape channels), which defines routing subfunction $R_1 \subseteq R$. For CHARM, this subset C_1 contains the channels $N2$, $S2$, E and W. Table 3 describes the *rfun* for the routing subfunction R_1 of CHARM.

Table 3: *rfun* for routing subfunction R_1

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

Lemma 1: The routing subfunction R_1 is connected and cycle-free.

Proof: The *rfun* for routing subfunction R_1 (Table 3) with the channel set C_1 is a non-minimal version of west-first routing [1], thus connected and cycle-free. ■

Lemma 2: The ECDG of the channel set C_1 does not have any cycle because of direct-cross and indirect-cross dependencies.

Proof: We can observe from the Tables 2 and 3 that the routing subfunction R_1 is defined using a channel subset C_1, according to the following expression:

$$R_1(i\_ch, des\_pos) = R(i\_ch, des\_pos) \cap C_1, \forall i\_ch, des\_pos \quad (1)$$

It means whenever the routing function R (Table 2) provides a channel from the set C_1 ($N2$, $S2$, E and W) for a particular destination, that channel is also provided by the routing function R_1 (Table 3) for the same destination. The cross dependencies may exist if we add any routing option between channels of C_1 while developing routing function R from R_1 by adding channels $N1$ and $S1$. We have not added any routing option

between channels of C_1. Thus, there does not exist any cross-dependency between channels in C_1 meaning ECDG is cycle-free because of cross dependencies. ■

Lemma 3: The ECDG of the channel set C_1 does not have any cycle because of direct and indirect dependencies.

Proof: From Lemma 1, R_1 is proved cycle-free. Thus no direct dependency can cause cycles in ECDG. Since, the R_1 is west-first routing algorithm, a west channel is always utilized before any other channel (south or north or east) in C_1. Thus, the dependencies towards west channel from any other channel (south or north or east) are absent. However, the additional channels ($N1$ and $S1$) introduced by R can cause indirect dependencies between west channels as a packet can use west channel, then any addition channel ($N1$ or $S1$) and later can use west channel of different row. But this indirect dependency does not introduce any cycle in ECDG. Because to form a cycle, at least one of the 90-degree turns ($S2$-W or $N2$-W) must be allowed. However these 90-degree turns are prohibited. Thus, these indirect dependencies introduce new dependencies between only the west VCs and do not result in cycles. Since there are no direct and indirect dependencies that produce cycle in ECDG of C_1. Therefore ECDG of C_1 is acyclic because of direct and indirect dependencies. ■

Theorem 2: The proposed routing algorithm is deadlock-free.

Proof: We can conclude from Lemmas (1, 2 and 3) and using Theorem 1 that the proposed routing algorithm is deadlock-free. ■

Theorem 3: The proposed routing algorithm is livelock-free.

Proof: The key idea behind livelock-freedom of CHARM is that only one 180-degree turn is allowed in each dimension. From Lemma 1, It is proved that R_1 routing is a non-minimal version of west-first routing [1], thus it is livelock-free. We design a new routing function (R_2) by splitting the north VC ($N2$) into two VCs ($N1$ and $N2$) and south VC ($S2$) into $S1$ and $S2$. We impose same routing constraints on both newly added VCs $N1$ and $N2$ as of old $N2$. Similarly, newly added VCs $S1$ and $S2$ are also having same routing constraints as of old $S2$. Table 4 describes the routing restriction of the new routing function R_2.

Table 4: *rfun* for the new routing function R_2 derived from R_1

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

We can observe that R_2 is also non-minimal west-first routing algorithm. The only difference is that R_2 uses a double-y network, whereas R_1 uses single VC in each dimension. Thus, we can conclude that R_2 is also livelock-free. It is well known that minimal routing never causes livelock and if we add minimal paths to the R_2, it will remain livelock-free. We can observe that the *rfun* of CHARM (Table 2) is result of addition of some minimal paths entries in the routing function R_2 (Table 4). We have shown these entries as highlighted text in the Table 2. These entries are corresponding to minimal paths, thus never cause livelock. Thus, we can conclude that CHARM is livelock-free. ■

Performance Evaluation: We have evaluated the CHARM with real (E3S) [10] and synthetic traffic (hotspot) profiles. To evaluate the effectiveness of CHARM, we have implemented four other routing turn models, named XY, Mad-y [2], LEAR [3] and HARAQ [4] using in-house systemC based simulator. We have considered a 7×7 mesh for performing all the experiments. The simulator is run for 10000 cycles to discount any start-up transients and the average performance is measured over another 100000 cycles. The packet size and input-channel buffer size for each VC are set to 8 and 6 flits, respectively with congestion threshold at 66% of the total buffer size. We use average latency, average throughput and power as the performance metrics. Figures 2(a) and 2(b) show average latency and average throughput for hotspot traffic. CHARM outperforms other algorithms as it can more evenly distribute traffic in a congested network using additional paths (both minimal and non-minimal) than other routing algorithms. Fig. 3(a) depicts normalized average latency to XY method. CHARM outperforms all other related work for all benchmark applications. The performance improvement of CHARM is 29% and 14% when compared with XY and other adaptive methods, respectively for a 7×7 mesh. Figure 3(b) illustrates the average power consumption for hotspot traffic with different traffic loads using ORION [11]. Deterministic XY consumes less power for all traffic loads because it always routes packets through minimal paths. At lower traffic loads, CHARM performs slightly better than other adaptive routing methods as it reduces the hop count for packets using additional minimal paths than others. However, at higher traffic loads, the performance of all adaptive methods is almost similar as they follow non-minimal paths frequently.

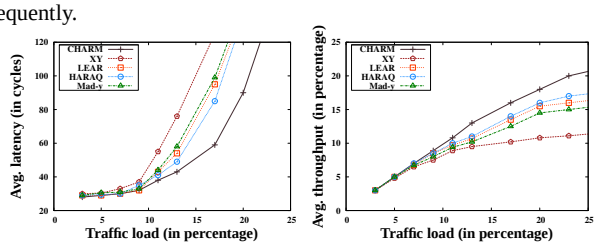

Fig. 2: Average latency and average throughput for hotspot traffic

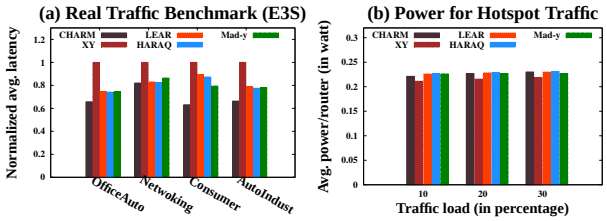

Fig. 3: Performance for real traffic benchmark (E3S) and average power per router for hotspot traffic

Conclusion: Acyclic CDG for deadlock-freedom prohibits several routing turns, thus reduces degree of adaptiveness. This work presents a highly adaptive routing CHARM to improve the performance of a 2D mesh NoC. The proposed routing provides a higher degree of adaptiveness by allowing cycles in the CDG while remaining deadlock-free. The deadlock-freedom of CHARM is ensured using Duato's Theorem. CHARM uses additional minimal and non-minimal paths than other previous algorithms to distributes traffic around the "hot-spot" regions of the network.

M. Kumar, M. S. Gaur, V. Laxmi (*MNIT, Jaipur, India*)
E-mail: 2011rcp7126@mnit.ac.in
M. Daneshtalab (*University of Turku, Finland*)
M. Zwolinski (*University of Southampton, UK*)
S. Ko (*University of Saskatchewan, Canada*)

References

1 Glass C. J. and Ni L. M.: 'The Turn Model for Adaptive Routing', in *ACM SIGARCH Computer Architecture News*, vol. 20, no. 2. ACM, 1992, pp. 278–287.
2 Glass C. J. and Ni L. M. : 'Maximally Fully Adaptive Routing in 2D Meshes', in *International Conference on Parallel Processing, volume I*, 1992, pp. 101–104.
3 Ebrahimi M. *et al.*: 'LEAR – a Low-Weight and Highly Adaptive Routing Method for Distributing Congestions in On-Chip Networks', in *Proceedings of 20th Euromicro International Conference on Parallel, Distributed and Network-Based Processing*, 2012, pp. 520–524.
4 Ebrahimi M. *et al.*: 'HARAQ: Congestion-Aware Learning Model for Highly Adaptive Routing Algorithm in On-Chip Networks', in *Proceedings of 6th International Symposium on Networks on Chip*, May 2012, pp. 19–26.
5 Dally W. J. and Seitz C. L.: 'Deadlock-free Message Routing in Multiprocessor Interconnection Networks', *Computers, IEEE Transactions on*, vol. 100, no. 5, pp. 547–553, 1987.
6 Kumar M. *et al.*: 'A Novel Non-minimal/minimal Turn Model for Highly Adaptive Routing in 2D NoCs', in *Proceedings of 8th International Symposium on Networks-on-Chip*, 2014, pp. 184–185.
7 Kumar M. *et al.*: 'A Novel Non-minimal Turn Model for Highly Adaptive Routing in 2D NoCs', in *Proceedings of 22nd International Conference on Very Large Scale Integration*, Oct 2014, pp. 1–6.
8 Kumar M. *et al.*:'Highly Adaptive and Congestion-aware Routing for 3D NoCs,' in *Proceedings of the 24th Edition of the Great Lakes Symposium on VLSI*, 2014, pp. 97–98.
9 Duato J.: '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.
10 E3S: http://ziyang.eecs.umich.edu/ dickrp/e3s/.
11 Kahng A.*et al.*: '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*, 2009, pp. 423–428.