Journal of Theoretical and Applied Information Technology
15th August 2018. Vol.96. No 15

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

PERFORMANCE EVALUATION OF VARIOUS ON-CHIP TOPOLOGIES

BANSIDHAR JOSHI and MANISH K THAKUR
Department of CSE/IT, Jaypee Institute of Information Technology, Noida, India – 201309
E-mail: bansidhar11@gmail.com, mthakur.jiit@gmail.com

ABSTRACT

With shifting of the present world towards automation, computer intensive applications are gaining popularity in day-to-day life, resulting into demands for highly scalable and reliable Network-on-Chip (NoC) based architectures along with low power consumption. NoCs are an exertion to impoverish the approach of huge networking architectures and implement them with embedded System-on-Chip (SoC) sphere. In reliable NoC based architectures, efficiency of the system primarily depends upon the interconnected structure or topology. The topologies are critically analyzed on the basis of various performance factors like throughput, latency, scalability, cost of traversal *etc*. The availability of fixed sized buffer queues at intermediate routers in the interconnection network topology holds a tight constraint over managing the incoming data packets from neighboring routers. With the traffic being heterogeneous and application-centric, the overlay architecture must be chosen wisely to deal with possible congestion scenarios. This paper analyzes the relative conduct of topologies and their altered versions and evaluates the prime performance parameters, *viz*. throughput and latency under various buffers' queue management schemes. Simulations (NS2) on two different nodes configuration, 16 cores and 64 cores shows Torus and Mesh of Tree as the favorable topologies respectively under diversified traffic.

Keywords: *Network-on-Chip, System-on-Chip, Topology, Interconnection Network, off-chip, on-chip*

1. INTRODUCTION

Since decades, distributed or parallel systems are playing major role for such applications which requires large computations. These systems involving various processing elements connected through interconnection network, *viz.* bus, mesh, torus, *etc.* often used to parallelize the tasks involved in the applications requiring large computations.

Conventional off-chip based architecture where processing elements are connected through various traditional inter-connects are becoming inapt to fulfill high throughput, scalability, low power consumption demands of today's computer chips due to their increased hardware complexities and unprecedented delays [1-3]. Limitations of off-chip architecture over various factors, *viz.* fault detection, security, delay, scalability, *etc.* is presented in Figure 1. Finding a workable solution to these problems and limitations with off-chip architecture is currently a major research area in the domain of computer architecture.

With advancements in semiconductor based technology the growing computational demands of the modern society led towards the development of on-chip networks or architecture. With relatively reduced size and power consumption, on-chip based architecture is now a day gaining popularity over off-chip architecture. System-on-chip (SoC) and Multi-Processor-System-on-Chip (MPSoC) are two on-chip based architectures proposed in recent years [2][4].

While SoC integrates a system involving various electronic components *viz.* microcontroller or microprocessor forming various Intellectual Property (IP) cores, graphics processing unit, and peripheral devices on a single integrated chip, MPSoC uses multiple heterogeneous processors or processing elements [4]. Various components of these on-chip networks or architecture are connected through different interconnection networks, *viz.* bus and shared bus. In bus-based topology, communication among the devices takes place on bus links whereas a set of wires is used in shared bus topology. Because of its low-cost as compared to bus-based system and simple control characteristics, the shared bus network architecture is more preferred for the communication between all the integrated processing units on on-chip architecture.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Characteristics	Effects/Limitations
Single Point Failure	Entire network fails
Traffic	Not suitable for networks with heavy traffic
Fault detection	Difficult to troubleshoot the fault in case of detection
Work stations	With increasing number of work stations, efficiency of the whole network decreases
Security	Security is very low as all the nodes encounter the signals transmitted from the source node.
Cost	Maintenance cost is very high
Wiring	Increasing the heating problem and also result in large wiring delays.
Scalability	Less scalable as only one processing unit can utilize the bus at one time.
Bandwidth	Bandwidth is inversely related to number of nodes. Higher the number of nodes, lesser is the bandwidth available.
Delay	Delay is directly proportional to the number of nodes. With increase in the number of nodes, delay also increases as data has to travel larger physical path
Energy	Due to use of wires, energy consumption also increases.
Performance	Performance is inversely related to the number of nodes and wiring.

Figure 1: *Limitations of conventional off-chip network architecture* [3][5]

With increasing number of IPs, performance of the SoC (built using bus-based interconnection architecture) may not be sufficient enough to meet the requirements of different applications. Obvious reasons behind this is the incapability of the bus-based interconnection architecture to provide required bandwidth, latency, and power consumption with increasing number of IPs [6-7].

In last decade, Network-on-chip (NoC) emerged as the remedy for such communication bottleneck. Implemented with SoC sphere, NoC provides a solution to the aforementioned limitations of bus architecture [8-10]. NoC also surpass the problem of large wiring delays in bus architecture. NoC provide an alternative to meet the communication requirements with higher efficiency and low power consumption as compared to bus based SoC.

In NoC, data is sent in form of the packets from source to destination selecting routing algorithm at the switches [11]. With the help of NoC architecture, an extensive count of buffers can be incorporated on a solitary chip. Various contributions made in recent years showing that NoC architecture has better scalability and power efficiency than bus based SoC architecture. A qualitative comparison between bus based SoC and the NoC architecture is presented in Figure 2. The NoC architecture consists of network elements and network interfaces (NI). The data in packets traverse the system components and the system interfaces. Utilizing switching components, removes the biasing of the clock flag, which is available in SoC [12].

NoC based system architecture is described by the following primary attributes [3]: Topology – it describes the layout of the system components; Switching – it describes the manner in which information is exchanged between different ports of the NoC based system; Routing – it finds the route of the packet to exchange the information; Flow Control and Deadlock prevention – it assigns routes to the packets with minimal resources (link bandwidth, buffer *etc.*). With utilization of NoC in the networking groundwork, we got the bisection of calculation and network connections**.** NoC emerges as reusable and adaptable in exchanging information [7].

With different architectural characteristics *viz.* the cardinality, diameter and bisection bandwidth specifying each topology (to connect the cores and other components in NoC architecture), the communication link(s) established from source to destination node(s) might cause congestion in the intermediate routers containing limited buffer capacity. These dynamic congestion scenarios sometimes lead to head-of-line blockage [13-14] of packets in these buffers, resulting to the packets being dropped primarily.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Bus based SoC	On Chip Networks (NoC)
As the number of devices connected increases, capacitance increases and hence performance decreases	All the devices are connected in point to point fashion hence no performance reduction
Delays can cause blockages	Routing decisions are distributed
Bounded bandwidth and distributed among all the devices	Bandwidth is directly proportional to the size of the network
Latency increases with number of devices	Inherent decisions increase delays
Simple architecture	Complex architecture

Figure 2: *Qualitative comparison between bus based SoC architecture and NoC architecture* [8-10]

In worst scenarios, the inefficient usage of buffer queues leads to the deadlock or livelock of multiple packets moving in different directions. Ultimately, these undesirable conditions put impact on various QoS parameters of NoC based topologies like performance, real time communication, and reliability [5][15].

Further, the traffic congestion in the NoC architecture is often controlled by various queue management techniques and flow control. Since the topology represents the first and foremost aspect while selecting the communication model based over NoC, it is imperative to test various queue management schemes and evaluate them for various NoC-based topologies with different node configurations for finding out the feasibility with respect to various QoS parameters.

In this paper, we evaluated the performance of various interconnection topologies based on various queue management techniques and performance parameters, *viz.* throughput and latency. Along with latency and throughput, the schemes used for the management of buffers' queues inside the routers in NoC framework make a significant impact on various performance parameters.

This paper is further organized as follows: Section 2 details with various topologies applicable for NoC; in Section 3 we present the performance evaluation parameters; discussion over the comparative performances is presented in Section 4; paper is concluded in Section 5.

2. TOPOLOGY

In general, topology in networking refers to the interlinking of the constituent parts of a networked system on a given space or area, *i.e.* it ascribe to the physical layout of the devices in the network graph. It defines static arrangements of the channels and the nodes (such as switches or routers) in an interconnection network and thus has effect on the performance of the network.

While determining a network topology, tradeoff between generality and customization or personalization is an important issue. The generality of a topology refers to the re-usability and the scalability of the network architecture where as the customization or personalization is aimed for the performance of the system [13][16]. Cost is another major issue, while choosing a network topology. A network topology is opted, if it fulfills the requirements of the traffic at reasonable costs. One of the major issues in NoC architecture is to choose the optimal topology to fulfill the throughput and latency requirements for an application at low energy/power and area costs. This paper analyzes various topologies having different characteristics and configurations in subsequent sections.

On the basis of connections in the network, topologies for NoCs are broadly classified into two categories, namely: direct topologies and indirect topologies [13][16] and discussed subsequently in Section 2.1.

Network topologies can also be classified on the basis of the links between different elements *i.e.* routers, namely: regular topologies and irregular topologies [7][13][16] and discussed subsequently in Section 2.2.

2.1 Direct and Indirect Topologies

In direct topologies, all the nodes in the network have the functionality of both – a terminal (*i.e.* processing element) and a switch (*i.e.* routing element). As shown in Figure 3, every node is directly connected with a subset of different nodes. Nodes are comprised of buffers and routers. It handles the transfer of packets among the nodes.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Hence, direct networking topology- based systems are otherwise called switch-based systems. Each router is connected directly to the routers of its neighbors.

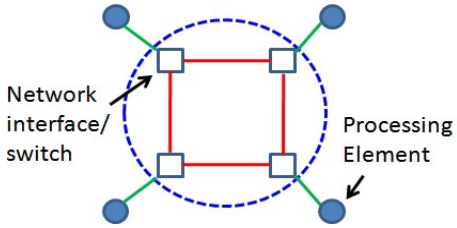


Figure 3: *Direct topology*

Some of the often-used direct network topologies are Mesh and Torus, *etc.* Figure 4 and Figure 5 present 8×8 Mesh and 4×4 Torus interconnected network topologies respectively.

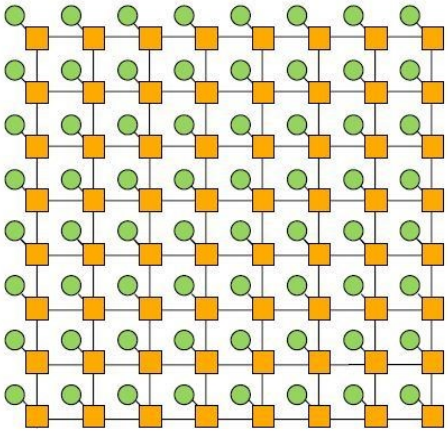


Figure 4: *8×8 Mesh as an example of direct interconnected network where rectangular boxes represent the switches and circles represent the processing elements*

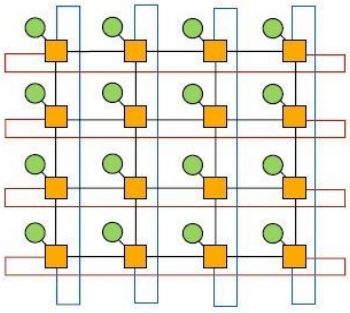


Figure 5: *4×4 Torus as an example of direct interconnected network where rectangular boxes represent the switches and circles represent the processing elements*

In direct topologies, the intrinsic conciliation is between availability and cost. In direct topologies, with increase in number of nodes, bandwidth available for the communication in the network also increases. Thus, direct network topologies are used to build large-scale parallel systems [13].

On the other hand, in indirect topologies, nodes behave either as a terminal or a switch. As shown in Figure 6 a switch acts as an intermediate for the communication between two nodes. Nodes are connected to exterior switch, and switches have direct connection to different switches [16].

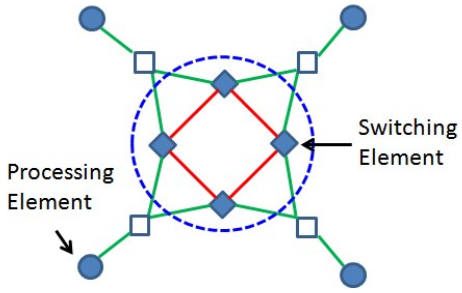


Figure 6: *Indirect topology*

Some of the often-used indirect network topologies are SPIN, Crossbar topologies, and Fat Tree, *etc.* Figure 7 presents an example of indirect interconnected network made using Fat Tree topology.

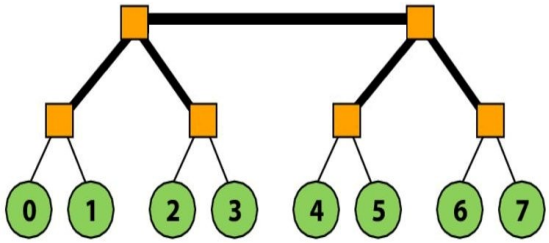


Figure 7: *Fat Tree as an example of indirect interconnected network where rectangular boxes represent the switches and circles represent the processing elements*

2.2 Regular and Irregular Topologies

In regular topologies, all nodes are similar with respect to the quantity of ports interfacing with other nodes in the system. 2D Mesh and Torus are some of the examples of regular topologies [7].

An irregular topology-based architecture is comparable to the fully connected network architecture excluding the demand of connecting a

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

computer to every computer is eliminated. System are made more application and specification oriented by implementing irregular topologies. To achieve this, several constraints are to be taken into consideration during the formation of network layout. In return, parameters such as power consumption, chip area, and number of nodes in the architecture are optimized. In irregular topologies, different connections may be represented by the nodes, which are generally defined on the basis of the application.

In this paper, performance of the Mesh, Torus, King Mesh, M Mesh, Hypercube, Ring, Star, Fat Tree, and Mesh of Tree based topologies (regular) have been evaluated over various performance parameters, *viz.* throughput, latency, queue management techniques *etc.* These performance parameters have been individually discussed in subsequent section.

3. PERFORMANCE PARAMETERS

Performance of the network topology depends on various parameters including the characteristics or properties, *viz.* diameter, bisection width, degree *etc.* of a topology. Often, it is desired to have a scalable topology so that the complexity of the network communication gets least increased after increasing the number of processing elements in NoC based system.

Because of difference in nature and design of each network, performance of a network can be measured using various parameters. These performance parameters include throughput, latency, delay, packet delivery ratio, and packet loss, *etc.* [7][17-18].

Further, various queue management techniques are also responsible to degrade the performance of a network when multiple sources led heavy traffic, causes congestion in the intermediate routers' buffers. We discuss following performance parameters in subsequent subsection: throughput, bandwidth, latency, and queue management techniques.

Subsequently we discuss the characteristics of the network topologies and different performance parameters.

3.1 Properties of Topology

Several properties are there which characterizes a topology and responsible for the effectiveness of the NoC based system implemented using these topologies. These properties include diameter, bisection width, degree, link complexity, articulation points and bridges, *etc*. and responsible to build scalable and stable system.

A topology is called as stable, if, communication is possible among all the nodes even if some nodes or links fail. Further, a topology is scalable if communication complexity does not increase much after the addition of nodes in the existing NoC system built over that topology.

Diameter of a network topology refers to the largest distance between any two nodes in a given topology. Lower diameter of a topology reduces the communication complexities (*viz*. message delay, troubleshooting, *etc*.) between arbitrary pair of nodes.

On the other hand, bisection width describes the stability of the network. It refers to the minimum number of edges required to be removed or cut from the network so that the network gets divided into two halves. Higher bisection width of a topology is desirable to build more stable network.

Degree of a node in the network topology is the number of edges or links connecting that node to other nodes. It is desirable that the degree of all the nodes in the topology should constant (symmetrical nodes) and independent of the network size. Constant degrees of the nodes need least effort to add new nodes and support to easily build more scalable system.

Link complexity is defined as the number of connects or links in the topology [18]. With increasing the count of links or connects, diameter of the network decreases (this facilitates the better communication among nodes), whereas, complexity of the network or hardware increases (resulting into several overheads, *viz*. area, cost, *etc*.). Inter node communication time is least in case of fully connected topology but this topology is having highest link complexity and hence least preferred network topology.

Articulation point in a network topology refers to a node which increases the number of components after the removal of that node and associated edges or links. On the other hand, if there is no path left between two nodes after removal of the edge or link connecting them, then that edge or link is called as bridge. Presence of articulation point and bridge make a topology highly unstable.

Diameter, bisection width, degree, link complexity, articulation point and bridge are the

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

major properties responsible to make a topology stable and scalable. Increasing the number of nodes increases the diameter of ring topology whereas it is constant in star based topology; however, star topology is highly unstable due to the presence of articulation point. In context of scalable topology, fully connected network is highly un-scalable whereas, star, ring, *etc*. are scalable topologies.

3.2 Throughput

Throughput is the rate in bits per second at which network accepts the data. It is the measurement of the channel bandwidth and measured as the ratio between total number of data packets received and transmission time [17].

Throughput can be considered similar to the number of highway lanes as follows: higher the number of lanes on highway, higher the capacity of highway to accommodate more traffic. Similarly, in a network higher the bit rate, faster the transfer. A network with higher bit rate will have faster data transfer and maximum throughput occurs when a channel is saturated.

Further, throughput depends on various other parameters such as available bandwidth, available signal-to-noise ratio, hardware utilization, data loss, protocol used, routing algorithm used, buffering, and flow control.

3.3 Bandwidth

Bandwidth is defined as the maximum amount of data that could be sent over a channel, *i.e.* it is used to measure the speed of the network. It is measured in bits per second. It constitutes to the net potential of the network. For better performance, bandwidth of the network should be high. By increasing number of cores, we can increase the bandwidth of the network.

3.4 Latency

Latency is defined as the time required from the inception of the packets to the reception at the destination node. Latency is measured in the units of time. Latency can be computed in following parts [13]:

(1) Head Latency (T_h) – it is the time needed by the head to pass over the network; and

(2) Serialization latency (T_s) – it is the time needed by an L length packet to pass over a channel of bandwidth B.

Hence, the latency is computed as the time required in both parts ($T_h + T_s$), *i.e.* $T_h + (L/B)$. Additionally, processing delay (T_c) at each router to accommodate the incoming packets from the neighbor routers is also incurred in computing the latency. Thus, the latency is finally computed as ($T_h + T_s + T_c$).

3.5 Packet Loss and Packet Delivery Ratio

While transmission of the packets, some of the data packets sent by the source node to different destination node (s) may not reach the destination node. This is referred as packet loss. Packet loss is inversely proportional to the throughput. Higher the packet loss ratio, lower is the throughput. Packet loss is detected by application layer protocols such as TCP protocol. Errors and the network overloading are the main reasons for packet loss in the network. Packet loss can be calculated as the difference between the count of the packets delivered by the source and count of the packets received by the destination.

3.6 Queue Management Techniques (QMT)

Performance of a network topology can be measured in terms of various queue management techniques [19-21] which are responsible to manage the flow of data packets by managing the buffers of the intermediate routers between different sources and destinations. These fixed size buffers in the routers holds a tight constraint over managing the incoming data packets from the neighboring routers.

Following techniques of the queue management is used in this paper to test the performance of the different interconnection network topologies forming the NoC architecture.

Drop Tail: This is the most basic, easy to implement and default approach applied for queue management. In drop tail, the queue management is done by dropping the subsequent arrived packets when there is no space left in the router's buffer. This technique suffers from buffer overflow which reduces the throughput significantly.

Random Early Detection (RED): RED is used to halt the queue being completely utilized by dropping packets arbitrarily (statistical probabilities) and acknowledge the sender to down turn before queue get completely filled. Here all arrived packets are accepted, if the buffer is empty, whereas, all arrived packets are dropped, if the buffer is full. Depending on the availability of the buffer space between empty and full, acceptance and dropping of some packets are randomly decided.

Stochastic Fair Queue (SFQ): SFQ generally doesn't allot a queue for every session, it first calculates the number of queues using hashing

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

algorithm, and then distribute traffic over those queues.

Random Exponential Marking (REM): Random exponential is used to maximize the delay and packet loss. It checks whether the client rate is contrasted and system limits and sum up the active links.

Fair Queuing (FQ): FQ grants various packets flow to relatively share the connection limit.

Deficit Round Robin (DRR): DRR is used to treat dissimilar packets without calculating their average sizes.

4. PERFORMANCE EVALUATION

In this section, we present the experimental analysis to evaluate the performance (throughput and latency) of various interconnection network topologies using different queue management techniques discussed in previous section.

To analyze the throughput and latency for different queue management techniques, following topologies had been simulated with 16 cores or nodes and 64 cores or nodes on NS2 simulator [8][17]: Mesh, Torus, Hypercube (HC), Fat Tree (FT), Ring, Star [18], King Mesh (KM) [22], Multi Mesh (MM) [23], and Mesh of Tree (MOT) [24]. Simulation environment for each experiment consists of the packet size of 500 bits at the packet injection rate 0.005 with the assumption of the packet delivery ratio close to 100%.

Under discussed simulation environment, achieved throughput and latency for different topologies and queue management techniques have been plotted in Figure 8 (QMT as Drop Tail), Figure 9 (QMT as RED), Figure 10 (QMT as REM), Figure 11 (QMT as SFQ), Figure 12 (QMT as FQ), and Figure 13 (QMT as DRR). Subsequently we discuss the observations for the topologies simulated with 16 cores or nodes and 64 cores or nodes.

Observations for the topologies simulated with 16 cores or nodes – When different interconnection network topologies simulated with 16 nodes and lesser packet size, values for latency and throughput (Figure 8 to Figure 13) for all the queue management algorithms (Drop Tail, RED, REM, SFQ, FQ, and DRR) are almost same.

Overall, Star based topology produced maximum throughput (around 800) and latency (around 13) for all the queue management techniques, but presence of articulation points and bridges make it less reliable. Further, ring based topology also produced optimal throughput but due to link failure and increased network latency, ring and star based topologies are not recommendable for NoC architecture.

Further, Torus, King Mesh and Hypercube achieved the near optimal throughput (between 796 and 797) and considerable latency (between 19 and 27). Architecturally, these topologies are highly fault tolerant. Unlike, Mesh and Torus, besides the movements in rows and columns, the packets in King Mesh topology can traverse in diagonals (like the movement of king in chess board) too, but it has higher link complexity and higher node degree (a node is connected with eight neighbouring nodes), *i.e.* higher connectivity of the routers for linking the cores in NoC architecture.

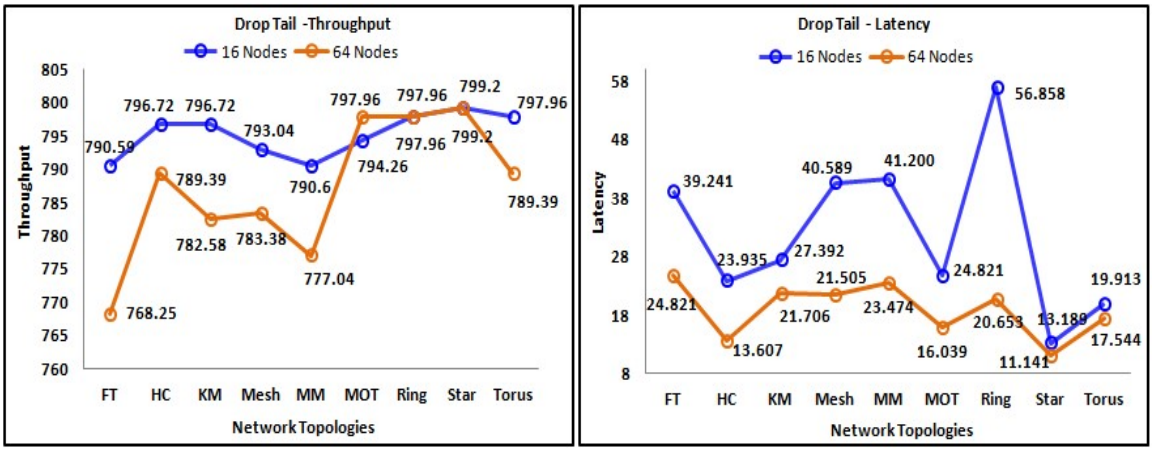


Figure 8: *Achieved throughputs and latencies for various topologies, FT (Fat Tree), MOT (Mesh of Tree), HC (Hypercube), Mesh, KM (K Mesh), MM (M Mesh), Ring, Star, and Torus with QMT as Drop Tail*

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

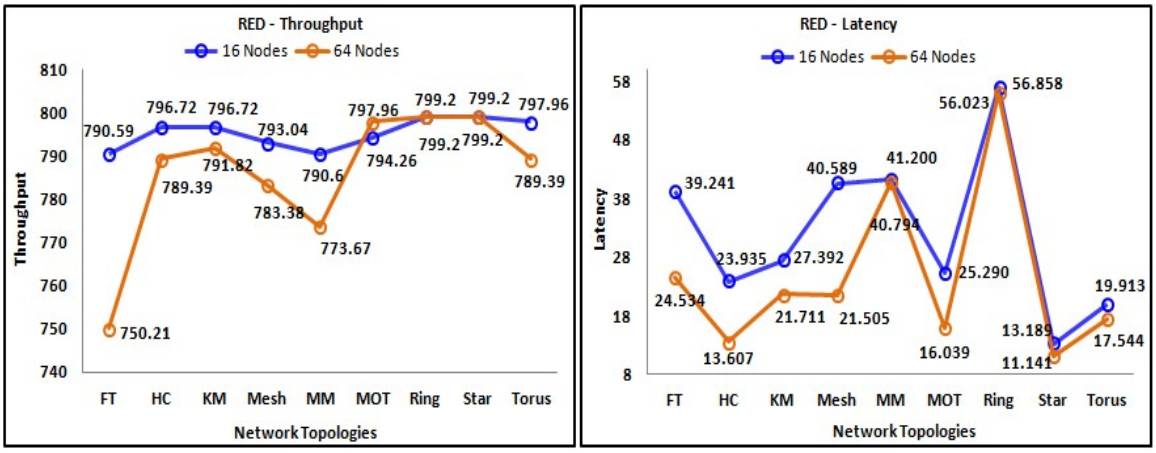


Figure 9: *Achieved throughputs and latencies for various topologies, FT (Fat Tree), MOT (Mesh of Tree), HC (Hypercube), Mesh, KM (K Mesh), MM (M Mesh), Ring, Star, and Torus with QMT as RED*

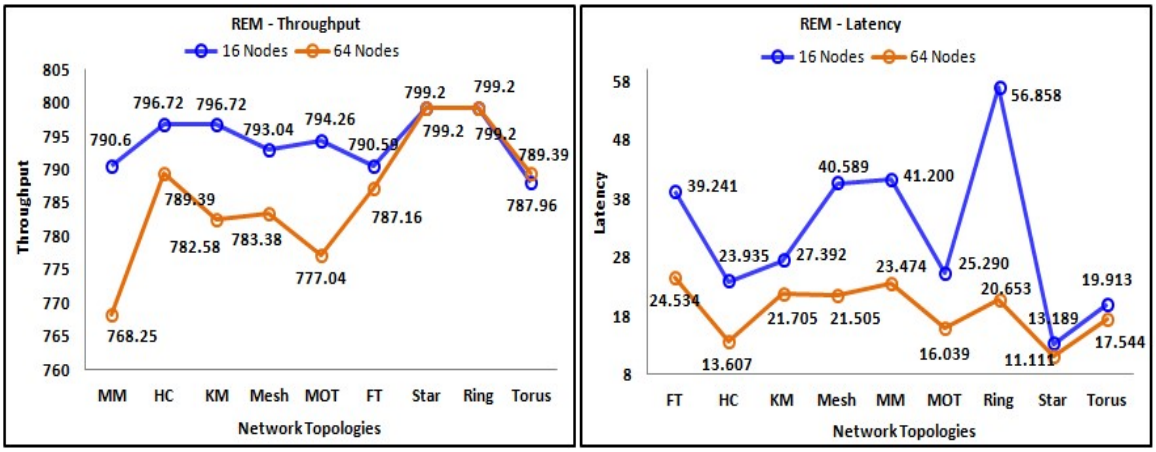
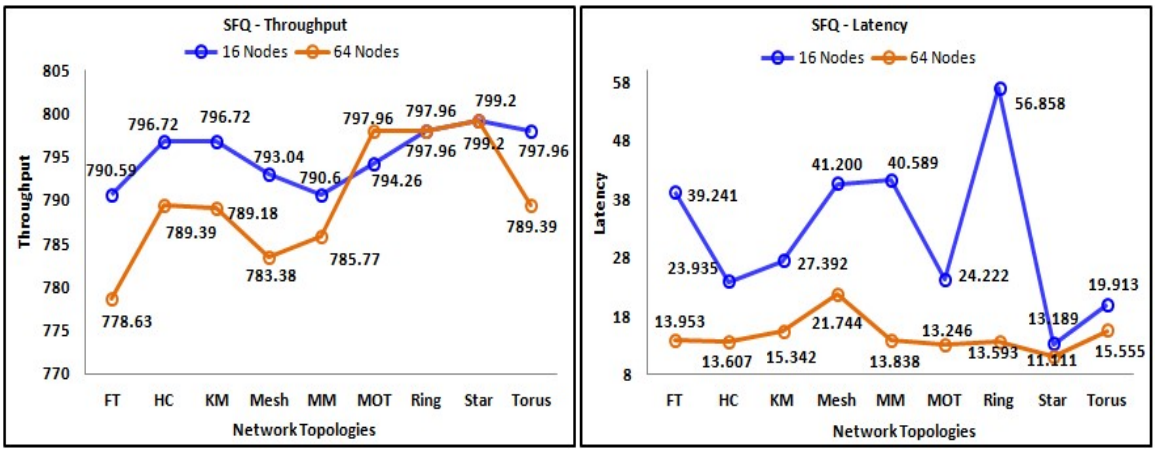


Figure 10: *Achieved throughputs and latencies for various topologies, FT (Fat Tree), MOT (Mesh of Tree), HC (Hypercube), Mesh, KM (K Mesh), MM (M Mesh), Ring, Star, and Torus with QMT as REM*



ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Figure 11: *Achieved throughputs and latencies for various topologies, FT (Fat Tree), MOT (Mesh of Tree), HC (Hypercube), Mesh, KM (K Mesh), MM (M Mesh), Ring, Star, and Torus with QMT as SFQ*

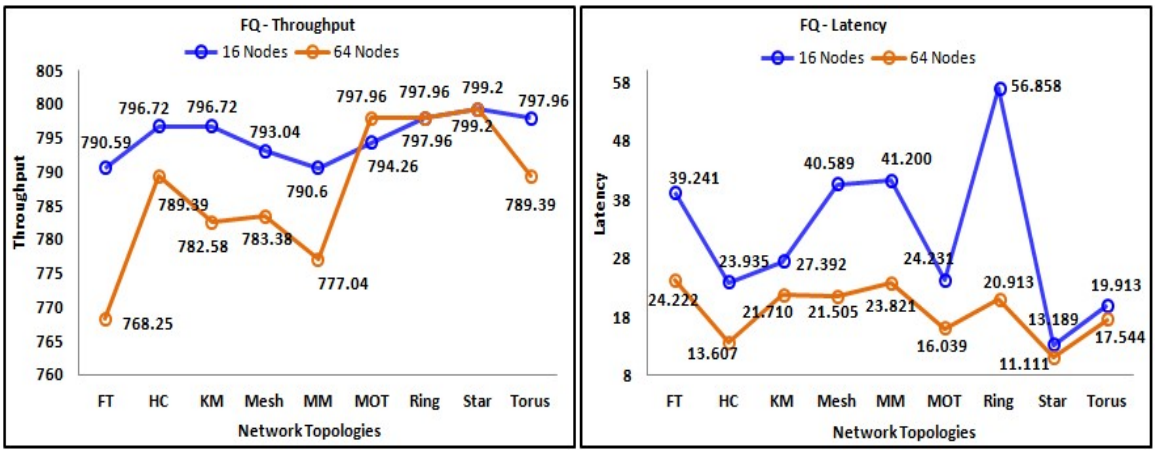


Figure 12: *Achieved throughputs and latencies for various topologies, FT (Fat Tree), MOT (Mesh of Tree), HC (Hypercube), Mesh, KM (K Mesh), MM (M Mesh), Ring, Star, and Torus with QMT as FQ*

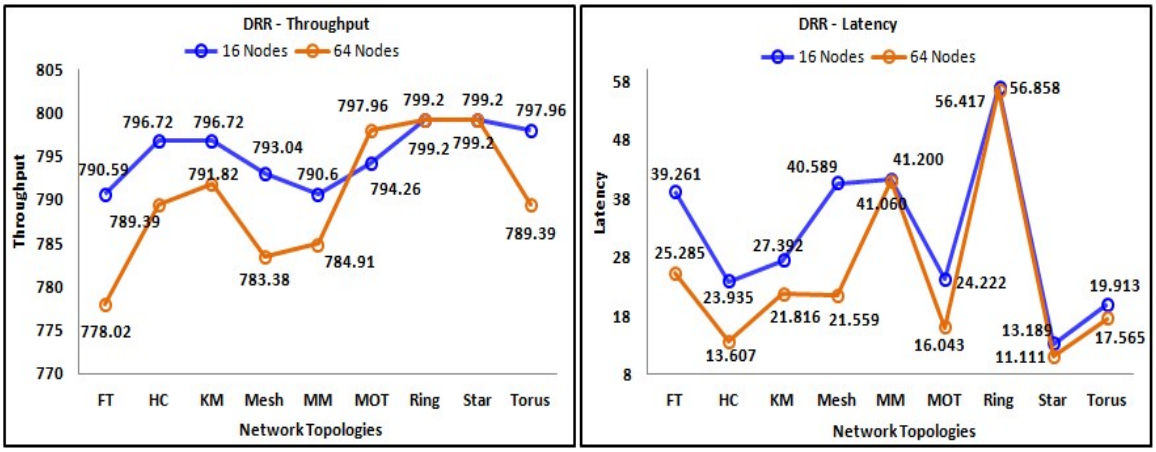


Figure 13: *Achieved throughputs and latencies for various topologies, FT (Fat Tree), MOT (Mesh of Tree), HC (Hypercube), Mesh, KM (K Mesh), MM (M Mesh), Ring, Star, and Torus with QMT as DRR*

Therefore, through the evaluation carried out in the presented work (in terms of throughput and latency) and architectural properties (link complexity, scalability, *etc*.) of these topologies, Torus and Hypercube based topologies are highly recommended topologies for the NoC architecture having 16 cores.

Observations for the topologies simulated with 64 cores or nodes – Like the simulations with 16 cores, all the queue management techniques produced almost same throughput for all the topologies under consideration with 64 cores, however, as seen from Figure 11 and Figure 13, SFQ and DRR based queue management techniques provided highest throughputs for all the considered topologies.

Unlike the simulations with 16 cores, as depicted from Figure 8 to Figure 13, significant variations have been observed in the latencies of the topologies under consideration for different queue management techniques. Among all the queue management techniques, least average latency (around 22) of all the considered topologies is observed for the SFQ queue management technique (Figure 11). As seen from Figure 9 and Figure 13, maximum average latency (around 37.5) of all the considered topologies is observed with RED and DRR based queue management techniques.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Further, Mesh of Tree (MOT) provided a good trade-off between throughput and latency. As seen from Figure 8 to Figure 13, it achieved moderate values for throughput and latency for all the queue management techniques. Further, in absence of articulation points and bridges in MOT based topology, it is highly reliable too. With fixed node degree and lesser link complexity (lesser connectivity of the routers is required for linking purpose) the MOT based topology is scalable too.

Based on the evaluation presented in this work (in terms of throughput and latency) and architectural properties (link complexity, scalability, *etc*.), MOT based topology is highly recommended for the 64 cores NoC architecture.

In some of the contemporary work, performance of various interconnection network topologies had been evaluated over different parameters. In [24], authors evaluated the performance of the MOT based NOC over fat tree based topology. In this paper, authors claimed that MOT based topologies can also be effectively applied in designing NoC based system. In this paper, our experimental study also validates the same, *i.e.* MOT based topology is highly recommended for the 64 cores NoC architecture. Using the topological properties, authors in [18] presented a comparative analysis of suitability of various interconnected topologies for NoC, whereas our analysis is based on the conducted experiments for various configurations of nodes/processing elements.

5. CONCLUSION

In NoC, the resources (*e.g.*, buffers and links) are shared among different flows originated and destined to same or different directions. Meantime, configuring a topology in NoC based architectures remain a significant trade-off between various QoS (quality of service) requirements. This scenario becomes more crucial while managing buffers at the routers to deliver packets under diversified traffic. Hence, selection of the topology to build the NoC system is a challenging task as it affects the performance of the system.

In this paper we dealt with the exploration of few architectural based NoC performance metrics while considering various topologies with two different node configurations, 16 cores and 64 cores. The experimental study suggested that NoC system built over MOT topology is highly suitable for 64 cores NoC system whereas, Hypercube and Torus topologies based NoC system are more suitable for 16 cores NoC system. The results thus obtained elucidate the suitability of these topologies for various performance factors. These results also conclude that the contentions are the major causes of latency variations. Efficiently modelling the effects of resources sharing on a network with more complicated contention scenarios is thus one key research issue.

REFERENCES

[1] Sanchez, D., Michelogiannakis, G., and Kozyrakis, C., "An Analysis of On-Chip Interconnection Network for Large Scale Chip Multiprocessors," ACM Transactions on Architecture and Code Optimization, Vol. 7, No. 1, Article 4, 2010, DOI: 10.1145/1756065.1736069

[2] Bjerregaard, T. and Mahadevan, S. "A Survey of research and practices of network-on-chip," ACM Computing Surveys, Vol. 38, Issue 1, Article No. 1, March 2006, DOI: 10.1145/1132952.1132953

[3] Cota, E., Amory, A. D. M., and Lubaszewski, M. S., Reliabilty, Availability, and Serviceability of Network-On-Chip, Springer, 2012

[4] Grecu, C., Pande, P. P., Ivanov, A., and Saleh, R., "Timing analysis of network on chip architecture for mp-soc platforms", Microelectronics Journal, Vol. 36, Issue 9, September 2005, pp. 833-845, doi.org/10.1016/j.mejo.2005.03.006

[5] Owens, J. D., Dally, W. J., Ho, R., Jayasimha, D. N., Keckler, S. W., and Peh, L-S, "Research challenges for on-chip interconnection networks," IEEE Micro, Vol. 27, Issue 5, Sept-Oct, 2007, pp. 96-108, DOI: 10.1109/MM.2007.4378787

[6] Karim, F., Nguyen, A., and Dey, S., "An interconnected architecture for networking systems on chips," IEEE Micro, Vol. 22, Issue 5, Sept-Oct 2002, pp. 36–45, DOI: 10.1109/MM.2002.1044298

[7] Umamaheswari, S., Perinbam, R. J., "Energy, Throughput and Area Evaluation of Regular and Irregular Network on Chip Architectures," International Journal of Distributed and Parallel Systems, Vol. 2, No. 5, September 2011, pp. 47-56

[8] Achballah, A. B. and Saoud, S. B., "A survey of Network-On-Chip Tools," International Journal of Advanced Computer Sciences and Application, Vol. 4, No. 9, 2013, pp. 61-67

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[9] Benini, L., Micheli, G. D., "Networks on Chips: A New SoC Paradigm," IEEE Computer, Vol. 35, No. 1, Jan 2002, pp. 70-78, DOI: 10.1109/2.976921

[10] Khan, G. N., "NOC: Networks on Chip SoC Interconnection Structures," Electrical and Computer Engineering, Ryerson University, COE838: Systems-on-Chip Design, http://www.ee.ryerson.ca/~courses/COE838/

[11] Tang, M. and Chunhui, W., "A New Method of Designing NoC Routing Algorithm," 2nd International Conference on Consumer Electronics, Communications and Networks, April, 21-23, 2012, pp. 3044-3047, DOI: 10.1109/CECNet.2012.6202077

[12] Chelcea, T. and Nowick, S. M., "Robust interfaces for mixed-timing systems with application to latency-insensitive protocols," in proceedings of the 38th Design Automation Conference, June 22-22, 2001, pp. 21-26, DOI: 10.1145/378239.378256

[13] Dally, W. J. and Towles, B., Principles and Practices of Interconnection Network, Morgan Kaufmann Publishers Inc. San Francisco, CA, USA, 2003

[14] Ortin-Obon, M., Suarez-Gracia, D., Villarroya-Gaudo, M., Izu, C., and Vinals-Yufera, V., "Anlysis of Network-on-Chip topologies for cost-efficient chip multiprocessors," Microprocessors and Microsystems, Vol. 42, May 2016, pp. 24-36, doi.org/10.1016/j.micpro.2016.01.005

[15] Pasricha, S. and Dutt, N., "Networks on Chip," On-Chip Communication Architectures: System on Chip Interconnect, Morgan Kaufmann Publishers Inc. San Francisco, CA, USA, 2008

[16] Duato, J., Yalamanchili, S., and Lionel, N., Interconnection Networks – An Engineering Approach, Morgan Kaufmann Publishers Inc. San Francisco, CA, USA, 2002

[17] Kardam, K. and Singh, A., "A review on comparison on network on Chip (NOC) using simulation tool NS2," International Journal of Computer Science and Information Technologies, Vol. 7 (3), 2016, pp. 1486-1489

[18] Chen, J., Li, C., and Gillard, P., "Network-on-Chip (NoC) Topologies and performance: A Review," In Proceedings of the Newfoundland Electrical and Computer Engineering Conference (NECEC), November 1, 2011

[19] Kochher, S. and Singh, G., "Comparison of various Queuing Techniques to Select Best Solution for Eradicating Congestion in Networks," International Journal of Engineering and Computer Science, Vol. 03, No. 7, July 2014, pp. 7010-7017

[20] Eshete, A. and Jiang, Y., "On the flow fairness of aggregate queues," in proceedings Baltic Congress on Future Internet and Communications, Feb 16-18, 2011, pp. 120-127, **DOI:** 10.1109/BCFIC-RIGA.2011.5733222

[21] Working Mechanism of FQ, RED, SFQ, DRR and Drop-Tail Queues available at https://sites.google.com/a/seecs.edu.pk/network-technologies-tcp-ip-suite/home/performance-analysis-of-impact-of-various-queuing-mechanisms-on-mpeg-traffic/working-mechanism-of-fq-red-sfq-drr-and-drop-tail-queues. Last accessed on July 12, 2018

[22] Stafford, E., Bosque, J. L., Martınez, C., Vallejo, F., Beivide, R., Camarero, C., and Castillo, E., "Assessing the Suitability of King Topologies for Interconnection Networks," IEEE Transactions on Parallel and Distributed Systems, Vol. 27, No. 3, March 2016, pp. 682-694, DOI: 10.1109/TPDS.2015.2409865

[23] Das, D., De, M., and Sinha, B. P., "A new network topology with multiple meshes," IEEE Transactions on Computers, Vol. 48, No. 5, May 1999, pp. 536-551, DOI: 10.1109/12.769436

[24] Kundu, S., Soumya, J., and Chattopadhyay, S, "Design and evaluation of Mesh-of-Tree based Network-on-Chip using virtual channel router," Microprocessors and Microsystems, Vol. 36, Issue 6, August 2012, pp. 471-488, doi.org/10.1016/j.micpro.2012.05.012