2D Hexagonal Mesh Vs 3D Mesh Network on Chip: A Performance Evaluation

Ravindra Kumar Saini[1] and Mushtaq Ahmed[1]

[1]*Department of Computer Science and Engineering, MNIT, Jaipur, India*

Abstract: 3D Network on Chip (NoC) has emerged as a new platform to meet the performance requirements and scaling challenges of System on Chip. More investigations require addressing challenges in multiport topologies, minimizing foot printing of nodes and interconnections of wires. This paper discusses multi-port NoC topologies and routing in 2D hexagonal and 3D mesh NoC. Deadlock free routing for 2D hexagonal mesh topology is compared to ZXY routing in 3D mesh with similar number of nodes. Routing algorithms shows promising results, thus making the architecture and algorithm more suitable for the future NoC design.

Keywords: Reliability, 3D NoC, Hexagonal mesh, Crossbase Routing, ZXY Routing

1. INTRODUCTION

Addition of new ports in the router can overcome problem of limited bandwidth and scaling. In conventional 2D mesh NoC, an addition of one and two ports in intermediate tiles can be done as shown in Figure 1 and Figure 2 to make a 2D hexagonal mesh. A single diagonal link is required to increase from four to six port interconnections. Figure 3 shows conventional 3 x 3 x 3 mesh topology. 3D NoC, architecture is constructed using multiple uniform silicon planes and each active plane is a 2D mesh topology connected with vertical links or interconnection wires. An efficient routing is required to explore all the available paths to optimize the performance in terms of throughput, latency and energy. NoC design must be simple and short-path route is highly considered for low latency and power dissipation. Figure 2 represents an example of 7 port Hexagonal mesh router and 6 ports Diagonalized mesh router respectively.

Recent development in Nano scale has open an option of an alternative to conventional on-chip communication network with a uniform stackable multi-chip modules (MCM) in three dimensions using through silicon via (TSV). The transition from 2D to 3D NoC is done by homogeneously distributing the tiles on to different layers of 3D NoC. [1] Explained design of homogeneous network on distinct layers using heterogeneous floor plans. The router assignment based design methodology is used for placement of processing element (PE) on first layers and their minimized connection to the routers is placed on the second layer. Application specific NoC design with optimized power consumption and minimum area dealt with ripup and reroute procedure for routing flows and a router merging procedure to optimize a given network topology[2].

In [3] virtual channels are used, to achieve fault tolerance in k-ary n-cube topologies. Their method uses $O(2^n)$ virtual channels for a fully adaptive fault-tolerant routing algorithm. [4] Introduces planner adaptive routing algorithm to reduce the number of virtual channels in an n-D mesh. Their routing is partially adaptive and routing process is divided into a sequence of phases and then forwarding packets in two dimensions within each phase.

The Simulation Allocation (SAL) method [5] is used to determine the best suitable network topology for a given application 3D NoC along-with optimizing the power, network latency and the data traffic path respectively. The scalability issue of 2D and 3D NoC (mesh and bus topology) is addressed in [6] based on buffer-less hot-potato routing algorithm for mesh topology, while bus protocol incorporates centrally arbitrated least-served- first priority scheme. Emergence of 3D IC technology enabled the short vertical connections between the dies by the means of Through Silicon Via (TSV) [7]. The main challenge to 3D NoC is thermal dissipation as the power density and length of

E-mail:ravisaini.bits@gmail.com, mahmed.cse@mnit.acl.in

heat dissipation path increases with vertically stacking of dies, resulting high temperature and longer propagation delay. The leakage power also adversely affects system reliability. Vertical stacking and gluing of wafers for 3D node integration has been presented in [8].
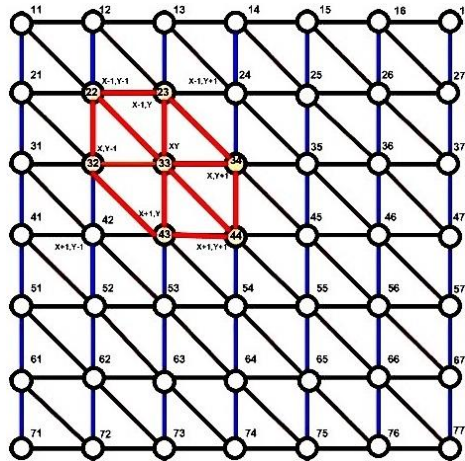
(a) Hexagonal Mesh type 1
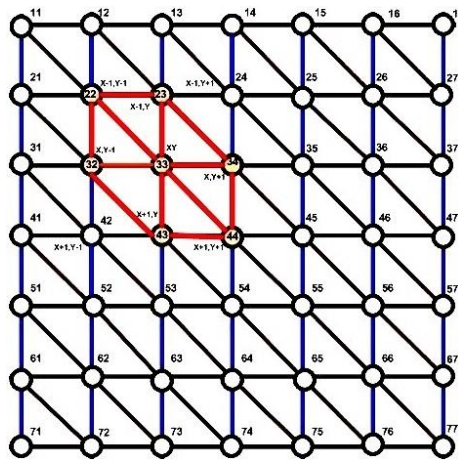
(b) Hexagonal Mesh type 2

Figure 1. Comparison of Hexagonal, and Diagonalize mesh with 2D mesh, Torus and Fat tree NoC topologies.

This paper is organized as follows: Section 2 discusses the conventional 2D hexagonal and 3D mesh NoC architectures. Section 3 discuss about the crossbase routing (CB) algorithm. Section 4 presents simulations and comparative analysis of routing algorithms for 2D Hexagonal and 3D mesh NoC. Section 5 discusses detailed analysis of experiment results and finally conclusion in section 6.
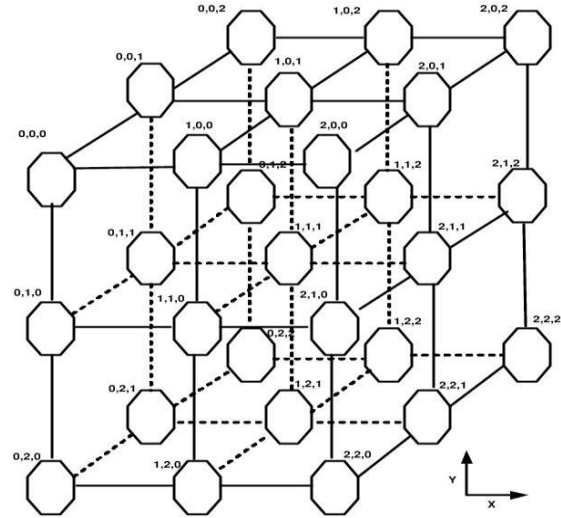

Figure 2. **Example 3D Mesh**

2.   TOPOLOGY COMPARISION

Comparative analysis of the 2D mesh, Torus and Hexagonal and Diagonalized mesh and Fat Tree topology is shown in Table 1 and summarized below:

1. **Node Degree:** Number of interconnections in a node is referred to as Node Degree of a topology. Node degree is an important factor in overall cost of NoC. In 2D mesh topology, corner tiles, or nodes have a node degree of three, as each corner node is connected with two adjacent nodes and a core or processing element. The intermediate tiles are connected with four adjacent tiles and a core. Similarly, in Torus, as all nodes are interconnected with each other the node degree of all nodes is same and equal to five.

2. **Diameter:** It is the length of the maximum shortest path between any two nodes measured in hopes in the network. For uniform $n \times n$ mesh, if D is the dimension and bisection width is $K = \sqrt{n}$, then diameter will be $D \times k - 1$, where k is the number of nodes in each dimension. Diameter for the Hexagonal mesh and Diagonalized mesh would also be almost equal to the mesh though using bisection link of distance $\sqrt{n}$.

3. **Average Distance:** For the uniform D dimensional mesh NoC, the average distance between two nodes can be stated as $D \times \sqrt{n}/3$. Similarly for Torus, it will be $n/2$. Because of symmetrical architecture of Hexagonal and Diagonalize mesh, average distance will be same as in mesh.

Table I. Comparison of Hexagonal, and Diagonalize mesh with 2D mesh, Torus and Fat tree NoC topologies.

Parameter	2D Mesh	Torus	Hexagonal Mesh (Six port)	Diagonalize Mesh (Five port)	K-ary n-Dim. Fat-Tree	3D Mesh (Six ports)
Degree	3-5	5	3-7	3-6	2k	4-7
Diameter	$2(\sqrt{n}-1)$	$\sqrt{n}-1$	$2(\sqrt{n}-1)$	$2(\sqrt{n}-1)$	$2d$	$3(\sqrt[3]{n}-1)$
Avg. distance	$\frac{2}{3}\sqrt{n}$	$\frac{n}{2}$	$\frac{2}{3}\sqrt{n}$	$\frac{2}{3}\sqrt{n}$	$\approx d$	$d(k-1)$
Wire cost	$2(n-\sqrt{n})$	$2n$	$2(n-\sqrt{n})+\sqrt{2}n$	$2(n-\sqrt{n})+\left[\frac{\sqrt{n}-1}{2}\right](\sqrt{n}-1)$	$O(nd)$	$O(n)$
Node scaling	$2n$	$2n$	$2n$	$2n$	n^2	n^3

4. **Wire Cost:** Considering an n system module interconnected in NoC, where the cost of each module does not change. Let, the unit length of wire is used between all the interconnected routers in NoC. Let, the number of interconnecting wires in each link is constant and equal to w. Then, the cost in terms of wire length of 2D mesh NoC, can be given as:

$$Wire\ cost = 2w\sqrt{n}(\sqrt{n}-1)\ (1)$$

Assuming unit wiring cost w for all links in the network, i.e., w=1, cost for different network topologies are listed in Table 1.

5. **Scaling:** Number of nodes increase with increase in dimension. This is referred to as scaling. $n \times n$ Hexagonal and Diagonalize mesh have similar structure and increase by factor of 2n. Whereas, k-ary n Dimension Fat tree increases with a factor of 2k.

2.1 Router Architecture for Hexagonal and Diagonalize mesh

Most important part of designing multiport NoC is the design of on-chip router. Complexity of router architecture increases with increase in number of ports. Symmetrical architecture of router is desired to ease fabrication process. Figure 4 shows typical router architecture for Hexagonal and Diagonalize mesh interconnection. Router consists of simple crossbar switch, input ports, output ports and programmable controller or arbiter. In multi-layer architecture, power consuming nodes are avoided to be stacked on top of each other to reduce the on chip thermal power. Usually power consuming nodes are put in the top layer and the low power consuming nodes are stacked on top of each other. 3D router design requires two additional physical ports to interconnect UP and DOWN links for inter-layer transfers in comparison to 2D design. Size of crossbar switch remains same for 2D hexagonal and 3D mesh. Number of input/output ports and flit length decides size and power consumption.

Hexagonal or Diagonalize mesh structure can be constructed if routers are designed and connected properly [8]. Figures 7 and 8 show hexagonal and diagonalize mesh structure using the 7-port and 6-port routers respectively. For the sake of simplicity, we assume cross links are routed across the IP CORE and two IP COREs can be fitted in the surrounding space by routers in Diagonalize mesh.

3. DEADLOCK FREE ROUTING ALGORITHM

Deadlock condition will occur if the flit is not able to find the next input channel to reach the destination or the queues in alternative output channels, supplied by the routing function, and are full. Due to this, the routing function $R(d_i, head(c_i))$ may not forward header flit to any adjacent output channel and data and tail flits get blocked as their header is in a full queue in the next channel. In a deadlocked situation, no header flit of any message can reach its destination.

3.1 Strategy for Deadlock Free Routing

For designing of deadlock-free routing algorithms for NoC, following assumptions are made [9]:

1. A node can generate messages asynchronously destined for any node in network and at any rate.

2. As the message arrives at its destination it is eventually consumed.

3. A node can generate message of any arbitrary length but it must be longer than the length of single flit.

4. When an input buffer of a router port accepts the first flit using wormhole routing, it must accept all the remaining flits of the same message before accepting any flit from any another message.

5. Buffer size in all the input ports must be same.

6. An available buffer should liaise only among the requesting messages and avoid arbitration between messages in waiting.

7. Buffer must entertain flits belonging to the same message only. Moreover, it must be emptied before accepting any other flit from any adjacent node.

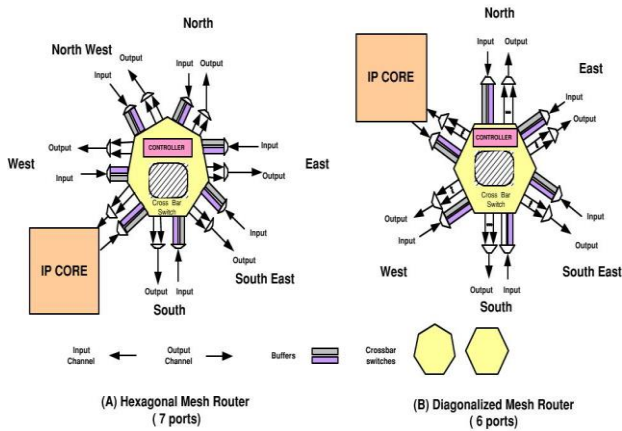8. For an adaptive routing the path taken by the message would depend of availability status of output ports.

Figure 3 Router for (a) Hexagonal mesh (b) Diagonalized mesh

Adaptive routing function R in a connected network is deadlock free if:

1. An order between the channels is established to route a packet, such that, there is no cycle in its channel dependency graph G_d^s.

2. There exists a subset of channels c ⊆ C such that, there is no cycle in its extended dependency graph $G_d^s E$ of the connected network.

3. Virtual channels can be added to provide more paths between all source-destination pairs. The added virtual channel's sub graph is also acyclic to ensure deadlock free routing.
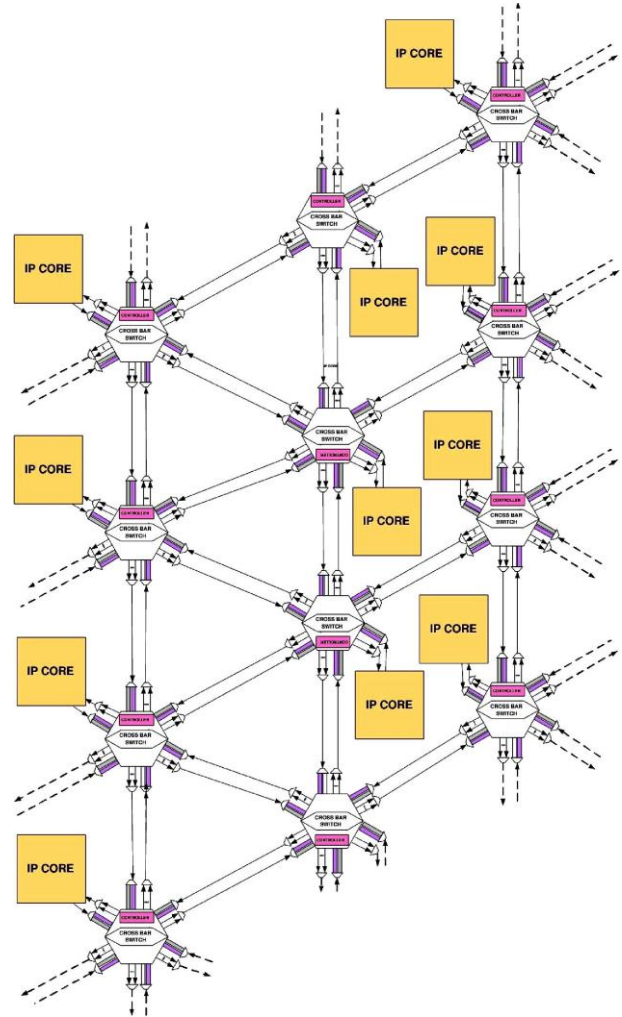

Figure 4 Interconnected Diagonalize mesh NoC using 6 ports

Suppose that there is a deadlock configuration for routing function *R. Let $c_i \in C$* be a nonempty queue such that there are no channels less than c_i with a nonempty buffer. If c_i is minimal, then the flit at the top of buffer can reach its destination in a single hop and there would be no deadlock. Otherwise, using channels less than c_i, the flit at the buffer head of c_i can advance and there is no deadlock.
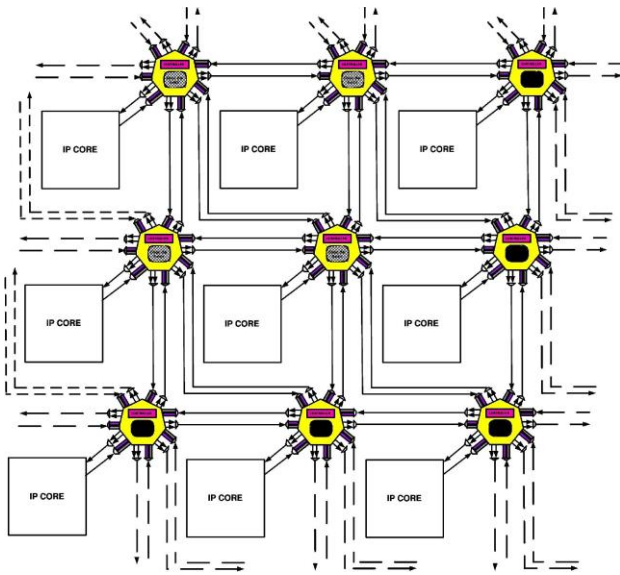
Figure 5 Interconnected Hexagonal mesh NoC using 7 ports

3.2 Adaptive Routing

Definition: For a network graph $G = (N, C)$, where, N is the set of nodes and C is the set of communication channels, an adaptive routing algorithm $R = (R_n; n \in N)$ for each $n \in N$ can be defined as a subset of routing algorithms $R_n = G_s^d$ where, $d \in N - n$ applied to subset of graph satisfying the following conditions [10]:

1. In a given graph $G = (N, C)$, consisting of input channel c and output channels $\tilde{c} \in C$ such that $c\tilde{c} \in E(G_s^d)$, i.e., all the edges connecting the channels from source to destination. A packet from source s will start from output channel $\tilde{c}$ towards destination d using the path P as a sequence n_1, (s_1, a_1), n_2, (s_2, a_2), n_3, (s_2, a_3) ... n_{k-1}, (s_{k-1}, a_k), where, n_i are channel nodes and (s_i, a_{i+1}) are channels, for $1 \leq i \leq k - 1$.

2. If a path between source to destination exists via input and output channel, such that, $c\tilde{c} \in E(G_p^{e(\tilde{c})})$ where, $e(\tilde{c})$ is the output channel edge, then a packet p will reach destination d using routing algorithm Rp, subject to the condition that there is no deadlock in the channel dependency graph.

Adaptive routing algorithm is capable of handling congestion without creating any deadlock condition. In congestion, a packet has to wait till availability of a link, through which it can be routed. One major design criterion of an adaptive algorithm is that it should be deadlock free. This is ensured either by restricting turns or ensuring that channel dependency graph has no cycles [12]. Adaptive routing may follow a non-minimal path.

Restricted turns in routing ensure deadlock free routing but, the number of shortest paths the algorithm allows from source to destination also known as (degree of adaptiveness) may vary.

Routing algorithm decide channel selection to route the packet from a source to destination. Routing strategy must be easier to implement and should comply with low latency and better throughput. Wormhole technique is the most suitable for NoC, but may cause deadlock or live-lock conditions. Breaking the cycle in channel dependency graphs will make deadlock free routing. ZXY and Crossbase routing algorithms are deadlock free without any virtual channel requirement as their prohibited turns do not constitute any cycle in the dependency graph and extended channel dependency graphs [13].
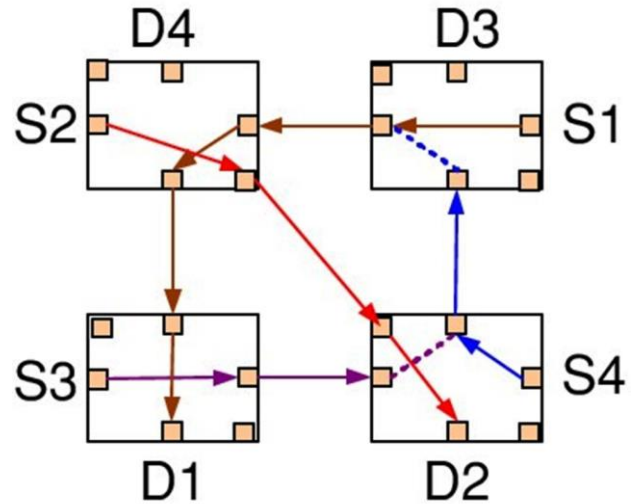

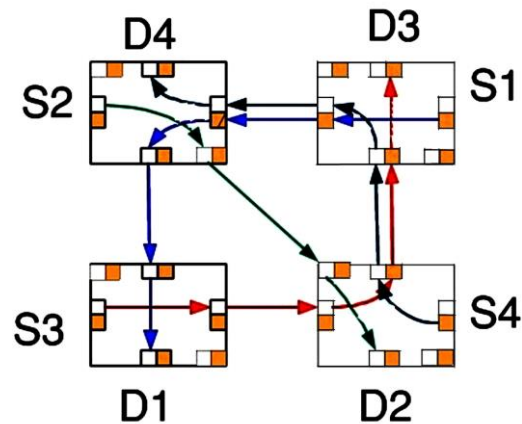**Figure 6 Crossbase routing without using VC's**

Figure 7 Crossbase routing using VC's

Proper use of virtual channels increases the efficiency of routing. More paths are available to route the packets in crossbase with VCs. Figure 6 and 7 shows crossbase routing without and with Virtual Channels (VC). Two virtual channels are sufficient for the deadlock free routing in any Hexagonal, Torus, and Mesh topology with the following restrictions:

- Use first VC, if destination D lies to the East of source S.

- Use second VC, if destination D lies to the West of source S.

- Use both VC, if destination D lies North or South of source S.

- Use all VCs for all horizontal links.

In ZXY, routing is restricted by routing a packet in slice first, then moves along the rows and then move along the column toward destination. Figure 2 shows an example of ZXY routing.

3.3. Crossbase Routing Algorithm

Crossbase routing is developed to use the diagonal link available for the interconnecting paths between the nodes. Additional path provides shorter link between the nodes as packets can move directly to another node bypassing the node in same row or column thus reducing delay. If the links are available between the sources to destination path, preference is given to access the diagonal link provided it has not been reserved; otherwise normal XY routing strategy is adopted. Detailed algorithm is presented in Algorithm 1.

4. Experimental setup

Routing algorithms are initialized and simulation experiments are made over NoC, using NOXIM real time simulator [14] written in SystemC. Different traffic patterns like random and transpose traffic used for simulation over the $4 \times 4 \times 4$ 3D mesh NoC and 8×8 2D hexagonal mesh network for the random data. Flow control unit (flit) of size 10, having two header payload and eight bytes data payload, are generated by increasing packet injection rate (PIR)[15] from 0.01 to 0.98 with 0.001 steps, over the proposed topology and simulated for 10000 cycles. Delay for increase in the length of cross-links is also considered while estimating global average delay.

Crossbase Routing for Hexagonal NoC

Require: S_x, $S_y\langle x, y\rangle$, C_x, $C_y\langle x, y\rangle$, D_x, $D_y\langle x, y\rangle$: Coordinates of source, current and destination node respectively node.

dirX, dirY, dir$\langle x, y\rangle$: horizontal, vertical and diagonal directions.

Ensure: Route from $\langle S_x, S_y\rangle \rightarrow \langle D_x, D_y\rangle$
Initialization
$C_x = S_x$, $C_y = S_y$

While (true) do
$diff_X = |Dx - Cx|$, $diff_Y = |Dy - Cy|$
 if $(diff_X == 0)$ OR $(diff_Y == 0)$
        then Return to IP CORE
end if
        if $(diff_Y > 0)$ then *dirY* = NORTH
        else *dirY* = SOUTH
        end if
if $(diff x > 0)$ then
        if $(Dy < Cy)$
                then *dir* = SOUTH EAST
        else *dirX* = EAST
        end if
else
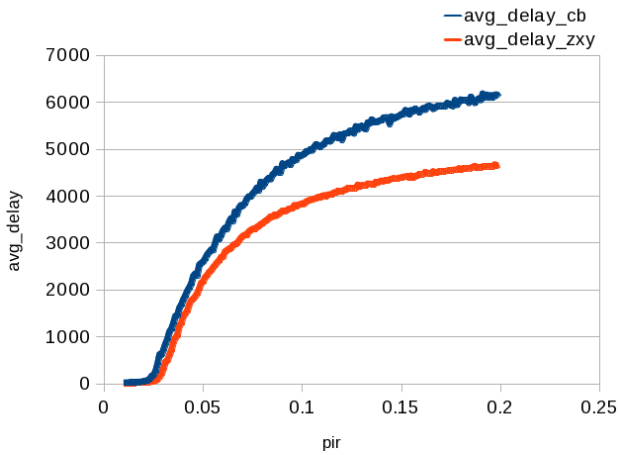   if $(Dy > Cy)$ then *dir* = NORTH WEST
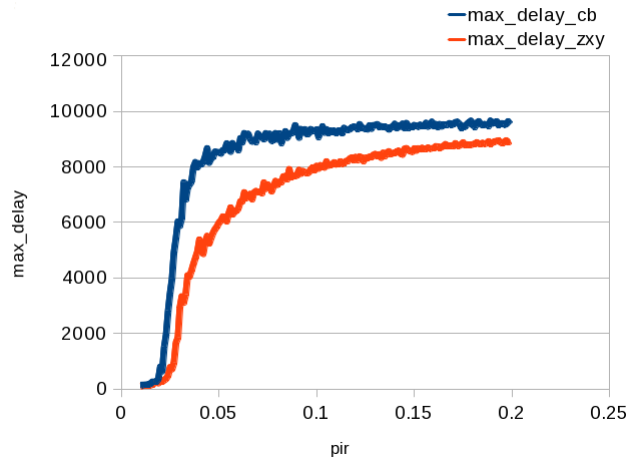   else *dirX* = WEST
        end if
end if
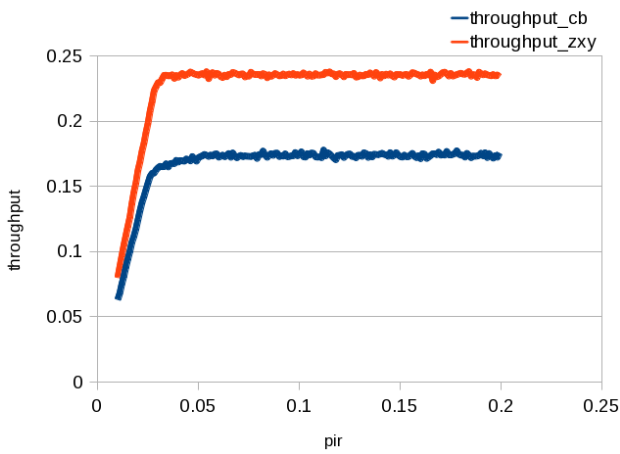Update $\langle Cx, Cy\rangle$
end while

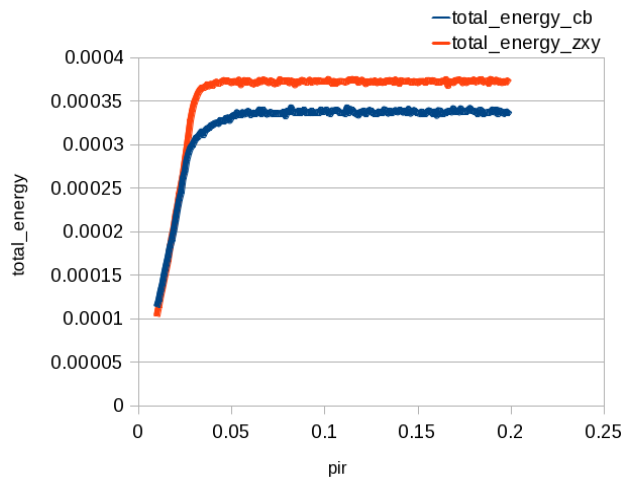**Algorithm 1 Crossbase routing algorithm for 2D Hexagonal mesh.**

(a) Average delay (in clock cycle) for Random traffic

(b) Maximum delay(in clock cycle) for Random traffic

(c) Throughput for Random traffic

(d) Total Energy for Random traffic

Figure 8. Effect on performance metrics with increase in Random traffic for CB and ZXY routing

(a) Average delay(in clock cycle) for Transpose traffic

(b) Maximum delay(in clock cycle) for Transpose traffic

(c) Throughput for Transpose traffic

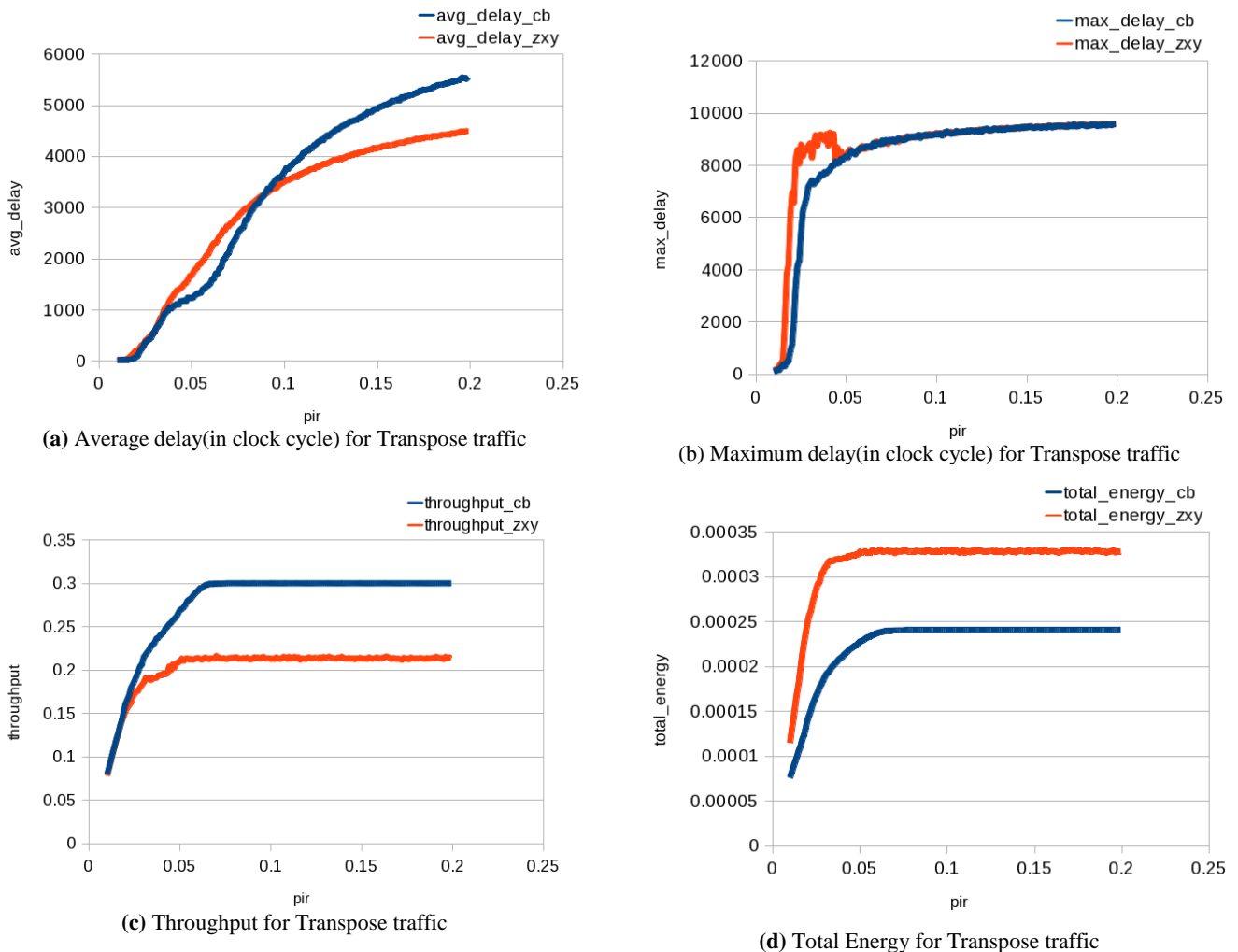**(d)** Total Energy for Transpose traffic

Figure 9. Effect on performance metrics with increase in Transpose traffic for CB and ZXY routing

5. Analysis of experimental results

Experiment for the same number of nodes, for 2D Hexagonal and 3D mesh NoC, is made for random and transpose traffic. As the congestion goes on increasing 3D NoC outperform 2D NoC. Average delay, maximum delay, throughput and total energy are the four performance parameters that are accounted in this work to evaluate the performance of CB and ZXY routing algorithms for random and transpose traffics on 2D Hexagonal and 3D mesh NoC. Figure 8(a) & 8(b) shows that, via ZXY routing less average and maximum delay can be achieved in random traffic as compared to CB routing at high congestion. While more realistic pattern are obtained in the transpose traffic, where the CB routing shows lower latency up to 5 percentage of Packet Injection Rate (PIR) as depicted by Figure 9(a). This is due to simplicity and available cross-link in the 2D NoC.

As CB routing is adaptive in nature, the maximum throughput can be achieved at higher congestion represented by the higher PIR in Figure 9(c). The saturated total energy gap between CB and ZXY routing for random and transpose traffic configuration remains 0.15×10^{-4} and 0.4×10^{-4} respectively as shown by Figure 8(d) & 9(d).

6. Conclusions

Our aim is to explore the 2D hexagonal mesh and 3D mesh to evaluate their performance using simple routing algorithm. For the small PIR the overall performance of 2D hexagonal mesh is better as compared to 3D, for the increased load condition the 3D mesh is suitable due to uniform architecture and equidistance of nodes. Power consumption is a critical issue in the NoC designing. Non stackable 2D architecture of Hexagonal mesh is better compared to 3D NoC, as it consumes significant low power, making it more suitable for NoC design. Thus we can state that CB routing gives more promising result

compared to ZXY routing technique. The routing and traffic sets configurations concluded in this paper will lead to achieve minimum latency with maximum throughput in NoC designing.

REFERENCES

[1] Vitor de Paulo and Cristinel Ababei, "3D Network-on-Chip Architectures Using Homogeneous meshes and Heterogeneous Floorplans." International Journal of Reconfigurable Computing, vol. 2010, Article ID 603059, 12 pages, 2010.

[2] Shan Yan; Bill Lin, "Design of application-specific 3D Networks-on-Chip architectures." Computer Design, 2008. ICCD 2008. IEEE International Conference, vol., no., pp.142,149, 12-15 Oct. 2008.

[3] Linder, Daniel H., and James C. Harden. "An adaptive and fault tolerant wormhole routing strategy for k-ary n-cubes." Computers, IEEE Transactions on 40, no. 1 (1991): 2-12.

[4] Chien, Andrew A., and Jae H. Kim. "Planar-adaptive routing: low-cost adaptive networks for multiprocessors." Journal of the ACM (JACM) 42, no. 1 (1995): 91-123.

[5] Pingqiang Zhou; Ping-Hung Yuh; Sapatnekar S.S., "Application-specific 3D Network-on-Chip design using simulated allocation." Design Automation Conference (ASP-DAC), 2010 15th Asia and South Pacific , vol., no., pp.517,522, 18-21 Jan. 2010.

[6] Weldezion, A.Y.; Grange, M.; Pamunuwa, D.; Zhonghai Lu; Jantsch, A.;Weerasekera, R.; Tenhunen, H. "Scalability of networkon-chip communication architecture for 3-D meshes."Networks-on-Chip, 2009. NoCS 2009. 3rd ACM/IEEE International Symposium, vol., no., pp.114,123, 10-13 May 2009.

[7] Ramanujam, R.S.; Bill Lin, "A Layer-Multiplexed 3D On-Chip Network Architecture. "Embedded Systems Letters, IEEE, vol.1, no.2, pp.50,55, Aug. 2009.

[8] Licheng Xue; Weixing Ji; Qi Zuo; Yang Zhang. "Floorplanning exploration and performance evaluation of a new Network-on-Chip." Design, Automation & Test in Europe Conference & Exhibition (DATE), 2011 , vol., no., pp.1,6, 14-18 March 2011.

[9] Jabbar, M.H. Houzet, D. Hammami, O., "Impact of 3D IC on NoC Topologies: A Wire Delay Consideration."Digital System Design (DSD), 2013 Euromicro Conference , vol., no., pp.68,72, 4-6 Sept. 2013.

[10] Harmanani, H.M. Farah, R., "A method for efficient mapping and reliable routing for NoC architectures with minimum bandwidth and area." Circuits and Systems and TAISA Conference, 2008. NEWCAS-TAISA 2008. 2008 Joint 6th International IEEE Northeast Workshop on NoC , vol., no., pp.29,32, 22-25 June 2008.

[11] Jingcao Hu; Marculescu R., "Exploiting the routing flexibility for energy/performance aware mapping of regular NoC architectures." Design, Automation and Test in Europe Conference and Exhibition, 2003 , vol., no., pp.688,693, 2003.

[12] Wen-Chung Tsai, Yi-Yao Weng, Chun-Jen Wei, Sao-Jie Chen, Yu-Hen Hu. "3D Bidirectional-Channel Routing Algorithm for Network-Based Many-Core Embedded Systems." Lecture Notes in Electrical Engineering, Springer Netherlands, Volume 260, 2014, pp 301-309.

[13] Jose Duato, "A New Theory of Deadlock-Free Adaptive Routing in Wormhole Networks." Proceedings of the 8th International Workshop on Interconnection Network Architecture: On-Chip, Multi-Chip, INA-OCMC '14, 2014

[14] Palesi, Maurizio, Davide Patti, and Fabrizio Fazzino. "Noxim." (2010).

[15] Tedesco, L.; Mello, A.; Garibotti, D.; Calazans, N.; Moraes, F., "Traffic Generation and Performance Evaluation for Mesh-based NoCs." Integrated Circuits and Systems Design, 18th Symposium, vol., no., pp.184,189, 4-7 Sept. 2005.

[16] Ahmed, Mushtaq, M. S. Gaur, and Vijay Laxmi. "Cross-Base Routing over Diagonalized Mesh Network on Chip." In proceeding of: 2010 IRAST International Congress on Computer Applications and Computational Science (CACS 2010), Volume: 978-981-08-6846-8

Ravindra Kumar Saini received B.E. in Computer Science and Engineering from University of Rajasthan, Jaipur in 2007. He has completed his M.E. in Computer Science and Engineering from BITS, Pilani in 2010. He worked as software engineer for embedded system during year 2010-2011. He is pursuing Ph.D. from MNIT, Jaipur. His research interest lies in the field of fault tolerant routing for Network-on-Chip.

Dr. Mushtaq Ahmed received his Ph.D. degree in Computer Enggineering from MNIT, Jaipur in 2012. He is faculty in Department of Computer Science and Engineering and teaching various courses like Computer Networks, Advanced Microprocessors, Computer Architecture, Logic System Design, Operating Systems, and VLSI Algorithms etc. for the past 15 years. His research areas are Network-on-Chip, Wireless communications, Cloud computing, Embedded system design, Image processing. Presently he is guiding research in Network on Chip and other area of interest.