International Journal of Engineering & Technology, *7 (2.7) (2018) 763-769*

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research Paper

Low latency Path Aware XY-X Routing Algorithm for NoC Architectures

Venkateswara Rao Musala[1]*, T.V Rama Krishna[2]

[1]Research Scholar Department of ECE, K L E F, Vaddeswaram, Guntur, 522502, India
[2] Professor Department of ECE, K L E F, Vaddeswaram, Guntur, 522502, India
**Corresponding author E-mail: venkatvlsi@kluniversity.in*

Abstract

Route specific information with the SoC needs a great deal of wiring, which increases the Resistance & Capacitance (RC) component of the system. Network on Chip (NoC) is utilized as the interface to address the problems in SoC, On-chip interconnection network in NoC has gained more consideration over steadfast wiring and buses, like lower latency, scalability and high performance. Present routing algorithms in NoC is suffered from load balancing at incarnation networks under non-uniform traffic conditions, causes increase the NoC trade-offs (latency and throughput). Adaptive routing is a technique to progress the load balance, but previous adaptive routing techniques used uniform traffic patterns to form the routing decisions. This paper proposes a new approach at non- uniform traffic patterns in channel state and path specific, Path Aware Routing (PAR XY-X) uses a timeout piggybacking for acknowledgement and load shedding to avoid congestion which choose optimistic path calculation unit to connect the destination node without glue logic decisions in routing. PAR XY-X outperforms the Normal XY routing by 20% and 33% with respect to Avg.latency and throughput

Keywords: *Latency; NoC (Network-on-Chip); PAR (Path Aware Routing); SoC (System-on-Chip);*

1. Introduction

SoC is a sort of small scale framework that incorporates numerous parts like processors, Digital signal processors and memory elements which generally perform errands of functions on specific dies. In [1], there are two confinements of SoC: as a matter of first importance, the correspondence among the Intellectual Property (IP) obstructs the improvement at the framework level and enhance the wire delays. Another issue with SoC is that SoC continuously incorporates many hard-core processors for various applications on a similar chip, due to these issues there is no common phenomenon for centralized SoC application as the design scales down [3,7,24-27], there is a need for NoC fabric to scale the design and high band width which arises trade-offs in NoC. Serialization latency translates the processor core into idol and energy hungry. As a result, reducing the serialization latency is critical to achieve performance in future Multi core processors.

An interconnection network establishes multiple communication paths between Source Node (SN) and Destination Node (DN). A routing algorithm is used to direct the data packet from source to the destination, Existing routing algorithms are concentrating on oblivious routing techniques such as Dimension Order Routing (DOR), which route packets to destination irrespective of the load balancing, though these algorithms [5] have less complexity they perform poor communication because of load imbalance. Adaptive routing is a technique to transfer the packet through the less congested path with channel weights for every S-D pair. For the implementation of the topology, routing algorithms [1] plays a major role. The selection of the required routing algorithm is based on the INF (Interconnection Network Fabric) with optimized network trade-offs such as latency, throughput, energy, area.

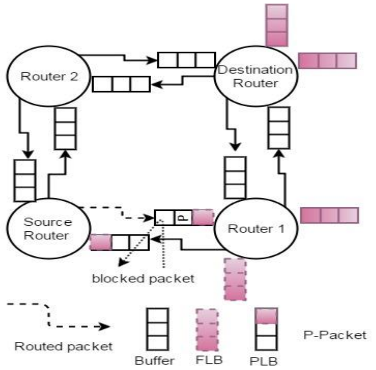

Fig.1 packet communication between nodes

The Fig.1 shows the transfer of packet (P) from source router to destination router, before transmitting the packet to the destination router packet must pass through the router 1, even though router1 suffers from congestion.

Congestion of path causes due to three stages:

1) Router contention: many packets wants to get the same channel (majority north direction) for communication leads to router contention

2) Link congestion: due to the insufficient memory space (buffer size) at the input buffers, buffers are unable to allocate with new packets

3) Router congestion: Whenever a packet is transmitted through output port (i.e. north), a few packets that receive an unsuccessful output request must be infertile and then be queued at the input buffers. The routed packet needs to wait for this channel to be released.

2. Related work

Simply abandoning support of the shared medium access without network topology modification increases the probability of packet losses in routers since control over the flow of packets sent by end nodes into the network is lost. When destination node needs the packet then the source node will send the packet in these cases there is a possibility of congestion[8-13] due to the sharing of common resources among the routers Usually, the congestion is not caused by the router being a blocking one, the true reason for congestion lies in the limited bandwidth of a specific output port, which is defined by parameters of the interconnection network If the incoming traffic is unevenly distributed among output ports, it is easy to imagine a situation[19] in which traffic with total intensity exceeding the network topology maximum is directed to some of the router's output ports

Most of the network topologies will have a quadratical implementation. Here the nodes are organized in an n-dimensional area. A perfect example for this can be an n-dimensional mesh. There are many topologies for the implementation of data transfer in NoC like mesh, torus etc. For the implementation of the topology routing algorithms plays a major role. Routing algorithm for the topology is selected based upon the requirements like for a normal deterministic routing with equal lengths between the nodes XY routing can be used to avoid obstruction of data between the S-D pair. The selection channel changes with the strategies taken into consideration like latency, network traffic and etc. [8]. Ville Rantala et al.[1] outline the features of oblivious routing algorithms in NoC era, Jongman Kim et al.[2] used a two pipeline approach and look ahead routing to minimize the latency in interconnection networks but which increase the area consumption of the network router, Umit Y. Ogras et al.[3] described a latency model with respect to *M/G/*1*/m* queuing model which describes the latency of channel based on the packet latency, Abbas Eslami Kiasari et al.[4] proposed a latency analysis model based on G/G/1 queuing model. En-Jui Chang et al[7] used a contention prediction technique to calculate the NoC-trade-offs A Routing-Algorithm (RA) determines how the data packet is routed from source to destination, oblivious and adaptive routing algorithms plays a major role in NoC, in oblivious routing data is routed without considering the channel traffic whereas adaptive routing considers both channel and router contention and which avoids the congestion in the NoC architectures, most of the last logical latency styles inside the on-chip networks used wormhole-switching[1]. Many networks are invented in NoC to develop a any particular topology by adding traffic patterns. in [7] uses a FIFO model to enhance the performance and overcomes the allocation of buffer space based on the size of the data packet is in communication among S and D nodes in NoC-based systems, nevertheless the techniques can not able to handle the wormhole-switched networks. The authors in [16] proposed link capacity allocation in NoCs by utilizing an analytic serialization latency model. However, the proposed model works only for networks with single flit buffers and which ignores the queuing delays and network contentions. A better analytical router model is proposed in [21]. The previous work assumes Poisson distribution to inject the data packets into the channel such models do not have the accuracy for replacements in applications with congested traffic patterns in many applications. In [28] A worst case(WC) investigation of flow latency is described for non-uniform data packets, the article [29] proposes an optimized traffic regulation parameters and avoid the congestion in the network channel although this approach is not good for random traffic such a system with real-time requirements, many NoC-based systems have more challenging timing constraints, in the proposed approach piggy backing technique is used to get the acknowledge from the recipient node on the basis of timeout basis, if there is no acknowledge from the destination node in the specified time the PAR XY-X algorithm choose another alternative path from Source-Destination in the next cycle of the clock due to use of timeout piggybacking[17] channel allocation latency is reduced, load shedding approach is used to place the data packets in priority order as per the sender node. To the best of our knowledge, the work proposes the first model to consider the on-chip routers average latency which takes into consideration. The proposed model can often develop an intensive performance analysis for network topology with wormhole switching under random traffic [33] patterns with real time data packet processing. Our proposed model, besides providing network trade-offs, for average latency and throughput based on the availability of buffer depth at each router in the network which, gives useful feedbacks around the network behavior which can be useful within the optimization of network latency, application mapping.

3. Materials and Methods

The routing algorithm is used for crucial networks to optimize the network latency. An enhanced routing-algorithm is used to supervise the load over the network, increase the throughput and optimize the serialization latency among the links during the inclusion of non-uniform traffic [18] patterns like transformation of traffic and random traffic [16]**.** Astonishingly, many routers that were built and are also being used today do a pitiable job of balancing the load. Rather, the traffic between each pair of nodes follows solitary, predetermined path. As proposed by [3,5,7,11-14], non-uniform traffic patterns can induce massive load misbalances in the network and provides less throughput. However, these routing choices can attend least partially because many of these routers happen to be built to optimize another critical facet from a routing-algorithm.

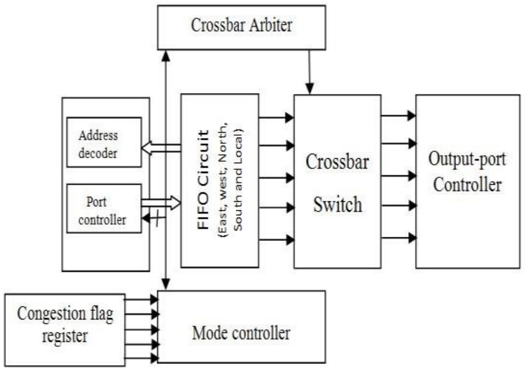

Fig. 2 Basic router architecture for PAR algorithm

There are two routing orientations are exist in interconnection network architectures **i-**Oblivious Routing: an oblivious routing approach is specified by path system which contains number of optional paths from S-D **ii-** Adaptive routing [1], in Oblivious routing algorithms data packet is routed based on the topology adapted in the network, this phenomenon arises congestion and load, the better approach for the issue is to use a effective path aware routing methodology or minimal traffic [15] to minimize the NoC trade-offs. X-Y routing-algorithm [27], in XY routing the data packet first move in the X-direction after that which moves in the direction of destination node by using the best path selection among the available paths at the same time it is lovelock free [1] because of the simplicity in the routing decisions make it beneficial for many NoCs [31]. However, it is deterministic in nature low latency aided by the guaranteed reliability.

4. PAR XY-X Algorithm

A well-designed routing-algorithm choose the communication channel lengths as optimize as possible, reduce hop count, overall serialization latency of a message through a channel, balancing the load and maximizing throughput. In fact, for *oblivious routing-algorithms*, to improve load balance over *all* traffic patterns and we are forced to increase the average path length of all data packets. The converse is also true. This trade-off exists for oblivious algorithms because they do not factor the current traffic pattern into the routing algorithm. Another imperative part of a routing-algorithm is its capacity to work within the sight of deficiencies in network. If the network fails to communicate between S-D nodes, and the entire system fails even though the algorithm can have a reprogrammed or adapt to the failure, the system can continue to operate with only a slight loss in performance. Obviously, this is critical for systems with high-reliability demands. Finally, routing interacts with the flow control of the network and careful design of both is often required to avoid deadlocks and/or lovelocks

The Fig.3 shows a 6x6 mesh structure in NoC, the Processing-Elements (PE)'s or Nodes P(0,0) –P(5,5) represents of 36 PE's each PE in the mesh network can able communicate with any of the PE in the entire network, each PE is connected through a router as shown in Fig.1, router is used to establish the communication path from source PE to Destination PE with the help of PAR XY-X routing-algorithm, routing algorithms are must be prone to latency, high throughput, low power consumption and reliability

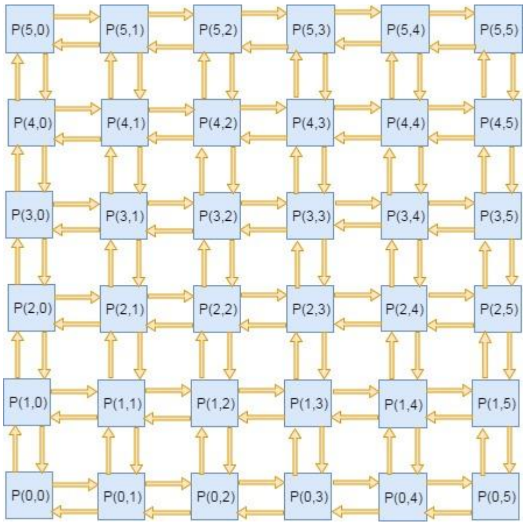

Fig. 3 6X6 Mesh Structure

PAR XY algorithm is measures the congestion and communication cost of all possible paths from source to destination a load shedding technique is used to avoid the congestion [20] and timeout piggybacking [28] is used to reduce the waiting time of the packet in the router

In PAR XY algorithm packet is send in the direction for which the local channel has the lowest *load*. We may approximate the load by either measuring the length of the queue serving this channel, recording how many packets it has transmitted over the last T slots and the calculation of communication cost from the Source Node (SN) to Destination Node (DN). Note that this decision is applied only once at the source node to minimize the latency in the channel and router

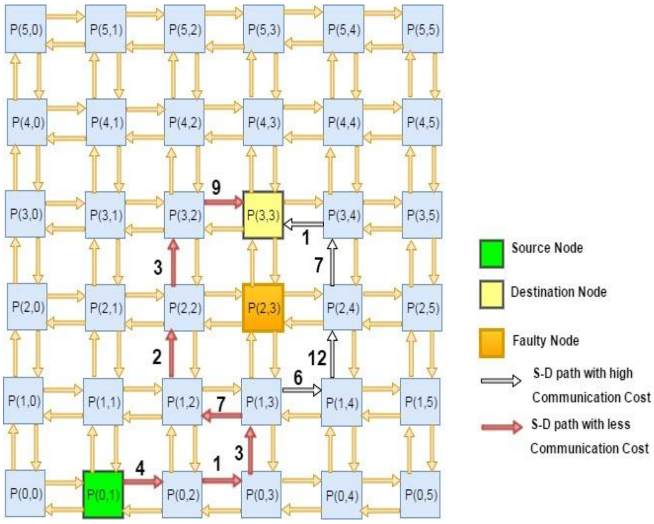

Fig. 4 6x6 Mesh with detection of faulty node

This following example has shown how the choice of routing function can significantly affect load balance, congestion control and detection of faulty node using PAR XY-X algorithm. In fig. 4 the SN is P(0,1) and the DN is P(3,3), usually XY routing means the first routing path must be in the X-direction from the SN after that the routing-algorithm creates path from SN to DN based on the communication cost between the SN and DN with optimized congested path, in the interconnection networks throughput is the main factor for effective communication

Source Node (SN) = P (0, 1)
Destination Node (DN) = P (3, 3)
Faulty Node (FN) = P (2, 3)

There are three communication paths from SN to DN,

P1= P(0,1)→P(0,2)→P(0,3)→P(1,3)→P(2,3)→P(3,3)
P2 = P(0,1)→P(0,2)→P(0,3)→
P(1,3)→P(1,2)→P(2,2)→ P (3, 2)→P (3, 3)
*P*3 = P(0,1)→P(0,2)→P(0,3)→
P(1,3)→P(1,4)→P(2,4) →P (3, 4) →P (3, 3)

Among all the above possible paths SN is unable to communicate through path-1(P1) because there is a faulty node P(2,3), the faulty node is identified using PAR XY-X with the help of timeout piggybacking technique in this technique the SN sends the request to the adjacent node in the X direction if there is an acknowledgement from the recipient node with in specified time limit then the SN establish the path between SN and adjacent node and then forms the communication path from recipient node in Y direction, if there is no acknowledgement then the SN sends the request to the adjacent node in –X direction then Y direction from the recipient of –X direction. there are two alternative paths from SN to DN Path-2(P2) and Path-3(P3) now SN calculates the Communication cost between to existing paths communication cost(CC) is calculated based on the distance between the SN and DN, the CC of the P2 is CC-P2=4+1+3+7+2+3+9=29 and CC-P3 = 4+1+3+6+12+7+1=34 based on the paths communication costs SN selects the P2 for communication among the SN and DN.

Communication cost is calculated using the bandwidth used to transmit the packet and the location of nodes to be communicate

Communication cost = channel bandwidth* distance between the nodes.

5. Results and Discussions

5.1. Simulation Tool

This paper uses a cycle accurate C++ based Network-on-Chip based simulator [34] to perform the evaluation on applied input stimuli with different buffer depths and traffic patterns. The proposed PAR XY-X routing is applied to 16X16 mesh network. This paper uses a wormhole-switching and FIFO arbitration to minimize the serialization latency at the time of arbitration. The inter-router communication cost is calculated using the timeout piggybacking technique and load shedding is used to avoid the congestion among the routers and channel before evaluate the desired network using NoC simulator we assume that one cycle is required to transfer a single flit to the successive router and each packet can have a flits ranges from one flit to nine flits including the head flit, each simulation runs up to 11,000 cycles and 1,000 cycles are used for warm-up time for NoC system

5.2. Traffic patterns

The proposed PAR XY-X algorithm is tested for different traffic scenarios like random and uniform the results are compared with existing [29-31] the results showing that the proposed algorithm is having a better latency and throughput.

5.3. Evaluation Metrics

The Avg. Latency and Throughput is considered as performance metrics [4,28] the time taken to send a data packet or message through the network which includes the injection of head flit in to the channel, waiting time in the channel and the receiving of tail flit at DN. Throughput is defined as the amount of data traffic (bits/sec) sent to the DN.

5.4. PAR XY-X Algorithm

```
Let source nodes are s1, s2:
Destination nodes are d1, d2:
B1, B2 are the blocked nodes;
while ((s1!=d1)&&(s2!=d2) )
{
    while (s1! =d1)
  {
 If (s1==b1)
   {
                 If(s2>d2)                    Decrement s2;
                 else if (s2<d2)
                     Increment s2 and count;
                        Break;
              }
              If (s1>d1)
              {           Decrement s1 and Increment count;
              else if (s1<d1)
                  Increment s1 and Increment Count.;
 }
          }
  while (s2! =d2)
    {
             If (s2==b2)
{
    If s1>d1
       Decrement s1;
    else if (s1<d1)
        Increment s1 and count;
  Break;
}
            If (s2>d2)
            {
               Decrement s2 and Increment count;
            else if (s2<d2)
                Increment s2 and Increment Count;
             }
           end if;
           end all;
        }
```

In this paper 16x16 mesh topology is designed and simulated with flit size of 64 for different traffic patterns like random, uniform and transpose1 with the help of XY-routing and PAR XY-X routing algorithms the results are showing that PAR XY-X routing performs outfit when compared with existing XY-routing algorithm the simulation is run with different PIR(Packet Injection Rates) rates from 0.2 - 1 packets/cycle the corresponding Avg.latency and throughput is calculated using cycle based NoC simulator for proposed algorithm the results are shown that the proposed algorithm is better than the existing XY-routing algorithm

Data packet to be sent from SN to DN is divided into three different types of flits those are Head flit, Body flit and Tail flit the no of body flits are depends on the size of data packet to be sent which are ranges from 1-9, at first the head flit is sent from SN to adjacent node through the link established by the PAR XY-X routing algorithm from SN to DN. if the adjacent node is ready to serve the SN then the adjacent node sends an acknowledgement while sending a acknowledgement from the recipient node which adds the data packet sent by the another adjacent node due to this phenomenon same communication channel is used for both acknowledgement and data transmission at same time communication cost between PE's is reduced up to optimum level but this technique enhances the area required to design the hardware. The status of routed flits is represented by using to control flags (VALID and WAIT) VALID represents the data packet is valid and WAIT indicates receiver node received the data packet

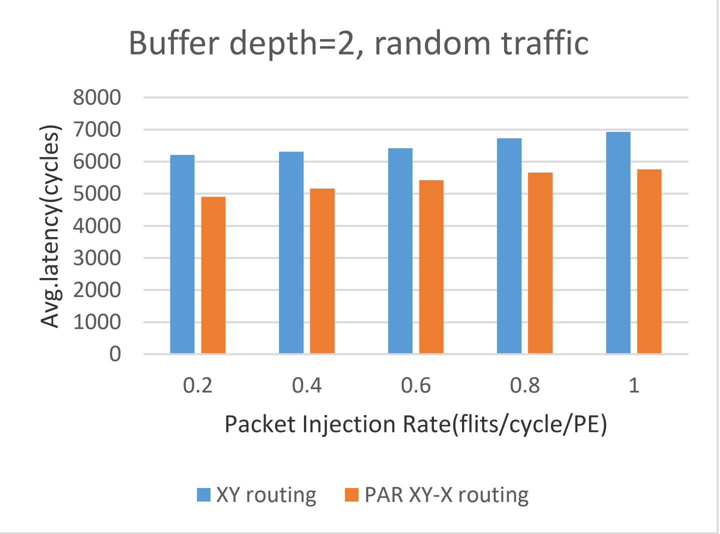

Fig.5 PIR vs. Avg.latency for BD=2, random traffic

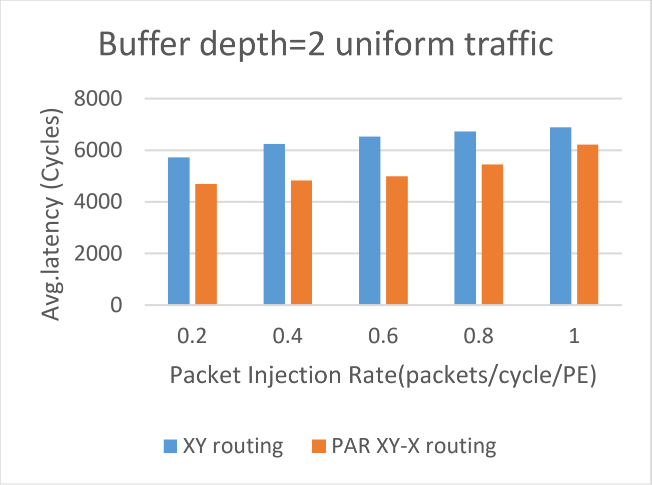

Fig.6 PIR vs. Avg.latency for BD=2, uniform traffic

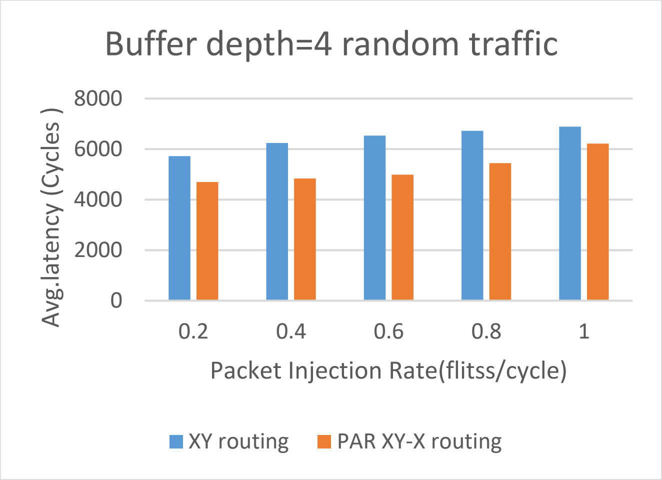

Fig.9 PIR vs. Avg.latency for BD=4, random traffic

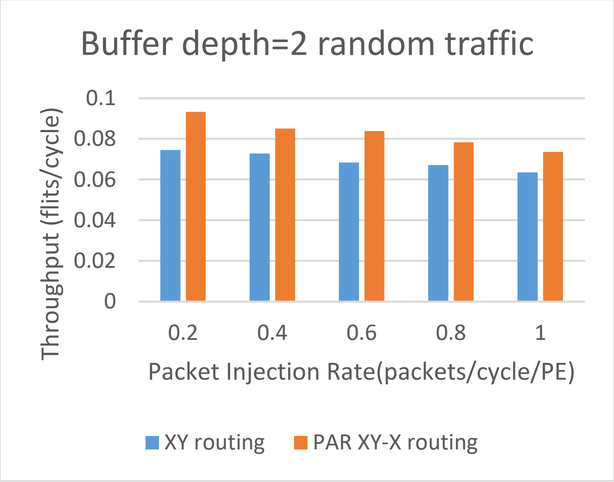

Fig.7 PIR vs. Throughput for BD=2, random traffic

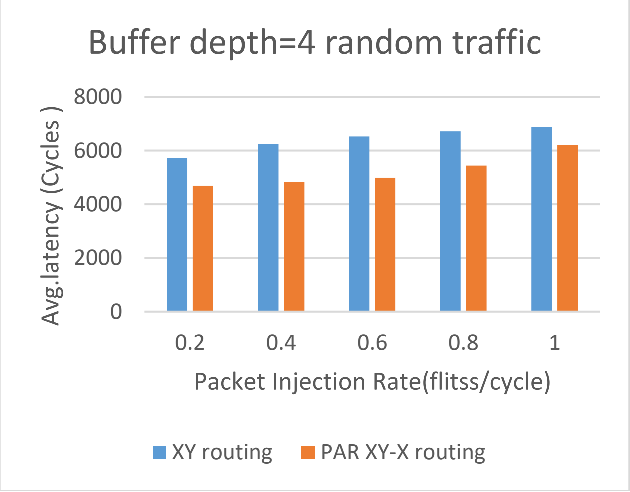

Fig.10 PIR vs. Avg.latency for BD=4, random traffic

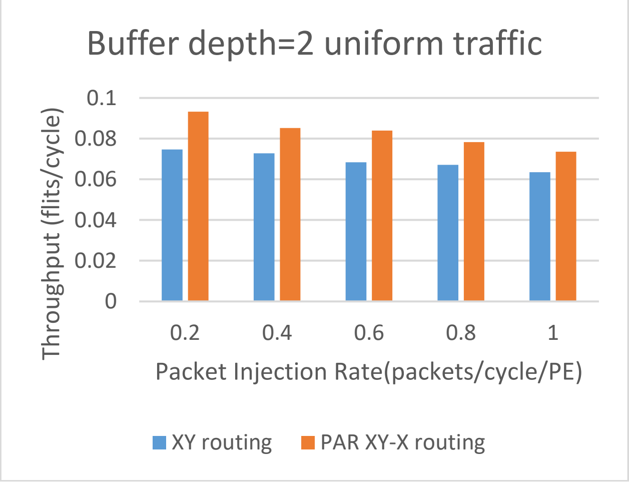

Fig.8 PIR vs. Throughput for BD=2, uniform traffic

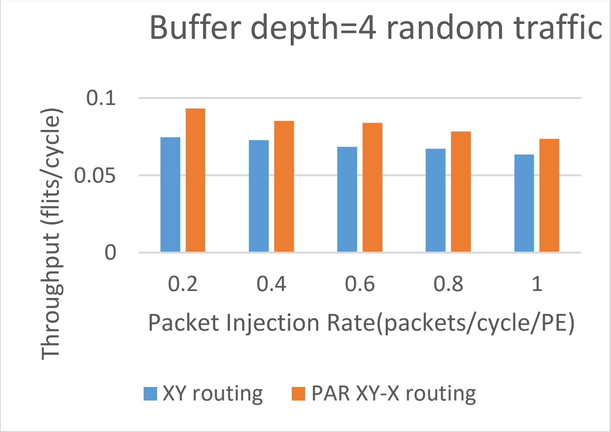

Fig.9 PIR vs. Throughput for BD=4, random traffic

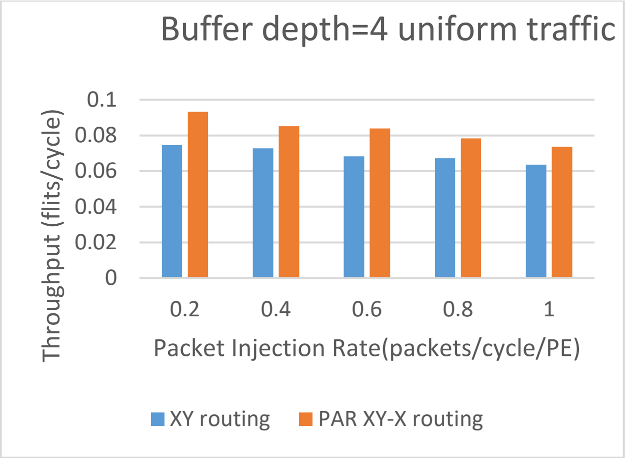

Fig.12 PIR vs. Throughput for BD=4, uniform traffic

Avg. latency and throughput of 16x16 mesh topology is measured using proposed PAR XY-X and conventional XY routing algorithms using random and uniform traffic patterns with different Buffer Depth(BD) values of 2 and 4.

Fig.5-12 shows the proposed PAR XY-X routing algorithm is more accurate than the existing routing algorithms with respect to Avg.latency and Throughput for BD=2 with random and uniform traffic patterns

6. Conclusion

Nowadays the Network-on-Chip (NoC) is considered as feasible solution to address the Avg.latency and throughput issues which effect the present multicore architectures. In this paper the performance of the proposed PAR XY-X algorithm is evaluated with random and uniform traffic patterns for 16X16 mesh topology. From the comparison the results showing that the proposed algorithm is better with respect to Avg.latency and Throughput by 20% and 33% respectively.

References

[1] Ville Rantala, TeijoLehtonen , JuhaPlosila “Network on Chip Routing Algorithms” in August 2006 .

[2] Jongman Kim,Dongkook Park, T. Theocharides, N. Vijaykrishnan, Chita R. Das “A Low Latency Router Supporting Adaptivity for On-ChipInterconnects” in September 2005 IEEE XPLORE.

[3] Umit Y. Ogras, Paul Bogdan, and RaduMarculescu“An Analytical Approach for Network-on-ChipPerformance Analysis” in DECEMBER 2010, IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS, VOL. 29, NO. 12.

[4] Abbas EslamiKiasari, Zhonghai Lu, and Axel Jantsch “An Analytical Latency Model for Networks-on-Chip” in January 2013, IEEE TRANSACTIONS ON VERY LARGE-SCALE INTEGRATION (VLSI) SYSTEMS, VOL. 21, NO. 1.

[5] Sahar Foroutan, YvainThonnart, and Frederic Petrot “An Iterative Computational Technique for Performance Evaluation of Networks-on-Chip” in August 2013, IEEE TRANSACTIONS ON COMPUTERS, VOL. 62, NO. 8.

[6] HUSSEIN G. BADR AND SUNIL PODAR”An Optimal Shortest-Path Routing Policy for Network Computers with Regular Mesh Connected Topologies” in October 1989, IEEE TRANSACTIONS ON COMPUTERS, VOL. 38, NO. 10.

[7] En-Jui Chang, Hsien-Kai Hsin, Shu-Yen Lin, and An-Yeu (Andy) Wu “Path-Congestion-Aware Adaptive Routing with a Contention Prediction Scheme for Network-on-Chip Systems” in January 2013, IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS, VOL. 33, NO. 1.

[8] Yu-Hsin Kuo1, Po-An Tsai1, Hao-Ping Ho1, En-Jui Chang2, Hsien-Kai Hsin2, and An-Yeu (Andy) Wu “Path-Diversity-Aware Adaptive Routing in Network-on-Chip Systems” in November 2012, 2012 IEEE 6th International Symposium on Embedded Multicore SoCs.

[9] Ge-Ming Chiu, the Odd-Even Turn Model for Adaptive Routing” in July 2000, IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 11, NO. 7.

[10] William J. Dally and Hiromichi Aoki “Deadlock-Free Adaptive Routing in Multicomputer Networks Using Virtual Channels” in April 1983, IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, VOL. 4, NO. 4.

[11] Partha Pratim Pande, Cristian Grecu, Michael Jones,Andre ´ Ivanov, and Resve Saleh “Performance Evaluation and Design Trade-Offsfor Network-on-Chip Interconnect Architectures” in August 2005, IEEE TRANSACTIONS ON COMPUTERS, VOL. 54, NO. 8.

[12] Wen-Hsiang Hu1, Jun Ho Bahn2 and Nader Bagherzadeh1 “Parallel LDPC Decoding on a Network-on-Chip Based Multiprocessor Platform” in21st International Symposium on Computer Architecture and High-Performance Computing.

[13] Yue Qian,Zhonghai Lu, and Wenhua Dou “Analysis of Worst-Case Delay Bounds for On-Chip Packet-Switching Networks” in May 2010, IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS, VOL. 29, NO. 5.

[14] Erland Nilsson, Mikael Millberg, Johnny Oberg, and Axel Jantsch “Load distribution with the Proximity Congestion Awareness in a Network on Chip” in December 2003, Design, automation and test in Europe Conference and Exhibition.

[15] Luca Benini, Giovanni De Micheli “Networks on Chip:A New Paradigm for *Systems on Chip* Design” in IEEE Explore.

[16] Arjun Singh, William J Dally, Amit K Gupta, Brian Towles “GOAL: A Load-Balanced Adaptive Routing Algorithm for Torus Networks” in June 2003, Computer Architecture, 2003.Proceeding. 30th Annual international Symposium.

[17] Rohit Sunkam Ramanujam, Bill Lin “Destination-Based Adaptive Routing on 2D Mesh Networks” in November 2010, Architectures for Networking and Communications Systems (ANCS), 2010 ACM/IEEE Symposium

[18] Loren Schwiebert2 and Renelius Bell “Performance Tuning of Adaptive Wormhole Routing through Selection Function Choice1” in 2000, Journal of Parallel and Distributed Computing

[19] Kun-Chih Chen, Shu-Yen Lin, Hui-Shun Hung, and An-Yeu (Andy) Wu “Topology-Aware Adaptive Routing for Non-Stationary Irregular Mesh in Throttled 3D NoC Systems” in IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS.

[20] Paul Gratz, Boris Grot, Stephen W. Keckler “Regional Congestion Awareness for Load Balance in Networks-on-Chip” in October 2008, High Performance Computer Architecture, 2008. HPCA 2008.

[21] Jingcao Hu RaduMarculescu “DyAD – Smart Routing for Networks-on-Chip” in May 2005, Design Automation Conference, 2004. Proceedings. 41st.

[22] VINCENZO CATANIA, ANDREA MINEO, and SALVATORE MONTELEONE “Cycle-Accurate Network on Chip Simulation with Noxim” in September 2015, Application-specific Systems, Architectures and Processors (ASAP), 2015 IEEE 26th International Conference on.

[23] MAKSAT ATAGOZIYEV “ROUTING ALGORITHMS FOR ON CHIP NETWORKS” in December 2007.

[24] Z. Guz, I. Walter, E. Bolotin, I. Cidon, R. Ginosar, and A. Kolodny, “Network delays and link capacities in application-specific wormhole

nocs,” *J. VLSI Design*, vol. 2007, 2007, Article ID 90941

[25] J. Sepulveda, M. Strum, W. Chau, and G. Gogniat, “A Multi-Objective Approach for Multi-Application NoC Mapping,” in *2011 IEEE Second Latin American Symposium on, Circuits and Systems (LASCAS)*, Feb. 2011, pp. 1–4

[26] J. Hu, U. Y. Ogras, and R. Marculescu, “System-level buffer allocation for application-specific networks-on-chip router design,” *IEEE*

[27] *Trans.Comput.-Aided Design Integr. Circuits Syst.*, vol. 25, no. 12, pp. 2919–2933, Dec. 2006

[28] A. E. Kiasari, H. Sarbazi-Azad, and S. Hessabi, “Caspian: A tunable performance model for multi-core systems,” in *Euro-Par 2008 Parallel Processing*, E. Luque, T. Margalef, and D. Benitez, Eds. New York: Springer-Verlag, 2008, pp. 100–109, Lecture Notes in Computer Science.

[29] F. Jafari, Z. Lu, A. Jantsch, and M. H. Yaghmaee, “Buffer optimization in network-on-chip through flow regulation,” *IEEE Trans.*

Comput.- Aided Design Integr. Circuits Syst., vol. 29, no. 12, pp. 1973–1986, Dec. 2010.

[30] Nezam Rohbani et.al., LAXY: A Location-Based Aging-Resilient Xy-Yx Routing Algorithm for Network on Chip," *IEEE Trans. Comput.- Aided Design Integr. Circuits Syst.*, Volume:36,Issue:10,pp.

[31] Amir Hosseini, Tamer Ragheb, and Yehia Massoud. A fault-aware dynamic routing algorithm for on-chip networks. In *Circuits and Systems, 2008. ISCAS 2008. IEEE International Symposium on*, pages 2653–2656. IEEE, 2008.

[32] Arseniy Vitkovskiy, Vassos Soteriou, and Chrysostomos Nicopoulos. A highly robust distributed fault-tolerant routing algorithm for nocs with localized rerouting. In *Proceedings of the 2012 Interconnection Network Architecture: On-Chip, MultiChip Workshop*, pages 29–32. ACM, 2012.

[33] Sourceforge(2008) Noxim: Network-on-chip simulator available online

[34] U. Y. Ogras, P. Bogdan, and R. Marculescu, "An analytical approach for network-on-chip performance analysis," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 29, no. 12, pp. 2001–2013, Dec. 2010.

[35] Vincenzo Catania et.al cycle accurate network on chip sim lation with noxim ACM Trans. On Modeling and computer simulation vol 27, No1, article 4 aug-2016