International Journal of Computer Systems (ISSN: 2394-1065), Volume 02– Issue 06, June, 2015
Available at http://www.ijcsonline.com/

Design Tradeoff's in Network on Chips and Distributed Adaptive Routing for Spidergon NoC

Yashasvi Chauhan

Department of Computer Science and Engineering
Suresh Gyan Vihar University
Jaipur, India.

Abstract

The performance of the system is heavily coupled by the efficiency of underlying communication and processing architecture. Growing need of high performance computing and decreasing size of technology has enhanced the growth of multicore era. They are more focused on communication then computation. And urge to design and develop an efficient communication architecture has forked a new dimension in computer architecture i.e. Network on chip (NoC). Bus-based system involves arbitration delay. Crossbar involves scalability issue and point to point increases in complexity in routing wires. Hence we connect the cores through network on the Chip. However, NoCs have been proved to be reliable, scalable and exible but still there are a lot of challenges related to fault tolerance, quality of service, power and area that need to be addressed. In this report, adaptive routing algorithms which have been developed to improved the performance of NoC in terms of fault tolerance and congestion awareness are explained. These algorithms are cost-effective and efficient. Firstly, we propose a cost-effective fault tolerant routing algorithm for irregular 2D mesh without using tables. We exploit one hop visibility of LBDR to eliminate tables. This algorithm handles one or many single link faults within 2D mesh and uses reconfigured paths (minimal and/or non-minimal), if links fail. We use turn model based approach to avoid deadlock. Since our method does not need virtual channels to achieve deadlock freedom, it is more area and power effective. Then, we present low-cost, congestion-aware, non-minimal and fully adaptive (CHARM) routing algorithm for 2D and 3D meshes that offers high degree of adaptiveness by permitting cycles in channel dependency graph while remaining deadlock free. CHARM uses only one, two and two virtual channels along the X, Y and Z dimensions, respectively, thus becomes low-cost algorithm. It is a a reconfigurable, deadlock free and cost-efficient fault tolerant routing algorithm without using VCs. To overcome limitation of dimension order routing in spidergon, an adaptive routing algorithm has been proposed which allows the packet to take across link at any place according to current congestion level. Our scheme is minimal path based and packet take across link only once if required. If a packet want to take across link and that link is congested, then it can take left or right link and go for across link at any other intermediate node. Spidergon is a popular NoC (Network-On-Chip) developed to realize cost effective multi processor SoC (MPSoC) using a fixed optimized topology [1]. Increasing diversity of ap-plications, quality of service requirements and deterministic routing schemes inhibit the performance by creating con-gestion bottlenecks. Proposed routing algorithm exploits the path diversity of spidergon NoC and selects the optimal path on the basis of congestion level (CL). CL depicts current traffic conditions. Proposed scheme is deadlock free and uses only one extra virtual channel to achieve improved communication. Proposed scheme is compared with native ndeterministic routing schemes of spidergon NoC i.e. aFirst and aLast. Results demonstrate that the our algorithm distributes traffic evenly while reducing hot-spots. Considerable performance improvement is achieved on modified router architecture.

Keywords: *NoC, SoC, Path diversity, LBDR, adaptive, spidergon.*

I. INTRODUCTION

Technology advancement and shrinking size of transistor has increased the number of IP components on chip. With grow-ing complexity, design paradigm is shifting toward multiple core on single chip instead of a single complex processing element [2]. Therefor focus has been shifted from computation to communication since area, performance and power consumption of SoC mainly depends on underlying interconnect architecture [3]. Network-on-Chip [4] [5] [6], a network based approach to interconnect all components of SoC has been proposed as an alternative to overcome drawbacks of bus and point to point based classical interconnects. NoC comprises of routers, links and network interface. Topology defines the way routers are connected with each other phys-ically using links, networks interface implements protocol to connects ip-cores and routers.

Regular topologies such as 2D-mesh offer good theoretical metrics but cannot be exploited due to nature of traffic in Multi-core SoC applications. But spidergon topology developed by ST Microelectronics provides good tradeoff be-tween theoretical performance and implementation cost [1].

In spidergon an even number of nodes are connected in bidi-rectional ring with an extra link connected to diagonally opposite node. Spidergon has smaller number of edges and competitive network diameter as compared to 2D-mesh and fat-tree for up to 60 nodes. For larger number of nodes, aggregation and hierarchy reduce network diameter and improves performance.

Spidergon is regular, point-to-point and constant degree vertex symmetric network. Hence uniformity and homogeneity offer simple and identical router implementation by reducing design and complexity. Moreover, the spidergon topology translates easily into a low-cost practical layout. Fig 1 is a possible equivalent planar representation in which physical connection between nodes only needs to cross at one point in the chip [7]. In spite of homogeneous spidergon scheme, heterogeneity is introduced due to different size and aspect ratio of IPs.

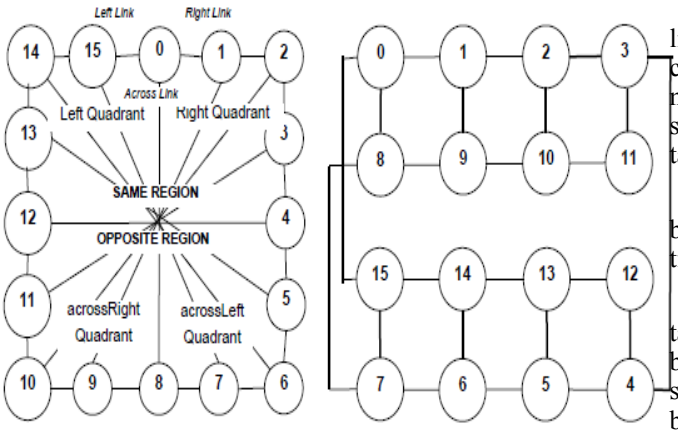

Figure 1: 16 Node Spidergon Topology and Connection Lay-out

Along with topology, choice of routing algorithm greatly affects the performance of NoC. Deterministic routing algorithms are simple to implement but they do not consider the current network status and always generate a fix path for a given pair of nodes. On contrary, adaptive routing algorithms provide more routing path diversity and distribute the traffic more evenly. Routing schemes proposed for spidergon topology are deterministic, source based and shortest path. In this paper we propose a novel adaptive routing algorithm for spidergon NoC architectures. Proposed routing is minimal and distributed in nature. It removes restriction on choice of across link. It allows to take across link at any intermediate node depending upon current network status instead of fixing at first or last and hence adding adaptivity to routing scheme.

The rest of the paper is structured as follows. In section II, an overview of already proposed routing schemes for spidergon architecture is presented. Section III is devoted to the detailed explanation of proposed method including motivation, working of algorithm , deadlock avoidance and modified router architecture. Section IV presents an analytical performance comparison in terms of channel utilization and Section V explains the experimental setup to check the effectiveness and correctness of proposed method using NIRGAM [8]. Finally section VI concludes with result analysis and directions for future work.

II. Related Work

Routing algorithms proposed so far for spidergon are very simple since they take the advantage of symmetry and simplicity offered by spidergon topology. To the best of ourknowledge, acrossFirst (aFirst), acrossLast (aLast) [1], aE-qualized [9] and dynamic stress deflection routing (DSWR)[10] are the proposed algorithms for this architecture. aFirst, aLast and aEqualized are deterministic, minimal and source-based routing. DSWR is table based fault-tolerant adaptive routing. In aFirst routing, if across link is to be traversed then it is taken as first hop and after that packet moves on left or right links. aLast traverse left or right link first and if across link is need to be traversed then it is taken as last hop. Across link is required to be traversed if destination lies in diagonally opposite region. A major limitation of above mentioned algorithms is that across link can be used only at first or last place and only for once even though multiple paths are available.

aEqualized routing algorithm proposed to overcome the limitations imposed by deterministic routing scheme and combines both aFirst and aLast algorithms. This algorithm mainly target networks with few nodes behaving as hot-spot nodes. Having prior knowledge of traffic pattern, it tags routers to run either aFirst or aLast routing algorithm.

This is considered as main limitation of this approach because traffic behavior of NoC may change during run time which could result in a performance degradation.

Dynamic wormhole deflection routing (DSWR) [10] is table based approach which calculates and stores path between each source and destination after mapping spidergon on 2xN mesh. It is fault tolerant and adaptive based upon concept of deflection routing [11]. But in prospect of low cost spidergon NoC it adds computational complexity and enhance the need of memory. Scalability is another issue that arises with table based approach.

In this paper, we overcome limitation of dimension order routing and propose an adaptive routing algorithm which allows the packet to take across link at any place according to current congestion level. Our scheme is minimal path based and packet take across link only once if required. If a packet want to take across link and that link is congested, then it can take left or right link and go for across link at any other intermediate node.

III. Proposed Work

A. *Motivation: Path Diversity in spidergon*

Path is ordered set of channels between any two node. A path is minimal if number of channel in that path is minimum. If more than one minimal path exit between a pair of node, then topology become more robust. This property of topology is called as path diversity [12]. Spidergon exhibits strong path diversity. Our proposed routing scheme explore this property of spidergon to induce adaptivity in routing.

This section presents a brief analysis of routing path diversities available in spidergon topology and paths used by deterministic routing schemes (aFirst, aLast). For all of our analysis, we have chosen source destination pairs which are on opposite side of the ring, because for all other source destination pairs only one minimal path is available

which is the one along the ring either in clockwise or anticlock wise as shown in Figure 1. For comparing path diversities of aFirst and spidergon topology, we have considered node 0 as source tile and all the nodes which are on opposite side ofring as destination node (node 5,6,7,8,9,10,11).

As shown in Table 1, aFirst provides a single path for each destinations. On the other hand, spidergon topology provides almost double paths as compared to aFirst routing scheme except for the one shown in row 1 for which a single minimal path is available.

Table 1: Routing Path Diversity of aFirst and Spidergon Topology

D	No. of available path in aFirst, route	No. of available path in spidergon, routes
8	1,0-8	1,0-8
7	1,0-8-7	2,0-8-7,0-15-7
6	1,0-8-7-6	3,0-8-7-6,0-15-7-6,0-15-14-6
5	1,0-8-7-6-5	4,0-8-7-6-5,0-15-7-6-5,0-15-14-6-5,0-15-14-13-5
9	1,0-8-9	2,0-8-9,0-1-9
10	1,0-8-9-10	3,0-8-9-10,0-1-9-10,0-1-2-10
11	1,0-8-9-10-11	4,0-8-9-10-11,0-1-9-10-11,0-1-2-10-11,0-1-2-3-11

For comparing path diversities with aLast, we have considered node 0 as destination node and all the nodes which are on opposite side of ring as source node (node 5,6,7,8,9,10,11).

As shown in Table 2, aLast provides a single path for all source destination pairs. On the other hand, topology provides almost double paths as compared to aLast routing scheme except for the one shown in row 1 for which a single minimal path is available.

Table 2: Routing Path Diversity of aLast and Spidergon Topology

S	No. of available path in aLast, route	No. of available paths in spidergon, routes
8	1,8-0	1,8-0
7	1,7-8-0	2,7-8-0,7-15-0
6	1,6-7-8-0	3,6-7-8-0,6-7-15-0,6-14-15-0
5	1,5-6-7-8-0	4,5-6-7-8-0,5-6-7-15-0,5-6-14-15-0,5-13-14-15-0
9	1,9-8-0	2,9-8-0,9-1-0
10	1,10-9-8-0	3,10-9-8-0,10-9-1-0,10-2-1-0
11	1,11-10-9-8-0	4,11-10-9-8-0,11-10-9-1-0,11-10-2-1-0,11-3-2-1-0

As per traffic pattern aEqualized tags certain nodes as aFirst and others as aLast. As a conclusion, aFirst and aLast and also aEqualized (combines aFirst and aLast) limit the path diversity provided by spidergon topology. In this paper we have exploited this path diversity provided by spidergon topology. In our scheme packets can take different routes depending on the current status of each link. As cyclic de-pendency may introduce deadlocks, two virtual channels are used for each link.

B. *Proposed Routing Algorithm*

As show in Figure 1 each node in spidergon is identified by a positional numerical value. A network of size N will have have id as 0, 1, 2, 3.....N-1. Spidergon consists of a bi-directional ring in both clockwise, and anti-clockwise directions. At any node We call clockwise links as Right link and Anticlockwise link as Left link. In addition to these links, each node is cross connected to diagonally opposite nodes, i.e. from node i,$0 \leq i < N$ to node (i + n)modN, we call it across link. Diameter of spidergon is defined as $\lceil N/4 \rceil$. Distance between two tile is number of links in min-imal path.

We divide the spidergon network into two region: same re-gion or opposite region in context of any arbitrary node. U is any arbitrary node. A node V is said to be in same region if distance between U and V is less than or equal to diameter otherwise it is said to be in opposite region.

We further divide it into four quadrants: left, right, acrossLeft and acrossRight. U is a arbitrary node and node A is connected directly to U using across link. V is said to be in left quadrant of U if it is in same region and is reached using left link. V is said to in right quadrant of U if it is in same region and is reached using right link. V said to in acrossLeft quadrant of U if it is in opposite region and in left quadrant of A. And V is is said to be right quadrant if it is in opposite region and in right quadrant of A.

Congestion level (CL) has been used as a measure of traffic load on a channel and hence depicting current traffic scenario. To keep router architecture simple we used a simple scheme using virtual channel. Congestion level of any link will be considered as high if requested virtual channel at next node in that particular output direction is not free, otherwise it is considered as low .

Algorithm 1 Algorithm: minCong(X,Y)

```
X,Y : directions
CL[ ] : Congestion Level of directions at a router
I/P : X, Y
O/P: X OR Y
if CL[X] ≤ CL[Y] then
  return X
else
  return Y
end if
```

As per algorithm 2, current node id is compared with destination node id. If destination node id is equal to current node, the packet is routed to core for processing otherwise the quadrant and region of destination corresponding to current node is identified. It checks for quadrant in which destination lies if destination is in same region of current node.

As per quadrant it routes the packet to left or right link. Across link is selected when destination node is diagonally opposite. When destination is in opposite region, it compares the congestion level of across link to left or right link

Algorithm 2 Algorithm: Route

```
CUR : current Node, DST : destination node
C, A, L, R : Core, Across, Left, Right directions
I/P : CUR , DEST
O/P: C OR A OR L OR R
if DST is equal to CUR then
  return C
else if DST is in Left Quadrant of CUR then
  return L
else if DST is in Right Quadrant of CUR then
  return R
else if DST is diagonally opposite of CUR then
  return A
else if DST is in acrossLeft Quadrant of CUR then
  return minConj(A,L)
else
  return minConj(A,R)
end if
```

as per destination quadrant using algorithm 1. When across link is congested, it moves the packet to left or right link so that it can take across link at any intermediate node whose across link is not congested. Figure 2 shows step by step flow of proposed routing algorithm.

Proposed routing scheme utilizes available minimal paths provided by spidergon topology and distributes the traffic accordingly. Packets take different routes depending on the current status of each link.

It behaves like aFirst if destination is in same region or in opposite region of ring and across link is free at first place and like aLast if destination is in same region or opposite region of ring and across link is busy at all places along the path so it route packets first along the ring in either clockwise or anticlockwise direction and at last it uses across link to deliver packet to its destination.

C. *Deadlock and Livelock Avoidance*

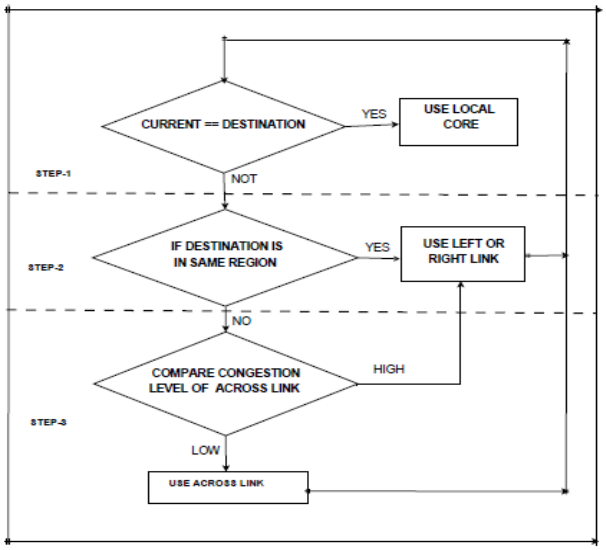

Figure 2: Proposed Method Flow

To ensure correct functionality of any routing algorithm, it must be made deadlock free. Deadlock is the situation where packets holding some resources (buffer or channel) request for resources held by some other packets in a circular way. Spidergon's deterministic schemes like aFirst and aLast re-stricts the location of taking across link to only at first and last place respectively, to avoid circular dependencies between across channel and links along the ring (left or right) leading to deadlock situation. Our scheme does not restricts the location of taking across link to first and last place, it allows it to take at any place along the minimal path. Fur-thermore circular dependencies between across link and the links along the ring may introduce deadlock situation like the one shown in Figure 3. To avoid these type of circular dependencies we have used two virtual channels (VC), along all the links. Out of two, one virtual channel is used as an escape channel. We forced packets to shift from normal vir- tual channel to escape virtual channel after taking across link. For example, packet in VC0 will be assigned VC1 after passing through across link, resulting in zero cyclic dependencies as shown in Figure 3. For avoiding circular dependencies along the ring channels in left and right directions,we have used multiple dateline VC selection algorithm [12].

Livelock is avoided by restricting packets to use only mini-mal path to reach destination.

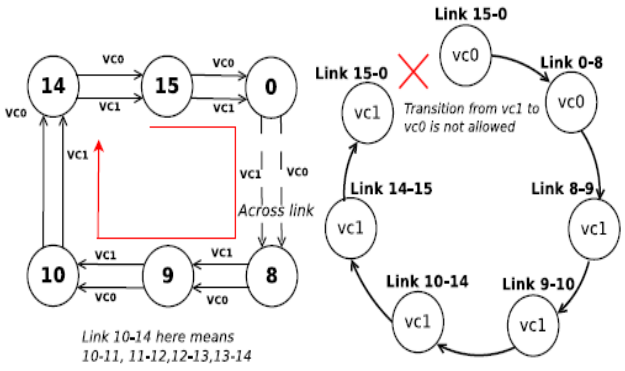

Figure 3: Deadlock Avoidance

D. Router Architecture

Figure 4 shows modified router architecture required for our approach. Two virtual channels are used in each direction. As packets destined for same tile can take either two directions at same time, Across - left or Across - right, we have used two virtual channels in LOCAL direction to avoid contention for a single queue placed at network interface. It will give performance benefit for cases where node act as a hot-spot node.

IV. ANALYTICAL EVALUATION OF TRAFFIC DISTRIBUTION

In this section we compared distribution of traffic generated by proposed scheme with aFirst and aLast. For comparing aFirst with proposed scheme we have chosen node 0 as source node and nodes which are on opposite side of ring as destination nodes (5, 6, 7, 8, 9, 10, 11). As shown in Figure 5, aFirst routing spreads most of the traffic over across link(shown in bold line) resulting in highly congested link and uneven distribution of traffic. On the other side, proposed scheme makes use of all available minimal paths to route the traffic and results in more even distribution of traffic across all links.

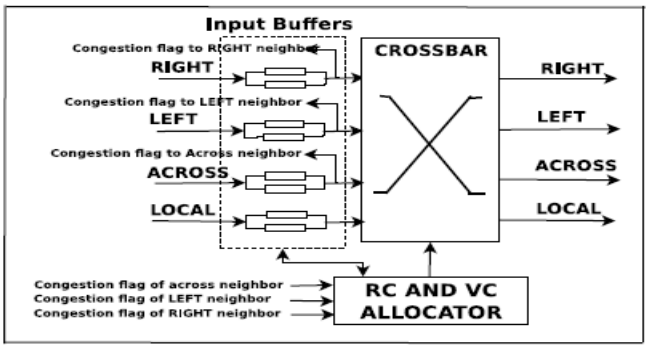

Figure 4: Proposed Router Architecture

For comparing aLast with proposed scheme, source nodes 5, 6, 7, 8, 9, 10 and 11 are sending traffic to one of its opposite side destination, node 0. Similar to previous one, in Figure 5, aLast routing spreads most of the traffic over across link(shown in bold line) resulting in highly congested link and uneven distribution of traffic. On the other side, proposed scheme makes use of all available minimal paths to route the traffic and results in more even distribution of traffic across all links.

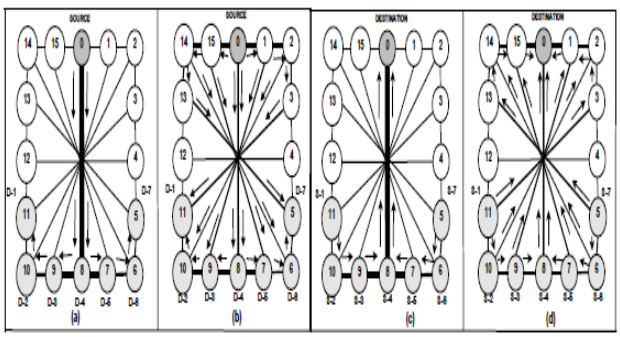

Figure 5: Channel Utilization (a)aFirst (b)Proposed (C)aLast (d)Proposed

V. EXPERIMENTAL ETUP AND PERFORMANCE EVALUATION

For simulation we have used NIRGAM [8] [13] (NoC Interconnect Routing and Application Modeling). NIRGAM is systemC based, extensible discrete event and cycle accurate simulator. We have used wormhole switching in 16 node spidergon topology. Packet is generated by nodes as constant bitrate (CBR) traffic pattern with packet size of 8 flits. In our simulation experiments, Load parameter defines offered rate of packet injection but packet generation is also affected by architectural factors such bandwidth and resources availability.

A. Experiment: Latency and Throughput

In this set of experiments, performance of network under different routing strategies (aFirst, aLast, proposed) and underuniform, non-uniform traffic patterns, respectively, is evaluated.

In uniform traffic pattern, each node generates traffic for all other node with equal probability. Traffic is evenly distributed across all links and each link is equally busy. Since all channels are equally congested so no improvement in latency is seen. As shown in Figure 6, under uniform traffic distribution average latency of proposed scheme is same as aFirst and aLast routings. However, in non-uniform traffic patterns, each node communicate with few nodes in network more frequently than other nodes, resulting in uneven traffic scenarios.

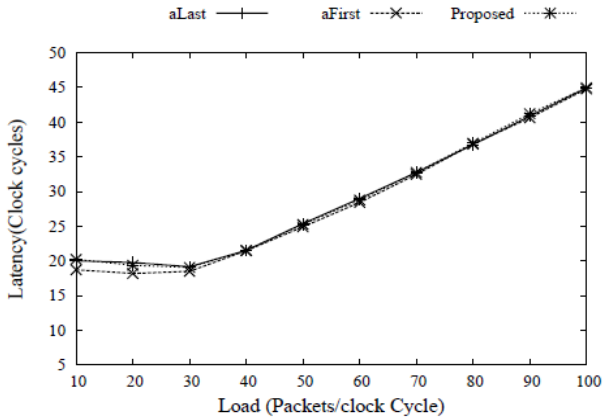

Figure 6: Latency Curve for Uniform Traffic

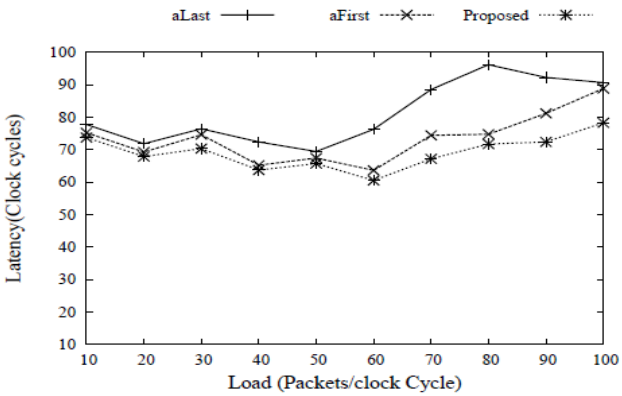

Figure 7: Latency Curve for Hot-spot Traffic

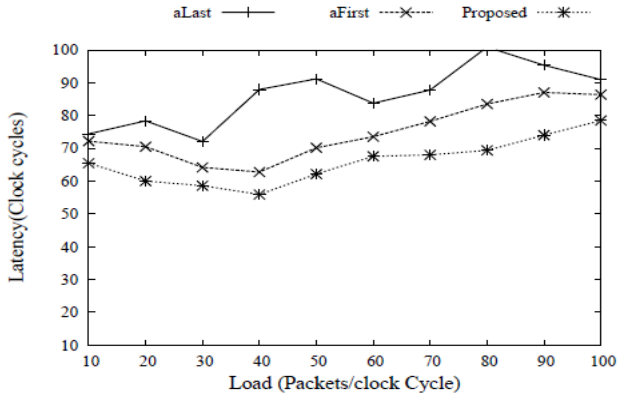

Figure 8: Latency Curve for Random Traffic

We considered hot-spot traffic for generating uneven traffic scenarios and evaluated performance under the same. In case of single hot-spot node, all nodes are sending packets to single destination. aLast routing generates a bottleneck traffic as n/2 traffic of total traffic, needs to pass through a single across link whereas, other links remain unused. On the other side, aFirst, removes this bottleneck problem at some extent as all source nodes first route traffic towards across link. But, in case of congestion it does not adapt as per the current network traffic, and increases traffic on clock-wise and anticlockwise channel of opposite side node because of single deterministic path between each source destinationpair. This scenario could be opposite for cases where a single hot-spot node generates traffic for rest of nodes. Proposed method addressed this problem by distributing traffic across all links. It takes advantage of status of current network traffic and sends traffic towards less congested output ports and results in improved overall average latency as shown in Figure 7. As number of hot-spot nodes increases, proposed scheme distributes traffic evenly and clearly outperforms the two routing algorithms as shown in Figure 8.

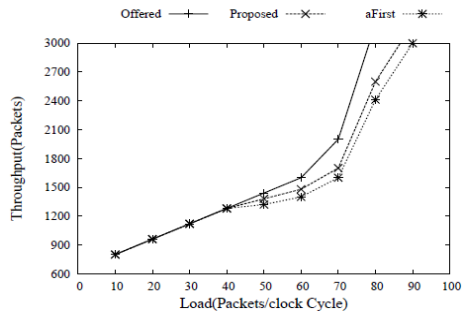

Figure 9: Throughput Curve for Uniform Traffic

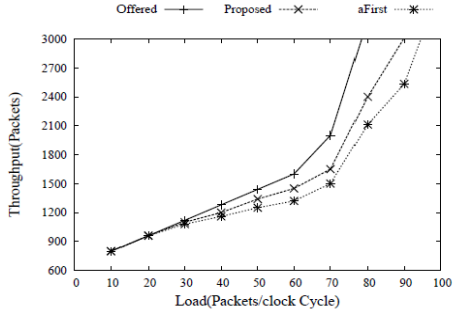

Figure 10: Throughput Curve for Non-Uniform Traffic

When we evaluated the throughput, we compared through-put achieved using our scheme to afirst throughput and offered throughput. As shown in Figure 9, in case of uniform traffic when more amount of traffic is injected then only we can see some improvement in throughput owning to runtime network adaptivity. While as shown in fig 10, unevenness of non-uniform traffic is handled quite efficiently by proposed scheme. Proposed scheme curve is similar to offered throughput curve. Their is quite significant improvement in throughput. Power utilization as shown in Figure 11 is not affected by proposed routing scheme. Since it provide minimal path, average hop count remain same and hence average switch traversed which cause maximum power consumption is same.

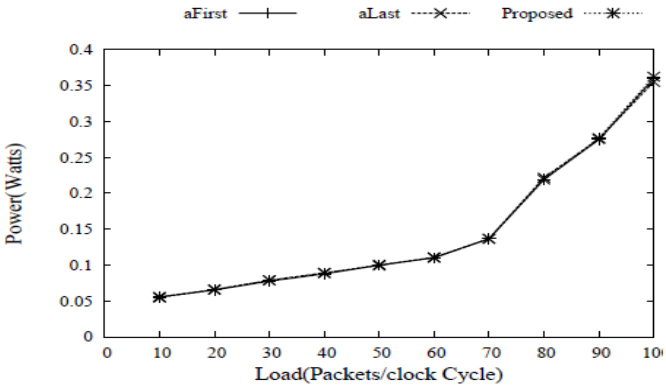

Figure 11: Power Curve for Non-Uniform Traffic

VI. CONCLUSION AND FUTUREWORK

In this paper, we proposed a novel adaptive routing algorithm for spidergon NoC. Proposed routing scheme is minimal, adaptive and distributed in nature. It uses the current network traffic status and distributes traffic across all links evenly by taking advantage of path diversity available in spidergon. For deadlock avoidance we used 2 virtual channels and modified standard router of spidergon architecture by adding support for each virtual channel at each input port to fully exploit the potential of our scheme. In case of non-uniform traffic proposed scheme shows considerable improvement in throughput and latency as compared to deterministic routing. In case of uniform traffic and low congestion in network it performs at par deterministic scheme.

In future, we would extended scheme to add support for fault tolerance by implementing it with logic based distributed routing(LBDR).

REFERENCES

[1] M. Coppola, R. Locatelli, G. Maruccia, L. Pieralisi, and A. Scandurra. Spidergon: a novel on-chip communication network. In System-on-Chip, 2004. Proceedings. 2004 International Symposium on, pages 15–, 2004.

[2] T.G. Mattson, R. Van Der Wijngaart, and M. Frumkin. Programming the intel 80-core network-on-a-chip terascale processor. In High Performance Computing, Networking, Storage and Analysis, 2008. SC 2008. International Conference for, pages 1–11, 2008.

[3] Tobias Bjerregaard and Shankar Mahadevan. A survey of research and practices of network-on-chip. ACM Comput. Surv., 38(1), June 2006.

[4] J. Henkel, W. Wolf, and S. Chakradhar. On-chip networks: a scalable, communication-centric embedded system design paradigm. In VLSI Design, 2004. Proceedings. 17th International Conference on, pages 845–851, 2004.

[5] W.J. Dally and B. Towles. Route packets, not wires: on-chip interconnection networks. In Design Automation Conference, 2001. Proceedings, pages 684–689, 2001.

[6] L. Benini and G. De Micheli. Networks on chips: a new soc paradigm. Computer, 35(1):70–78, 2002.

[7] M. Coppola, M.D. Grammatikakis, R. Locatelli, G. Maruccia, and L. Pieralisi. Design of Cost-Efficient Interconnect Processing Units: Spidergon STNoC. CRC Press, Inc., Boca Raton, FL, USA, 1st edition, 2008.

[8] L. Jain, B. Al-Hashimi, M.S. Gaur, V. Laxmi, and A. Narayanan. Nirgam: A systemc based cycle accurate noc simulator, 2010. http://www.nirgam.ecs.soton.ac.uk/.

[9] N. Concer, S. Iamundo, and L. Bononi. aequalized: A novel routing algorithm for the spidergon network on chip. In Design, Automation Test in Europe Conference Exhibition, 2009. DATE '09., pages 749–754, 2009.

[10] N Satya Lakshmikanth, K N.I. Kumaar, and Sudha S. Dynamic stress wormhole routing for spidergon noc, 2010.http://www.hipc.org/hipc2010/HIPCSS10/.

[11] F. Chaochao, Jinwen Li, Zhonghai Lu, A. Jantsch, and Z. Minxuan. Evaluation of deflection routing on various noc topologies. In ASIC (ASICON), 2011 IEEE 9th International Conference on, pages 163–166, 2011.

[12] W. J. Dally and B. P. Towles. Principles and practices of interconnection networks. Morgan Kaufmann, 2004.

[13] Nirgam .http://wiki.mnit.ac.in/mediawiki/index.php/Nirgam/.