Local Congestion Avoidance in Network-on-Chip

Minghua Tang, Xiaola Lin, and Maurizio Palesi, *Member, IEEE*

Abstract—Network-on-Chip (NoC) has been made the communication infrastructure for many-core architecture. NoC are subject to congestion, which is claimed to be avoided by many researchers. However, there is no completely understanding of congestion in literature, which hinders its solution. Toward this direction, we firstly carry out study on congestion in this paper. We find that congestion usually occurs at a portion of nodes in a local network region. Moreover, local congestion will significantly decrease system performance and mostly impact some particular communication pairs. Then we attempt to solve local congestion by addressing different local region size, based on Divide-Conquer approach and routing pressure. It avoids congestion in every local region by keeping routing pressure of every local region minimum. Using different local region size will create different routings. Our study shows that the local region size is closely related with the routing performance. When local region size is 5×5 the optimal routing performance of large size network could be achieved.

Index Terms—Network-on-chip, routing algorithm, divide-conquer, routing pressure, local congestion

◆

1 INTRODUCTION

INTEGRATION of a large number of processing cores into a single chip (SoC) have been enabled by the semiconductor technology improvement. In order to cope with the SoC communication requirements, Network-on-Chip (NoC) has been proposed to substitute the traditional bus-based interconnects [1], [2], [3]. NoC architectural parameters includes not only architecture parameters (i.e., network topology and links) but communication parameters (i.e., switching and routing strategies) as well.

After network topology is selected, routing algorithm is the important element that determines the network performance. For the most widely studied mesh topology [4], [5], a huge number of NoC routing algorithms have been presented in literature [6], such as dimension order routing (DOR) [4], turn model [7], [8], Odd-Even (OE) turn model [9], abacus [10], Segment-Based Routing (SR) [11] APSRA [12], RABC [13], DyAD [14], etc.

One important objective of designing NoC is to make network traffic uniform. However, network traffic is usually non-uniform. Essentially, traffic non-uniformity could be caused by a lot of factors, such as topological artifacts, routing biases, traffic scenario, long range dependence, etc [15], [16], [17], [18].

Due to traffic non-uniformity, some network nodes have to dispatch much more packets than others. Congestion will take place at those nodes and significantly affects network performance. Many researchers have proposed all kinds of strategies to overcome the negative impact of congestion in the following four aspects. First, local network status is used to design routing algorithm to avoid congestion [19], [20], [21], [22]. Second, global network status is used to design routing algorithm to avoid congestion [23], [24], [25], [26], [27], [28]. Third, end-to-end flow control strategy is proposed to reduce injecting packets into network in the case of congestion [29], [30], [31], [32], [33]. Fourth, selection strategies are designed to make it possible for the packets to go around the congested nodes [34], [35], [36].

As the first contribution, we carry out study on congestion in 2D mesh network trough simulation in detail. Four important findings are made from the simulations. First, congestion usually happens at partial nodes in a local network region. It will spread to other network nodes, the speed and range depending on the routing and traffic scenario. Second, some particular communication pairs will be mostly affected by congestion and always have the largest packet delay. Moreover, the affected short path communication pairs always have the largest packet delay. Third, the few affected communication pairs greatly contribute to the global average packet delay (APD). The average packet delay will sharply decrease if those pairs are excluded. Fourth, congestion will cause that under some routings and traffic, the network throughput will firstly increase and then decrease after the maximum.

As the second contribution, we attempt to solve local congestion using Divide-Conquer [37] approach and routing pressure [38] techniques. Our further study shows that varying the size of local region will bring out different routing algorithm performance. The local region size has significant impact on the routing algorithm performance. As square region varies from 3×3 to 4×4, to 5×5, the created routing performance increases. Using 5×5 region size could create routing algorithm that has the globally optimal routing performance.

2 RELATED WORK

Although a large variety of NoC topologies have been presented in the literature, 2D mesh is the most widely

---

- *M. Tang is with the Department of Computer Science and Technology, GuangDong University of Finance, Guangzhou 510521, China. E-mail: fractal218@126.com.*
- *X. Lin is with the School of Information Science and Technology, Sun Yat-sen University, Guangzhou 510275, China, and the Key Laboratory of Digital Life (Sun Yat-sen University), Ministry of Education, Guangzhou 510275, China. E-mail: linxl@mail.sysu.edu.cn.*
- *M. Palesi is with the Kore University, Kore, Italy. E-mail: maurizio.palesi@unikore.it.*

Manuscript received 25 Dec. 2014; revised 27 July 2015; accepted 23 Aug. 2015. Date of publication 27 Aug. 2015; date of current version 15 June 2016. Recommended for acceptance by G. Wang.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TPDS.2015.2474375

studied [4]. A huge number of routing algorithms are proposed for 2D mesh topology to overcome congestion and improve performance. In 2D mesh topology, some paths have to be prohibited in order to avoid deadlock. Routing algorithms could be classified into deterministic routing in which each communication pair has only one path and adaptive routing in which more than one path is allowable for the communication pairs. If all paths provided by the topology are allowable the routing becomes fully adaptive.

We choose the following representative routing algorithms to carry out our study. These routings include deterministic, adaptive and fully adaptive routings. The typical deterministic routing is XY routing, in which packets are first transferred horizontally and then vertically. Glass and Ni [7], [8] propose the turn model which leads to adaptive routing. According to their turn model, three routing algorithms could be gotten: negative-first (NF), north-last (NL), and west-first (WF). Based on it, OE turn model [9] is proposed, in which different paths are prohibited to improve performance. SR [11] is another adaptive routing. In SR routing, bidirectional routing restrictions are positioned on the partitioned segments independently and then communication paths for all source-destination pairs are computed.

Although these routing algorithms have made some progress in overcoming the negative impact of congestion, and therefore improve the network performance, they do not carry out study on congestion itself. Consequently, they could not show how their performance improvement comes. To this end, we firstly carry out study on congestion and find that it usually happens in a local region. Then basing on this discovery, we propose new routing algorithm designing method and the resulted routing algorithm could significantly improve network performance.

In adaptive routing, the packets may have more than one path to reach their destinations. In this case, a proper selection strategy is needed to choose the appropriate path for packets to reach their destinations as fast as possible. A lot of selection strategies are available in literature, such as random, buffer-level, NoP [35], CAIS [34]. The Neighbors-on-Path (NoP) selection chooses the freest path to forward a packet. When a router R has two choices for a coming packet: forwarding it to neighbor R_1 or R_2. With the NoP strategy, it grades R_1 and R_2 by free buffer space of all possible downstream neighbor nodes, respectively. Then the packet will be forwarded to the neighbor that has higher grade. The contention-aware input selection (CAIS) selection forwards the packet that will mostly decrease the network traffic load. Within a router, if two inputs channels c_1 and c_2 request the same output channel c_3. The access will be granted to the input channel that has higher contention level. The CAIS selection could not ensure whether the granted packet could reach its destination as soon as possible. Consequently, the NoP selection is chosen in this paper.

3 LOCAL CONGESTION

Network congestion could significantly affect network performance. In this section, we study network congestion through a detailed simulation in terms of four different points of view: buffer occupancy, the largest packet delay, congestion impact on average packet delay and throughput drop.

TABLE 1
Simulation Setup

Simulator	Noxim [39]
Topology	Mesh-based
Network size	7×7
Port buffer	Four flits
Switch technique	Wormhole switching
Routing algorithm	XY, NF, NL, WF, OE and SR
Arbitration	Round-Robin
Selection strategy	Random and NoP
Traffic scenario	Uniform, Transpose1, Transpose2, Hs-c and Hs-tr
Packet size	Eight flits
Traffic distribution	Poisson
Virtual channels	No

When congestion occurs the buffer space of the related nodes will be occupied by the waiting packets. Therefore, through buffer occupancy we can know where congestion happens. Through the largest packet delay and congestion impact on average packet delay we could know which communication pairs are mostly affected by congestion and analyze whether the congestion happen in a local region or in global range. Through throughput drop we show the impact of local congestion.

3.1 Simulation Configuration

In an $M \times N$ 2D mesh NoC, each node is considered as a point in a 2D coordinate system. The origin of the coordinate system is at the top left corner of the mesh. The X axis is the horizontal direction and Y axis is the vertical direction. The positive direction of X axis points to the east, while the positive direction of Y axis points to the south.

Each node is identified by its coordinate (x, y) with $0 \leq x < M$ and $0 \leq y < N$. Each node also has an identifier (ID) computed by the formula: $ID = y * M + x$.

In this section, we choose six representative routing algorithms to study congestion: XY, NF, NL, WF, OE and SR, including deterministic, adaptive and fully adaptive routings. The simulation setup is summarized in Table 1. The network payload traffic is regulated by the packet injection rate (PIR). For example, each node under PIR of 0.1 injects a packet into network every 10 cycles, on average. The simulation runs 50,000 cycles after 1,000 cycles of warm-up period.

We consider uniform, transpose1 and transpose2 and hotspot traffic scenarios. In uniform traffic, each node randomly generates packets to every other node with the same probability. In transpose1 traffic, node (i, j) only sends packets to node $(N-1-j, N-1-i)$. In transpose2 traffic, node (i, j) only generates packets to node (j, i). In the hotspot traffic, several nodes are identified as *hotspot* nodes. If h is the hotspot percentage, each node sends a portion of h percent packets to the hotspot nodes. The remaining packets are randomly sent to other nodes in the network. We assume h is 0.2 in our simulations. In the first hotspot traffic (hs-c), the four hotspot nodes are 24, 25, 31, 32. In the second hotspot traffic (hs-tr), the four hotspot nodes are 5, 6, 12, 13. Poisson distribution that is suitable to describe the

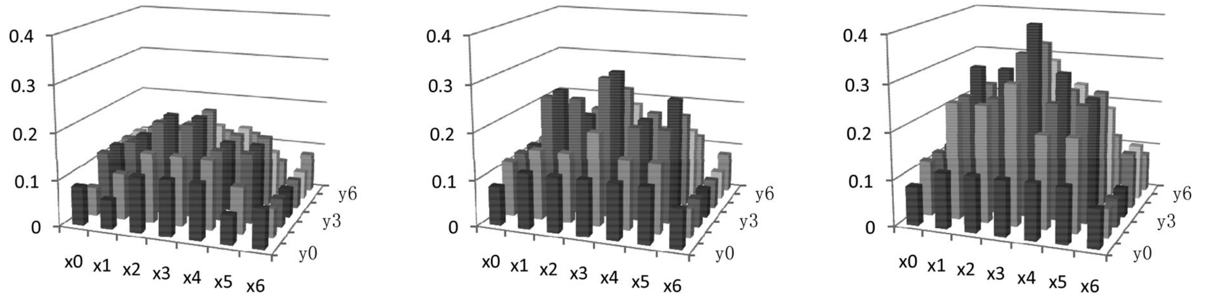


Fig. 1. Buffer occupancy under OE routing and uniform traffic.

occurrence times of random event in unit time could describe lots of random event. It is chosen to describe network traffic distribution in this paper.

We consider the *average packet delay* and *throughput* as performance metrics, which are defined as follows:

$$Average\ packet\ delay = \frac{1}{K}\sum_{i=1}^{K} lat_i,$$

where K refers to the total number of received packets and lat_i is the delay of the ith packet.

$$throughput = \frac{total\ received\ flits}{(number\ of\ nodes) * (total\ cycles)},$$

where *total received flits* represents the total number of flits received by all the destinations, *number of nodes* is the number of network nodes, and *total cycles* is the simulation length in clock cycles.

3.2 Buffer Occupancy

During packets going to their destination nodes, they will be forwarded by a group of network nodes. The packet will temporary stay at a node and occupy its buffer space until it could be accepted by the next hop node. Congestion occurs when more and more packets have to wait in the network. The waiting packets should occupy the buffer space of the network nodes. Consequently, nodes buffer occupancy is highly related with congestion. According to [33], congestion occurs at a network node when more than a quarter of its buffer space has been occupied.

Routing algorithm is an important factor among the factors which make the network traffic non-uniform. In this section, we study congestions that was made by routings through observing network nodes buffer occupancy. We try to observe where will the congestion appear and how will it spread in the network.

We study congestion for each routing under uniform, transpose1, transpose2, hs-c and hs-tr traffic scenarios. Under each traffic, we carry out three simulations. Firstly, we carry out simulation under random selection and small PIR to observe where will the congestion occur. Then, we carry out the second simulation under random selection and large PIR to observe how will congestion spread in the network. At last, the third simulation is run under NoP selection and small PIR to observe how will the congestion change. In these simulations, we calculate the buffer occupancy of each node.

Fig. 1 depicts the results for OE routing and uniform traffic. As shown in Fig. 1b, congestion appears at the central network nodes. When PIR is increased congestion spreads along the congested nodes as shown in Fig. 1c. NoP selection could remove congestion in this case, Fig. 1a.

Fig. 2 shows the results for OE routing and transpose1 traffic. As shown in Fig. 2b, congestion happens at those nodes in a region which X coordinates range from 2 to 5 and Y coordinates range from 0 to 3. Congestion gets more serious when PIR is increased as shown in Fig. 2c. NoP selection takes great effect and nearly remove congestion in this case, Fig. 2a.

The results for OE routing and transpose2 traffic are depicted in Fig. 3. Fig. 3b shows that congestion occurs at the region nodes which X coordinates range from 2 to 5 and Y coordinates range from 3 to 6. Congestion becomes more serious under large PIR, Fig. 3c. Congestion is nearly removed under NoP selection, shown in Fig. 3a.

Fig. 4 shows the results for OE routing and hs-c traffic. Fig. 4b shows that congestion occurs at those nodes which X coordinate is 3 and Y coordinates range from 0 to 4. Fig. 4c shows that congestion becomes more serious and spreads to some nodes which Y coordinates range from 0 to 1 when

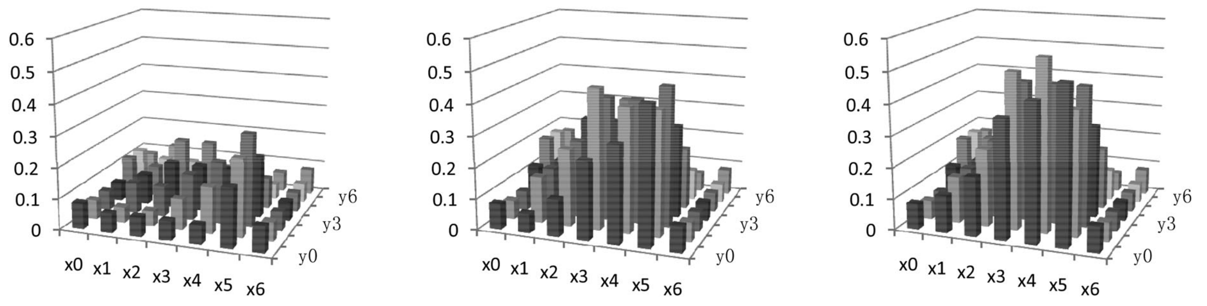


Fig. 2. Buffer occupancy under OE routing and transpose1 traffic.

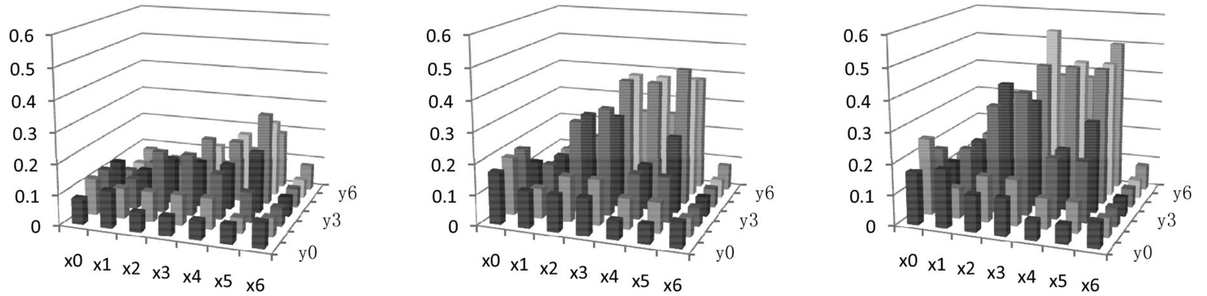


Fig. 3. Buffer occupancy under OE routing and transpose2 traffic.

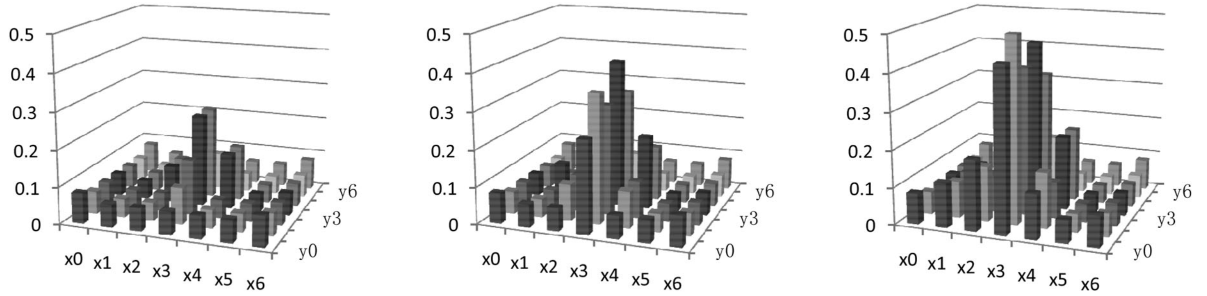


Fig. 4. Buffer occupancy under OE routing and hs-c traffic.

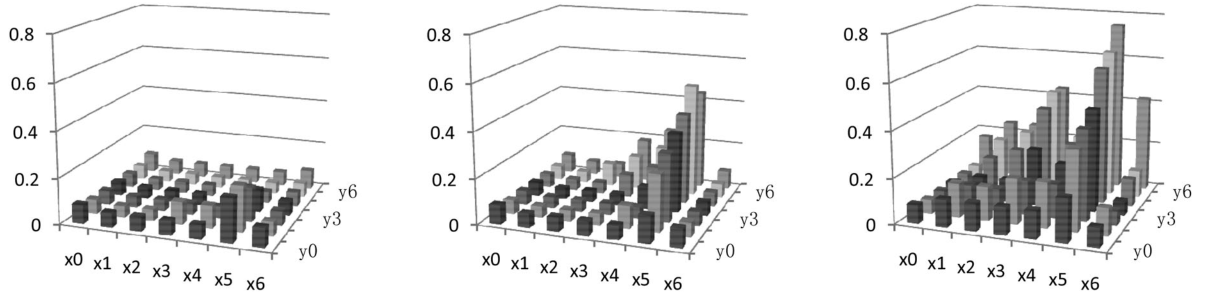


Fig. 5. Buffer occupancy under OE routing and hs-tr traffic.

PIR is increased. Fig. 4a shows that NoP selection could significantly decrease congestion.

Fig. 5 depicts the results for OE routing and hs-tr traffic. Congestion happens at nodes which X coordinate is 5 as shown in Fig. 5b. When PIR is increased congestion spreads to some nodes which Y coordinates range from 4 to 6 as shown in Fig. 5c. NoP selection could remove congestion in this case, Fig. 5a.

The simulation results for XY, NF, NL, WF and SR routings are summarized in Table 2. Under each routing and traffic scenario, the number of congested nodes is depicted. As shown in Table 2, only a partial of network nodes are congested. When PIR is increased, more and more nodes may be congested.

In summary, from this section, we can get the following observations:

1) The network is not completely congested. On the contrary, congestion usually happens at partial nodes in a local region.
2) Congestion could spread to other non-congested nodes if PIR is increased. The spreading speed depends on the routing and traffic pattern.
3) Selection strategy may have important impact on the congestion status.

3.3 The Largest Packet Delay

Usually, the long path packet should have large packet delay. However, this trend will be changed by the congestion. When congestion occurs some packets should have to wait in the network or in the source buffer. The waiting packets will have large packet delay than usual.

In this section, we carry out simulations to study which communication pairs will be mostly impacted by congestion. In transpose1 and traspose2 traffics, the communication pairs that will pass through a region are fixed. Then the communication pairs impacted by local congestion are also fixed. However, in the other three traffics, uniform, hs-c and hs-tr, there are many communication pairs which could pass through a region. All of them will be affected by the local congestion. The affected communication pairs are not fixed. Consequently, we only take transpose1 and transpose2 traffics in this section.

In this section, we conduct 200 simulations under each traffic with random selection and NoP selection, respectively. After each simulation, we record the communication pair that has the largest packet delay. After the 200 simulations, we sum the total times for each communication pairs. Then we show the dominate communication pairs.

Fig. 6 shows the results for OE routing and transpose1 traffic. Three communication pairs (10,26), (3,27) and

TABLE 2
The Number of Congested Nodes Under Various Routings and Traffic Scenarios

Routing	Traffic	NoP	Small PIR	Large PIR
XY	Uniform	13	13	33
	Transpose1	11	11	12
	Transpose2	10	10	12
	hs-c	16	16	21
	hs-tr	7	7	22
NF	Uniform	18	27	41
	Transpose1	11	12	12
	Transpose2	21	22	25
	hs-c	5	15	16
	hs-tr	3	3	12
NL	Uniform	16	18	35
	Transpose1	5	5	9
	Transpose2	5	6	6
	hs-c	4	4	17
	hs-tr	0	13	23
WF	Uniform	2	15	26
	Transpose1	6	6	6
	Transpose2	5	6	6
	hs-c	5	5	15
	hs-tr	2	2	16
SR	Uniform	1	18	26
	Transpose1	6	15	19
	Transpose2	4	11	18
	hs-c	1	4	6
	hs-tr	5	5	27

TABLE 3
The Communication Pairs Under Various Routings and Traffic Scenarios

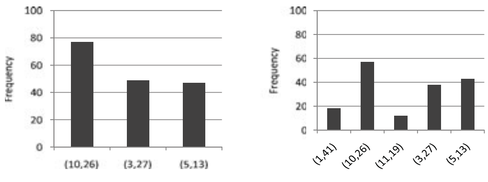
Routing	Traffic	Communication pairs
XY	Transpose1	(0,48)12,(1,41)10,(47,7)10,(48,0)12
	Transpose2	(42,6)12,(43,13)10,(5,35)10,(6,42)12
NF	Transpose1	(0,48)12,(47,7)10,(48,0)12,(7,47)10
	Transpose2	(11,29)6,(12,36)8,(18,30)4,(19,37)6, $29,11)6,(30,18)4,(36,12)8,(37,19)6
NL	Transpose1	(47,7)10,(48,0)12
	Transpose2	(42,6)12,(43,13)10
WF	Transpose1	(47,7)10,(48,0)12
	Transpose2	(5,35)10,(6,42)12
OE	Transpose1	(10,26)4,(3,27)6,(5,13)2
	Transpose2	(38,26)4,(45,27)6,(47,41)2
SR	Transpose1	(21,45)6,(22,38)4,(35,43)2
	Transpose2	(18,30)4,(19,37)6,(25,31)2,(26,38)4

(5,13) are those exhibiting the largest packet delay as shown in Fig. 6a. As it can be noticed, even if such communication pairs use relatively short paths (of five, six, and two hops, respectively), they exhibit the highest packet delay. The use of NoP selection, Fig. 6b, significantly change this trend. In nearly 20 simulations, communication pair (1,41) generates the largest delay packets. The path of communication pair (1,41) is among the longest in transpose1 traffic (10 hops).

The simulation results for XY, NF, NL, WF, OE and SR routings are summarized in Table 3. Under each routing and traffic, the communication pairs that have the largest packet delay and the path lengths in terms of hops are depicted.

The following observations could be made from this section:

1) If a routing is deterministic, then long path communication pairs have the largest delay packets and NoP selection does not change this trend.
2) If a routing is fully adaptive, then short path communication pairs in the central network have the largest packets and NoP selection does not change this trend.
3) If a routing is partially adaptive, then short path communication pairs that was affected by local congestion have the largest packets. Moreover, NoP selection could take significant effect to change this trend.



(a) Random selection (b) NoP selection

Fig. 6. Frequency of max delay communication pairs under OE routing and transpose1 traffic.

3.4 Congestion Impact on Average Packet Delay

In this section, we study how congestion impacts average packet delay. We consider two delay metrics, namely, average packet delay and excluded average packet delay (EAPD). The first one is the average packet delay among all the communicating pairs. The second one is the average packet delay among all the communicating pairs excluding those which exhibit the highest delay. For transpose1 and transpose2 traffics, the excluded communication pairs are shown in the previous section. For uniform, hs-c and hs-tr traffics, since the communication pairs that have the largest delay are not fixed, we calculate EAPD excluding the largest 10 and 5 percent delay packet, respectively. Finally the decrease ratio of EAPD over APD is computed. The ratio is calculated three times according to large PIR, small PIR and NoP selection respectively.

The results for XY routing are listed in Table 4. For uniform traffic, excluding the top 10 percent (5 percent) communication pairs the average packet delay decrease by 11 percent (6 percent). Increasing PIR does not significantly increase the ratios. Congestion does not significantly affect these communication pairs. However, For transpose1 and transpose2 traffics, excluding the top four communication pairs could decrease 46 and 38 percent average packet delay. Increasing PIR sharply increase the ratios. The four pairs are seriously affected by congestion. Similarly, congestion affects partial communication pairs in hs-c and hs-tr traffics. NoP selection does not take obvious effect for XY routing.

Table 5 depicts the ratios for NF routing. Since for transpose2 traffic NF routing is fully adaptive, excluding the top eight pairs only has trivial effect. Increasing PIR or using NoP selection do not change this trend. For other traffics, partial communication pairs are seriously affected by congestion. Excluding those pairs will significantly decrease

TABLE 4
Average Packet Delay Decrease Ratio for XY Routing

Traffic	Excluded pairs	Large PIR	Small PIR
Uniform	10%	13%	11%
	5%	8%	6%
Transpose1	48 ↔ 0 47 → 7 1 → 41	83%	46%
Transpose2	6 ↔ 42 5 → 35 43 → 13	78%	38%
hs-c	10%	75%	62%
	5%	55%	42%
hs-tr	10%	61%	53%
	5%	45%	30%

TABLE 5
Average Packet Delay Decrease Ratio for NF Routing

Traffic	Excluded pairs	Large PIR	Small PIR	NoP sel.
Uniform	10%	46%	20%	15%
	5%	33%	12%	9%
Transpose1	48 ↔ 0 47 ↔ 7	69%	49%	36%
Transpose2	11 ↔ 29 12 ↔ 36 18 ↔ 30 19 ↔ 37	7%	6%	6%
hs-c	10%	71%	40%	37%
	5%	58%	27%	23%
hs-tr	10%	36%	25%	24%
	5%	18%	15%	12%

TABLE 6
Average Packet Delay Decrease Ratio for NL Routing

Traffic	Excluded pairs	Large PIR	Small PIR	NoP sel.
Uniform	10%	19%	16%	13%
	5%	12%	9%	8%
Transpose1	48 → 047 → 7	75%	45%	27%
Transpose2	42 ↔ 6 43 ↔ 13	57%	38%	20%
hs-c	10%	55%	25%	21%
	5%	48%	11%	11%
hs-tr	10%	61%	52%	40%
	5%	48%	35%	21%

average packet delay. Increasing PIR makes congestion serious and increases the ratios. On the contrary, NoP could decrease congestion status and the ratios.

Table 6 shows the ratios for NL routing. For all traffics, local congestion affect some partial communication pairs. Excluding those pairs could significantly decrease average packet delay. Increasing PIR makes congestion sharply become serious. Thus the partial pairs are affected more. NoP could take significant effect.

Table 7 shows the ratios for WF routing. The decrease ratio for uniform traffic is low, which means that congestion does not affect the top pairs significantly. Increasing PIR or using NoP selection does not change this trend. For other four traffics, the excluded pairs are significantly affected by congestion. Increasing PIR makes the excluded pairs more seriously affected. On the contrary, NoP selection only has marginal effect.

Table 8 depicts the ratios for OE routing. The excluded pairs greatly contribute the average packet delay. Increasing PIR would significantly increase their contribution. NoP selection could sharply decrease the packet delay of the excluded pairs and their contribution.

TABLE 7
Average Packet Delay Decrease Ratio for WF Routing

Traffic	Excluded pairs	Large PIR	Small PIR	NoP sel.
Uniform	10%	17%	13%	11%
	5%	10%	7%	7%
Transpose1	48 → 047 → 7	61%	53%	41%
Transpose2	6 → 42 5 → 35	72%	51%	41%
hs-c	10%	51%	28%	25%
	5%	46%	17%	16%
hs-tr	10%	26%	23%	22%
	5%	13%	12%	11%

TABLE 8
Average Packet Delay Decrease Ratio for OE Routing

Traffic	Excluded pairs	Large PIR	Small PIR	NoP sel.
Uniform	10%	11%	10%	8%
	5%	6%	6%	4%
Transpose1	10 → 26 3 → 27 5 → 13	28%	26%	7%
Transpose2	38 → 26 45 → 27 39 → 33	29%	15%	7%
hs-c	10%	42%	39%	20%
	5%	31%	24%	9%
hs-tr	10%	52%	45%	20%
	5%	43%	27%	11%

TABLE 9
Average Packet Delay Decrease Ratio for SR Routing

Traffic	Excluded pairs	Large PIR	Small PIR	NoP sel.
Uniform	10%	36%	16%	8%
	5%	20%	10%	5%
Transpose1	21 → 45 35 → 43 22 → 38	26%	23%	7%
Transpose2	26 → 38 18 → 30 19 → 37 25 → 31	25%	17%	8%
hs-c	10%	54%	28%	19%
	5%	37%	16%	7%
hs-tr	10%	57%	41%	29%
	5%	34%	21%	18%

Table 9 depicts the ratios for SR routing. Local congestion has great impact on partial communication pairs. These pairs are more seriously affected when PIR is increased. With NoP selection, negative impact on partial pairs are greatly reduced.

From this section, several observations could be made:

1) Local congestion has significantly impact on particular communication pairs. Since the packets from these pairs have to pass through the congested region, they will have very large packet delay.
2) As PIR increases, the delay of packets from these affected pairs may sharply increase.

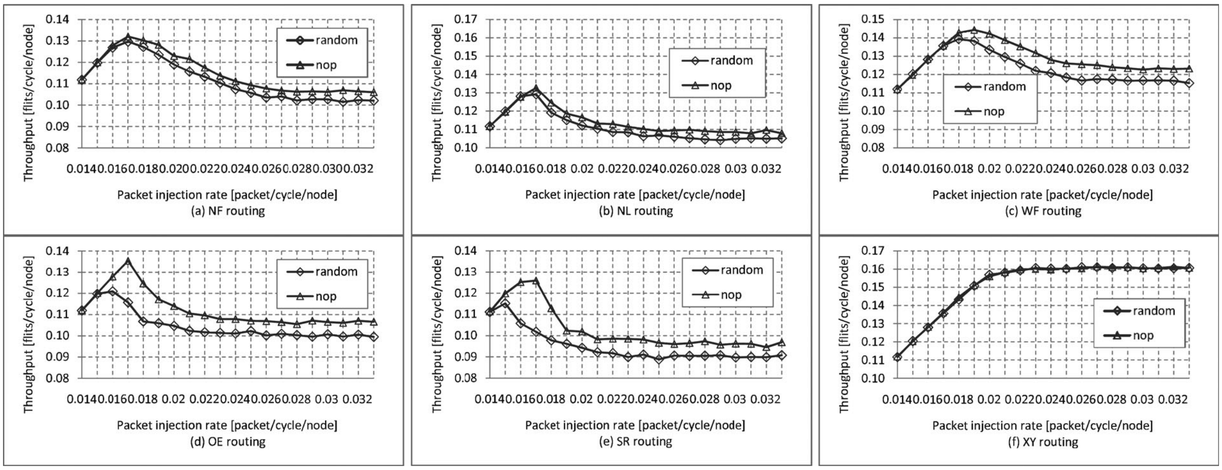


Fig. 7. Throughput variations under (a) NF, (b) NL, (c) WF, (d) (OE), (e) SR and (f) XY routing for uniform traffic.

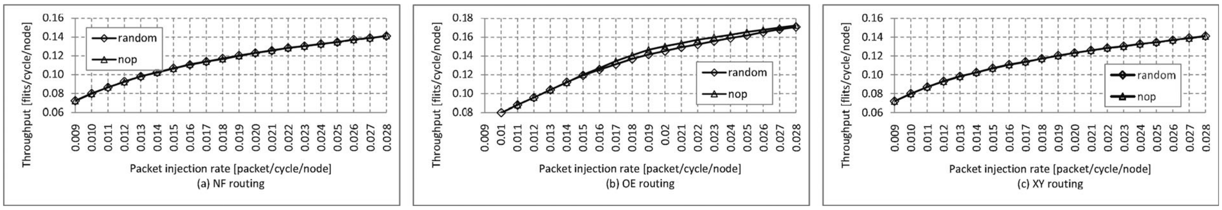


Fig. 8. Throughput variations under (a) NF, (b) OE and (c) XY routing for transpose1 traffic.

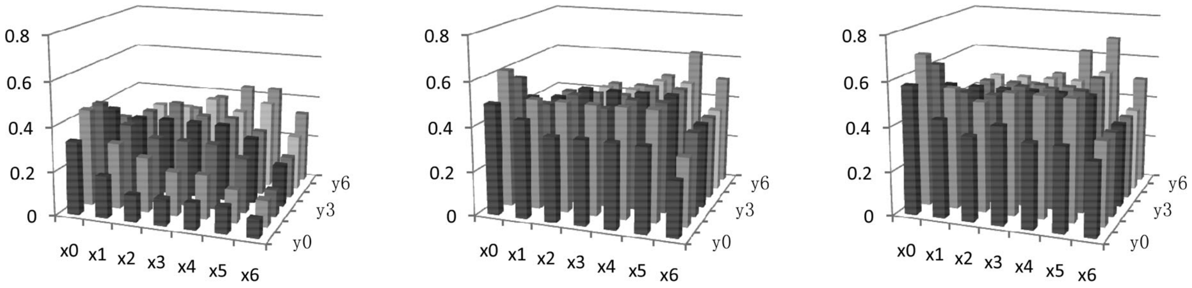


Fig. 9. Buffer occupancy variations under NF routing and uniform traffic.

3) NoP could remarkably make network traffic more uniform to reduce the abnormal large delay of partial communication pairs.

3.5 Explain Network Throughput Drop

As shown in [35], network throughput may drop as PIR increases after reaching the maximum. In this section, we study how throughput drop is related with local congestion. We conduct abundant simulations under all routing and traffic combinations and record the throughput variations.

Fig. 7 shows the results for six different routing algorithms under uniform traffic. With exception of XY routing, the network throughput variations have three phases: rise, drop, keep, as PIR increases. At first, throughput rises until it reaches the top point. Then it drops. At last, it keeps the current level.

Under the other traffic scenarios, the same behavior is observed: throughput keep rising without drop. It will keep rising without drop. Therefore, we only show the results for transpose1 traffic due to space limitation. Fig. 8 depicts the results for NF, OE and XY routings under transpose1 traffic.

The network throughput variations are related with network congestion variations. Three cases are observed:

1) NF, NL, WF, OE and SR routings under uniform traffic belong to the first category. In the first phase, congestion firstly occurs at a local region and the remaining network nodes are not congested, as shown in Fig. 9a. The un-congested nodes could transmit more packets as PIR increases. The network throughput will still rise in this phase. As PIR increases further, the un-congested nodes will gradually get congested. In the second phase, after all network nodes become congested, then the network throughput drops in the second phase as shown in Fig. 9b. In the third phase, the network traffic status becomes stable. The network throughput does not change anymore as shown in Fig. 9c.
2) XY routing under uniform traffic and NF routing under transpose2 traffic belong to the second category. Traffic is quite uniform in this case. When congestion occurs, it quickly spreads to the whole network, as shown in Fig. 10. After all network

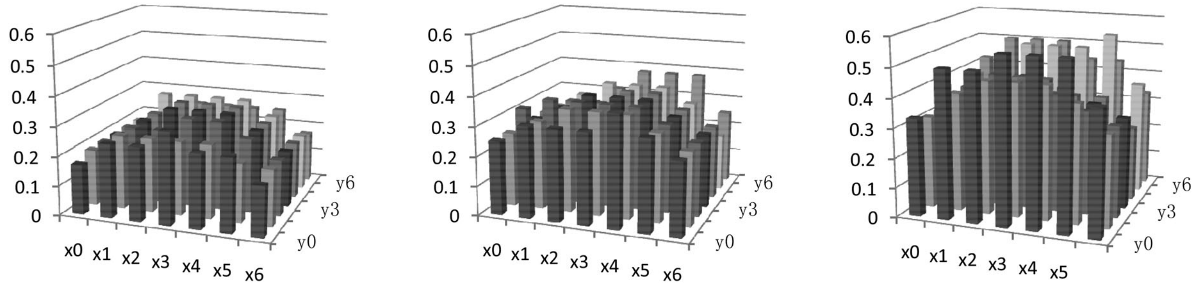


Fig. 10. Buffer occupancy variations under XY routing and uniform traffic.

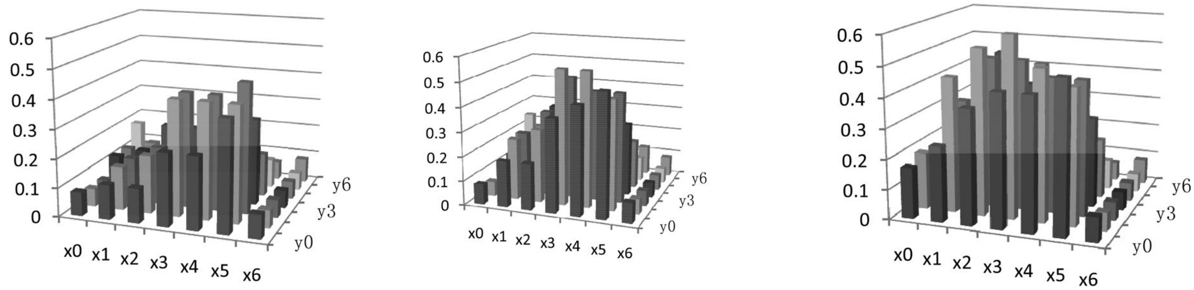


Fig. 11. Buffer occupancy variations under OE routing and transpose1 traffic.

nodes are congested, network throughput keeps at stable level.

3) Other combinations of routing and traffic pattern belong to the third category. Because of traffic pattern and routing, network traffic is very non-uniform. Very little packets could arrive at some partial nodes. These nodes will never be congested as shown in Fig. 11. As PIR increases the buffer occupancies of nodes which X coordinates are **6 increase very slowly. Under these routings and traffics, packets are kept in the source buffer space. Packets that are injected into network slowly rises up, which makes network throughput slowly goes up.

4 SOLVE LOCAL CONGESTION PROBLEM

The task of designing routing algorithm is to find new routing algorithms which have better performance than the existed ones. For a 2×2 mesh network, there are totally 12 routing algorithms as shown in Fig. 12. For a 7×7 mesh network, it contains 36 2×2 sub-networks. The total routing algorithms are 12^{36}. Consequently, new routing algorithms should come from that huge group of candidate routing algorithms.

The routing algorithm designing for 2D mesh topology is so complex that it could not be straightforwardly solved, especially for large size network [37]. Divide-Conquer approach has been proposed to design routing algorithm for 2D mesh network [37]. It divides a large network into two small networks. Then the routing algorithms for the small networks are designed, which is relatively simple. Finally, the routing algorithms for the large network are obtained from merging the routings of the small networks. The divide and conquer steps may recursively proceed several times.

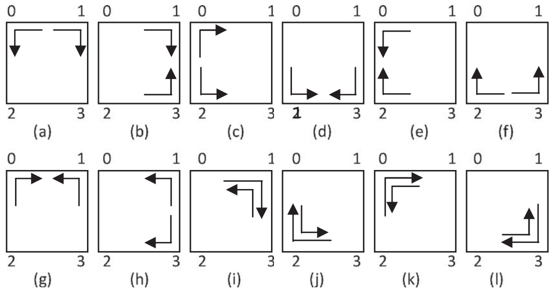


Fig. 12. The 12 2×2 network routings.

As shown in Section 3, congestion always happens in a local network region and greatly affects network performance. However, the size and position of local region are not fixed. In this section, we attempt to solve local congestion problem using Divide-Conquer approach, addressing different local region size. The simulation results show that addressing different local region size could significantly impact routing performance.

4.1 Solve Local Congestion when Designing Routing Algorithm

Fig. 13 depicts a 7×7 network, where a dashed square illustrates a local region. As the square move from left to right and from up to down, it covers every local region in the network.

Because routing performance could be decreased by congestion occurs in any local region, the routing algorithm has

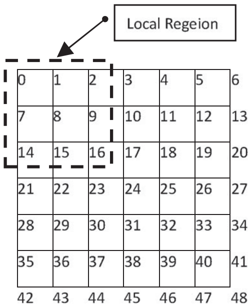


Fig. 13. A 7×7 network.

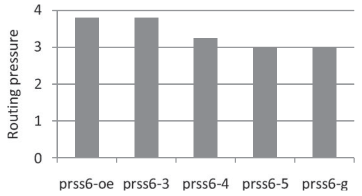


Fig. 14. Routing pressure statistics for 6×6 network.

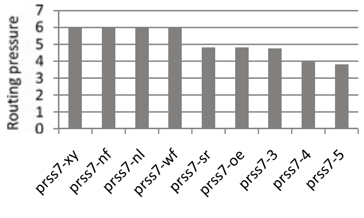


Fig. 15. Routing pressure statistics for 7×7 network.

to ensure that congestion does not happen in every local region. Research [38] shows that in order to avoid congestion routing algorithm should have small routing pressure which is a useful metric to measure the performance of routing algorithms. We refer the readers to paper [38] for the detail of the routing pressure. In this section we briefly introduce its basic idea. Given a routing algorithm and traffic pattern, a portion of traffic will pass through a channel. The traffic passing a channel is called its channel pressure. The maximum channel pressure of all network channels is referred to as the routing pressure of the routing algorithm. Routing algorithm with large routing pressure has poor performance, and vice versa.

In this section, we use divide-conquer approach and routing pressure to solve the local congestion problem while designing routing algorithm. The basic idea is as follows. Because congestion always occurs at nodes in a local network region, if we keep routing pressure in every local region minimum then congestion could be avoided as far as possible. When a routing for the small network is obtained, we check if all routing pressures of its local regions are minimum under both transpose1 and transpose2 traffic. If so, that routing is used to merge the large network routing. If not, that routing is discarded. In that process, we choose transpose1 and transpose2 traffics because we try to filter out non promising solutions to sharply reduce the searching space and make the searching time acceptable.

4.2 Impact of Local Region Size on Routing Pressure

In order to compute routing pressure under transpose1 and transpose2 traffics, the local region should be square. The size of local region has multi choices. For example, in a 7×7 network, the local region may be 3×3, 4×4, 5×5, and so on. Consequently, there are 25, 16, 9 local regions for 3×3, 4×4, 5×5 region size, respectively.

The size of local region has a significant impact on the routing pressure. We take 6×6 network as an example to show the impact of region size. To this end, we respectively design routing algorithms for 6×6 network using 3×3, 4×4 and 5×5 region size. We will ge three group of routings for 6×6 network. In the three group of routings, we compute the routing pressure for every routing algorithm under both transpose1 and transpose2 traffics. In each group, the routing pressure that is smallest under both transpose1 and transpose2 traffics is calculated. The three smallest routing pressures of the three categories are referred to as prss6-3, prss6-4 and prss6-5, respectively.

Then we figure out all 6×6 network routings without any constraint. The smallest routing pressure is the globally smallest since all the routings are computed. It is termed as prss6-g. For reference, the routing pressure of OE routing is also computed. It is termed as prss6-oe.

Fig. 14 depicts the histogram of the five routing pressures. Two important observations could be make from Fig. 14. Firstly, as local region size increases, the routing pressure of obtained routing for 6×6 network decreases. Secondly, the routing pressure under 5×5 region size is the optimal since it equals the globally smallest. It means that it is possible to use 5×5 region size to design large network routing which performance approximates the optimal.

4.3 Simulation Results

In this section, we compute routing algorithms for 7×7 network using 3×3, 4×4 and 5×5 region size, respectively. We will get three groups of routing algorithms. Their routing pressures are computed under both transpose1 and transpose2 traffics. In each group, the routing pressure that is smallest under both transpose1 and transpose2 traffics is selected. The three smallest routing pressures are referred to as prss7-3, prss7-4 and prss7-5, respectively. The routing pressures for XY, NF, NL, WF, SR and OE routings are also computed. They are termed as prss7-xy, prss7-nf, prss7-nl, prss7-wf, prss7-sr and prss7-oe respectively.

Fig. 15 depicts the histogram of the nine routing pressures. Since NF routing is fully adaptive for transpose2 traffic its routing pressure under that traffic is smallest. Except this, using 5×5 region size could leads to the smallest routing pressure under both transpose1 and transpose2 traffics.

Next, we compare the performance of the obtained routing algorithms and the state-of-the-art routing algorithms. The routings created by using 3×3, 4×4 and 5×5 region size are referred to as TMPL3, TMPL4 and TMPL5 respectively. To guarantee the accuracy of simulation results, the simulation at each PIR is repeated a number of times, to guarantee accuracy with a 95 percent confidence interval within 2 percent of the means.

Fig. 16 depicts the average packet delay variations under transpose1 traffic. The routing pressures for XY, NF, NL and WF routings are 6.0 as Fig. 15 shows. The packet delays under these four routings are larger than 200 even when PIR is 0.013. TMPL5 has the best performance under this traffic.

Fig. 17 shows the average packet delay variations under transpose2 traffic. The average packet delays for XY, NL, and WF are so large that they could not be shown in the figure. NF routing has the smallest average packet delay since it is fully adaptive. Except NF routing, TMPL5 has better performance than other routings.

The results for hs-c traffic are depicted in Fig. 18. Under this traffic, TMPL5, NL and WF routings have better performance than other routings.

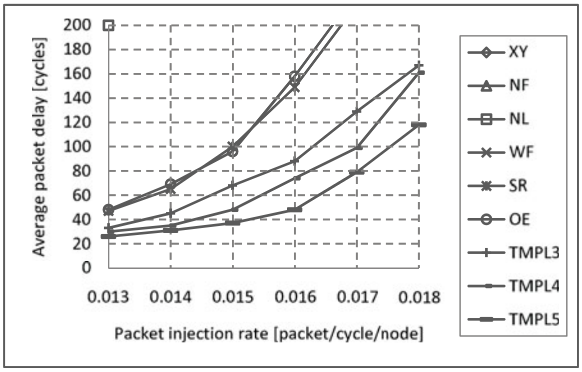


Fig. 16. Delay variations under transpose1 traffic scenario.

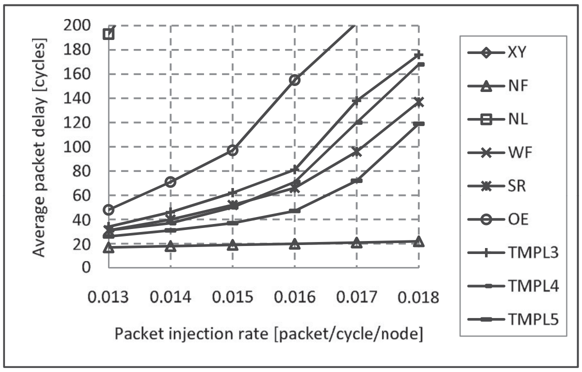


Fig. 17. Delay variations under transpose2 traffic scenario.

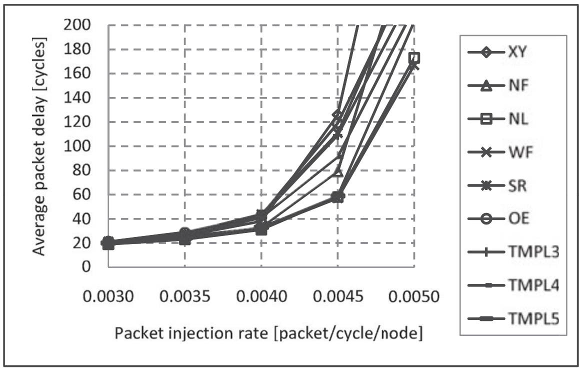


Fig. 18. Delay variations under hs-c traffic scenario.

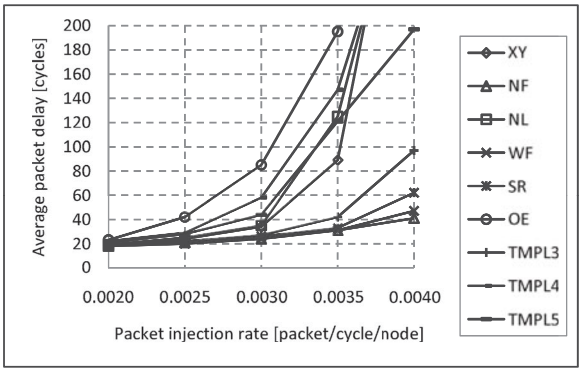


Fig. 19. Delay variations under hs-tr traffic scenario.

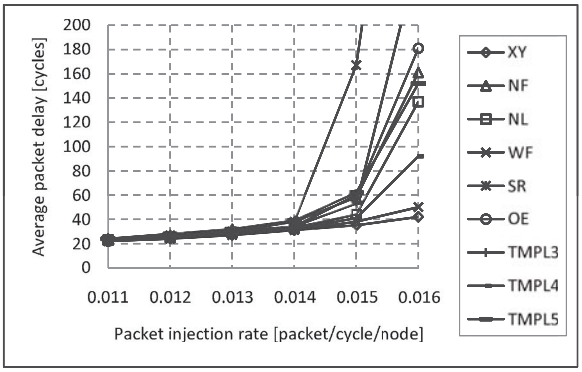


Fig. 20. Delay variations under uniform traffic scenario.

The delay variations for hs-tr traffic are shown in Fig. 19. NL and WF routings have the best performance.

The average packet delay variations for uniform traffic are shown in Fig. 20. XY routing has the best performance under uniform traffic.

5 CONCLUSIONS

In this paper, we carry out detailed study on network congestion. We found that congestion usually happens at partial nodes in a local network region. Some specific communication pairs will be mostly affected by congestion and always have the largest packet delay. The few affected communication pairs greatly contribute to the average packet delay. Local congestion will cause that under some routings and traffic scenarios, the network throughput increases before the maximum, and then drops to the stable level.

A method to overcome local congestion is proposed while designing routing algorithm. It requires that routing pressure in every local region is smaller than a specified minimum. After conducting experiment on different region size we found that 5×5 region size could lead to optimal performance routing algorithm.

As the future study, we will explore the local congestion in other kinds of networks and how to avoid it.

ACKNOWLEDGMENTS

This work was supported in part by the National Natural Science Foundation of China under Grants No. 61472454. X. Lin is the corresponding author.

REFERENCES

[1] H. G. Lee, N. Chang, U. Y. Ogras and R. Marculescu, “On-chip communication architecture exploration: A quantitative evaluation of point-to-point, bus, and network-on-chip approaches,” *ACM Trans. Design Autom. Electron. Syst.*, vol. 12, pp. 1–20, Aug. 2007.

[2] W. J. Dally and B. Towles, “Route packets, not wires: On-chip interconnection networks,” in *Proc. ACM/IEEE Design Autom. Conf.*, 2001, pp. 684–689.

[3] L. Benini and G. D. Micheli, “Networks on chips: A new SoC paradigm,” *IEEE Comput.*, vol. 35, no. 1, pp. 70–78, Jan. 2002.

[4] W. Dally and B. Towles, *Principles and Practices of Interconnection Networks*. San Mateo, CA, USA: Morgan Kaufmann, 2004.

[5] F. Martti, “A scalable high-performance computing solution for networks on chips,” *IEEE Micro,*vol. 22, no. 5, pp. 46–55, Sep. 2002.

[6] M. Palesi and M. Daneshtalab, *Routing Algorithms in Networks-on-Chip*. New York, NY, USA: Springer, 2014.

[7] C. J. Glass and L. M. Ni, "The turn model for adaptive routing," in *Proc. 19th Ann. Int. Symp. Comput. Archit.*, 1992, pp. 278–287.

[8] C. J. Glass and L. M. Ni, "The turn model for adaptive routing," *J. Assoc. Comput. Mach.*, vol. 41, pp. 874–902, 1994.

[9] G.-M. Chiu, "The odd-even turn model for adaptive routing," *IEEE Trans. Parallel Distrib. Syst.*, vol. 11, no. 7, pp. 729–738, Jul. 2000.

[10] B. Z. Fu, Y. H. Han, J. Ma, H. W. Li and X. W. Li, "An abacus turn model for time/space-efficient reconfigurable routing," in *Proc. Int. Symp. Comput. Archit.*, Jun. 2011, pp. 259–270.

[11] A. Mejia, J. Flich, and J. Duato, "On the potentials of segment-based routing for NoCs," in *Proc. 37th Int. Conf. Parallel Process.*, 2008, pp. 594–603.

[12] M. Palesi, R. Holsmark, S. Kumar and V. Catania, "Application specific routing algorithms for networks on chip," *IEEE Trans. Parallel Distrib. Syst.*, vol. 20, no. 3, pp. 316–330, Mar. 2009.

[13] M. H. Tang and X. L. Lin, "Network-on-chip routing algorithms by breaking cycles," in *Proc. 10th Int. Conf. Algorithms Archit. Parallel Process.*, 2010, pp. 163–173.

[14] J. Hu and R. Marculescu, "DyAD smart routing for networks-on-chip," in *Proc. 41st Design Autom. Conf.*, 2004, pp. 260–263.

[15] P. Bogdan and R. Marculescu, "Non-stationary traffic analysis and its implications on multicore platform design," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 30, no. 4, pp. 508–519, Apr. 2011.

[16] M. Bakhouya, A. Chariete, J. Gaber, and M. Wack, "A buffer-space allocation approach for application-specific network-on-Chip," in *Proc. IEEE 10th Int. Conf. Comput. Syst. Appl.*, 2011, pp. 263–267.

[17] A. Scherrer, A. Fraboulet and T. Risset, "Long-range dependence and on-chip processor traffic," *Microprocess. Microsyst.*, vol. 33, pp. 72–80, 2009.

[18] P. Bogdan and R. Marculescu, "Statistical physics approaches for network-on-chip traffic characterization," in *Proc. 7th IEEE/ACM Int. Conf. Hardw./Softw. Codesign Syst. Synthesis*, 2009, pp. 461–470.

[19] M. Valinataj, S. Mohammadi1, J. Plosila, and P. Liljeberg, "A fault-tolerant and congestion-aware routing algorithm for networks-on-chip," in *Proc. IEEE 13th Int. Symp. Design Diagnostics Electron. Circuits Syst.*, 2010, pp. 139–144.

[20] E. Nilsson, M. Millberg, J. Öberg, and A. Jantsch, "Load distribution with the proximity congestion awareness in a network on chip," in *Proc. Design, Autom. Test Eur. Conf. Exhib.*, 2003, pp. 1126–1127.

[21] N. Jiang, D. U. Becker, G. Michelogiannakis, and W. J. Dally, "Network congestion avoidance through speculative reservation," in *Proc. IEEE 18th Int. Symp. High Performance Comput. Archit.*, 2012, pp. 1–12.

[22] E. J. Chang, H. K. Hsin, S. Y. Lin, and A. Y. Wu, "Path-congestion-aware adaptive routing with a contention prediction scheme for network-on-chip systems," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 33, no. 1, pp. 113–126, Jan. 2014.

[23] R. S. Ramanujam and B. Lin, "Destination-based adaptive routing on 2D mesh networks," in *Proc. IEEE Int. Symp. Archit. Netw. Commun. Syst.*, 2010, pp. 1–12.

[24] G. Michelogiannakis, J. Balfour, and W. J. Dally, "Elastic-buffer flow control for on-chip networks," in *Proc. IEEE 15th Int. Symp. High Perform. Comput. Archit.*, 2009, pp. 151–162.

[25] M. Ramakrishna, P. V. Gratz, and A. Sprintson, "GCA: Global congestion awareness for load balance in networks-on-chip," in *Proc. IEEE 7th Int. Symp. Netw. Chip*, 2013, pp. 1–8.

[26] L. Tedesco, T. Rosa, F. Clermidy, N. Calazans, and F. Moraes, "Implementation and evaluation of a congestion aware routing algorithm for networks-on-chip," in *Proc. 23rd Symp. Integr. Circuits Syst. Design*, 2010, pp. 91–96.

[27] X. Chang, M. Ebrahimi, M. Daneshtalab, T. Westerlund, and J. Plosila, "PARSC—An efficient congestion-aware routing method for networks-on-chip," in *Proc. IEEE 16th Int. Symp. Comput. Archit. Digital Syst.*, 2012, pp. 166–171.

[28] P. Gratz, B. Grot, and S. W. Keckler, "Regional congestion awareness for load balance in networks-on-chip," in *Proc. IEEE 14th Int. Symp. High Perform. Comput. Archit.*, 2008, pp. 203–214.

[29] M. S. Talebi, F. Jafari, A. Khonsari, and M. H. Yaghmae "A novel congestion control scheme for elastic flows in network-on-chip based on sum-rate optimization," in *Proc. Int. Conf. Comput. Sci. Its Appl.*, 2007, pp. 398–409.

[30] J.W. v. d. Brand1, C. Ciordas, K. Goossens, and T. Basten, "Congestion-controlled best-effort communication for networks-on-chip," in *Proc. Design, Automation Test Eur. Conf. Exhibition*, 2007 pp. 1–6.

[31] G. Nychis, C. Fallin, T. Moscibroda, and O. Mutlu, "Next generation on-chip networks: What kind of congestion control do we need?" in *Proc. 9th ACM SIGCOMM Workshop Hot Topics Netw.*, 2010, pp. 1–6.

[32] G. Nychis, C. Fallin, T. Moscibroda, O. Mutlu, and S. Seshan, "On-chip networks from a networking perspective: Congestion and scalability in many-core interconnects," in *Proc. ACM SIGCOMM Conf. Appl., Technol., Archit., Protocols Comput. Commun.*, 2012, pp. 407–418.

[33] M. H. Tang and X. L. Lin, "Quarter load threshold (QLT) flow control for wormhole switching in mesh-based network-on-chip," *J. Syst. Archit.*, vol. 56, no. 9, pp. 452–462, 2010.

[34] D. Wu, Bashir, M. Al-Hashimi, and M. T. Schmitz, "Improving routing efficiency for network-on-chip through contention-aware input selection," in *Proc. Asia South Pacific Conf. Design Autom.*, 2006, pp. 36–41.

[35] G. Ascia, V. Catania, and M. Palesi, "Implementation and analysis of a new selection strategy for adaptive routing in networks-on-chip," *IEEE Trans. Comput.*, vol. 57, no. 6, pp. 809–820, Jun. 2008.

[36] M. Ebrahimi, M. Daneshtalab, P. Liljeberg, J. Plosila, and H. Tenhunen, "Agent-based on-chip network using efficient selection method," in *Proc. IEEE/IFIP 19th Int. Symp. VLSI Syst.-on-Chip*, 2011, pp. 284–289.

[37] M. H. Tang and C. H. Wu,"A new method of designing NoC routing algorithm," in *Proc. 2nd Int. Conf. Consumer Electron., Commun. Netw.*, 2012, pp. 3044–3047.

[38] M. H. Tang, X. L. Lin, and M. Palesi, "Routing pressure: A channel-related and traffic-aware metric of routing algorithm," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 3, pp. 891–901, Nov. 2013.

[39] Sourceforge.net. (2008). Noxim: Network-on-chip simulator. [Online]. Available: http://noxim.sourceforge.net

Minghua Tang received the BS degree in applied math from Chongqing University, Chongqing, China, in 1999, and the MS degree in applied math from Chongqing University, Chongqing, China, in 2005., and the PhD degree in computer software and theory from Sun Yat-Sen University 2010. He is an assistant professor at the Department of Computer Science and Technology, Guangdong University of Finance, Guangzhou, China, since 2010. His research fields include the computer architecture and algorithm design for network-on-chip.

Xiaola Lin received the BS and MS degrees in computer science from Peking University, Beijing, China, in 1982 and 1985, respectively, and the PhD degree in computer science from Michigan State University, East Lansing, MI, in 1992. He is currently a professor in the Department of Computer Science, School of Information Science and Technology, Sun Yat-sen University, Guangzhou, China. His research interests include parallel and distributed computing, and computer network.

Maurizio Palesi (M'06) received the MS and PhD degrees in computer engineering from the Universit di Catania, Catania, Italy, in 1999 and 2003, respectively. Since November 2010, he is an assistant professor at Kore University, Enna, Italy. He serves on the editorial board of *VLSI Design Journal* as an associate editor since May 2007. He has served as a guest editor for the *VLSI Design Journal - Special Issue on Networks-on-Chip* in 2008, as a guest editor for the *International Journal of High Performance Systems Architecture - Special Issue on Power-Efficient*, *High Performance General Purpose and Application-Specific Computing Architectures* in 2009, and as a guest editor for *Elsevier MICPRO Journal - Special Issue on Network-on-Chip Architectures and Design Methodologies* in 2010. He serves as the Technical Program Committee member for the following IEEE/ACM International Conferences: RTAS, CODES +ISSS, ESTIMedia, SOCC, VLSI, ISC, and SITIS. He was also the coorganizer of International Workshops on Network-on-Chip Architectures (NoCArc) from 2008 to 2012. He is a member of the IEEE.