

CAFT: Cost-aware and Fault-tolerant Routing Algorithm in 2D Mesh Network-on-Chip

Akram Reza¹, Parisa Jolani², Midia Reshadi³

1- Department of Computer Engineering, Shahr-e-Qods Branch, Islamic Azad University, Tehran, Iran.(a.ak.reza@gmail.com)
2,3- Department of Computer Engineering, Science and Research Branch, Islamic Azad University, Tehran, Iran.

Received (2018-06-17)

Accepted (2019-07-25)

Abstract: By increasing, the complexity of chips and integrating more components into a chip has made network –on- chip known as an important infrastructure for network communications on the system, and is a good alternative to traditional ways and using the bus. By increasing the density of chips, the possibility of fault in the chip network increases and providing correction and fault tolerance methods is one of the principles of today's chip design. Faults may have undesirable effects on the correct system operation and system performance. In this paper the communication infrastructure fault has been considered as same as link and router fault and the fault tolerance low cost routing algorithm has been suggested base on local fault information By using quad neighbor fault information to avoid back tracking in routing in order to select possible minimal path to destination. In this article, we have suggested cost aware fault tolerance (CAFT) routing algorithm. Our contribution in this algorithm is minimum local fault information, minimum routing decision overhead by implementing routing logic base and determining shortest possible path. For deadlock freedom using an additional virtual channel along Y dimension and prohibiting certain routing turns. In order to evaluate the performance of our routing, we compared it with other fault tolerant routing in terms of average packet latency, throughput and power.

Keywords: Network-on-Chip, Fault tolerance, Deadlock-free, routing algorithm, 2D-NOC, Adaptive routing.

How to cite this article:

Akram Reza, Parisa Jolani, Midia Reshadi. CAFT: Cost-aware and Fault-tolerant Routing Algorithm in 2D Mesh Network-on-Chip. J. ADV COMP ENG TECHNOL, 5(4) Autumn 2019 : 205-212

I. INTRODUCTION

MODERN microchips consist of billions transistors and hundreds of processing elements on a single chip [1]. As the number of processing elements is increasing, powerful communication infrastructure is needed. Network on chip has been developed as a strong, scalable, and extensible communications infrastructure for the complex MPSOCs [2], [3]. Fault is considered as a real problem when it causes a malfunction in the global network. Base on lifetime of fault two types of fault can be occur in networks-on-chip, Transient

and permanent faults. Transient faults are temporary and unpredictable but permanent faults are caused by physical damages such as manufacturing defects, in order to enhance reliability of NOC, two strategy suggested overcoming fault, repair fault component and tolerant fault component. One of the fault tolerance methods is fault tolerant routing algorithms that will be converse in this article. Fault tolerance routing algorithm tolerates fault by rerouting packets around a faulty area. [4].

Prior to proposing a routing algorithm, it should be noted that the appropriate topology selection is one of the factors affecting



This work is licensed under the Creative Commons Attribution 4.0 International Licence.

To view a copy of this licence, visit <https://creativecommons.org/licenses/by/4.0/>

network performance. Therefore, in this paper, the 2D topology, which is one of the most popular topologies, has been selected. One of the most significant feature of the proposed topology is the existence of path diversity, which there are different paths between source and destination. The fault tolerant routing algorithm can select alternative paths when the original path is faulty. Therefore, the proposed algorithm should be adaptive. In the adaptive routing, alternative paths between two nodes may be used if the original path or a local link is congested or faulty [5].

One of the main challenges in the adaptive algorithm is the deadlock problem. Several techniques exist to solve this problem in a NoC for 2D topologies. Glass and Ni [6] proposed a technique called the "turn model" to prevent deadlock, adds restrictions to the routing choices for the non-dependence of channels from each other [7], [8], [9]. The second possible solution uses virtual channels and Turn Model, Dally and Seitz [10] introduce virtual channels to make the deadlock region acyclic. [11] The second method has higher adaptation and higher cost [11]. Fault repair scheme is replacing spare component using hardware plugins [12], fault tolerance means that the protocol is able to get through network faults in order to continue its normal service. [13] Fault-tolerant routing algorithm and mapping are two available methods for tolerating fault which in this paper we have exploited the routing algorithms. Fault tolerance routing algorithm requires local or global fault information [5], and as much as the algorithm's adaptiveness is increased, the fault tolerance increases. In this paper for implement low cost fault tolerance routing, the first part we use square local information for minimizing data overhead and in second step to reduce cost in implement adaptive routing. In order to distribute the traffic overhead, we have exploited non-adjacent nodes information by two-hop distance because there is no step back for selecting minimal or non-minimal routes. We use two virtual channels in one dimension and for Speeding up routing decision, routing algorithm has logic base dimension. This paper aims to introduce new deadline-free and fault tolerant routing algorithm in 2D mesh network-on-chip using two-hop-neighbor fault data with minimum number of extra/additional VC.

The remainder of this paper is organized as follows: section 2, some related work is given and in section 3 diagnosis for faulty links and assumptions are described. Then turn model and virtual channel are presented in section 4. section 5 illustrates the proposed fault tolerant routing algorithm. The experimental setup and various results for the performance of the proposed method are presented in section 6. Finally, section 7 concludes this paper.

II. RELATED WORK

The content of related work is divided into two sections. The first section is about fault tolerance. The second section is the introduction of the previous fault tolerance routing algorithm. Fault tolerance is ability to route packets in the presence of faulty components. Although it seems that fault tolerance implies adaptivity, this is not necessarily true. Fault tolerance can be performed without matching the packaging path in two or more phases, and maintains it at some medium nodes. In some cases, fault tolerance also requires additional hardware mechanisms [5], [17], [18]. There are many fault-tolerant routing algorithms to tolerate faulty components in two dimensional (2D) mesh networks, to avoid deadlock conditions some of these algorithms use turn model and the others use virtual channels [11]. The routing schemes introduced in [14], produce an equivalent fully adaptive and minimal routing algorithm called double-y for 2D mesh. It requires two virtual channels in Y dimension. Mad-y allows more number of routing turns by making better utilization of virtual channels to increase adaptivity.

The LEAR turn model has same turn constraints as in mad-y, except 180-degree turns. The deadlock freedom of both mad-y and LEAR routing algorithms is proved by using Dally and Seitz work [10] but they are not fault tolerance algorithms [14]. Unlike traditional methods, by using HiPFaR, packets possibly have alternative choices of minimal paths when facing a faulty network using the shortest paths, as long as a path exists [15]. In this paper, the proposed fault-tolerant routing algorithm is based on a fully adaptive routing algorithm using the minimum number of virtual channels.

III. PROPOSED APPROACH

To define fault tolerance routing algorithm with minimal cost, this subsection has four parts.

The four level of our method can be define as follows:

- 3.1- Fault information distribution.
- 3.2- Cost aware fault tolerance routing algorithm (CAFT)
- 3.3- The turn model and deadlock freedom
- 3.4- Logic base routing circuits to reduce overhead time

3.1- Fault information distribution:

In this work, we do not define fault diagnosis method and only an effective way to determine the fault situation will be presented, The fault information can be propagate in network locally or globally. In this work because of cost constrain we use local information. To avoid extra cost to obtain fault information of all nodes, we use k-neighborhood diagnosis, in the method each node records the statues of all faulty nodes or links within distance k. this is an important property of CAFT to find the best distance where there is no backtrack or deadlock.

Let us consider the example of fig 1 where S represents the source node, D is the destination node located in the northeast of the source node, and we assume that current node knows fault information of direct neighboring nodes west, east, north and south. Due to the faults shown in the figure 1, we have to backtrack. Therefore, this form of storage fault data is not appropriate. By testing different forms and distances, the best method is to store square of neighbors same as fig2. For simplicity, indirect nodes (NE, SE, NW,SW) are called corner square nodes (CS), and direct nodes (N,S,E,W) are called boundary square nodes (BS).

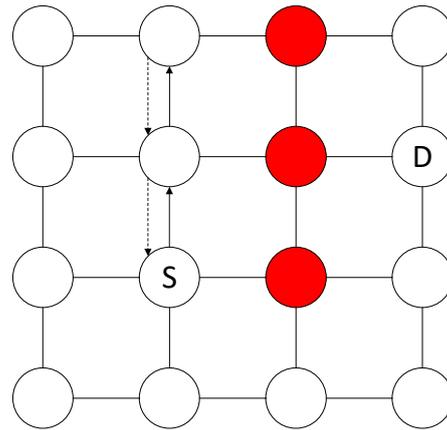


Figure 1. Backtracking in routing because of faulty chain node in rout.

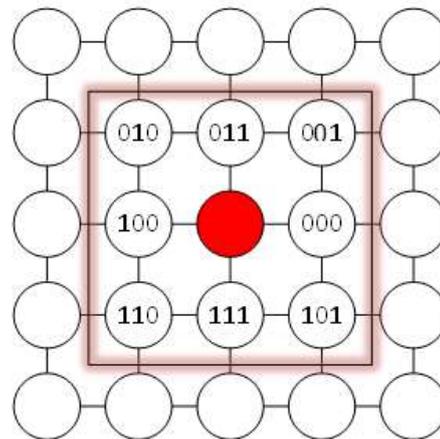


Figure 2. The statuses of links and routes are needed by proposed algorithm

For determining fault location we use 3 bits overhead, based on this three bits we can define location of faulty node. Circuits in figure 3 show the location of faulty nodes in BS and CS and their interconnecting links. For example A (0), A (1), A (2) are output signals and 3-bit fault information. If each router is faulty, this 3-bit specify that router is faulty or not.

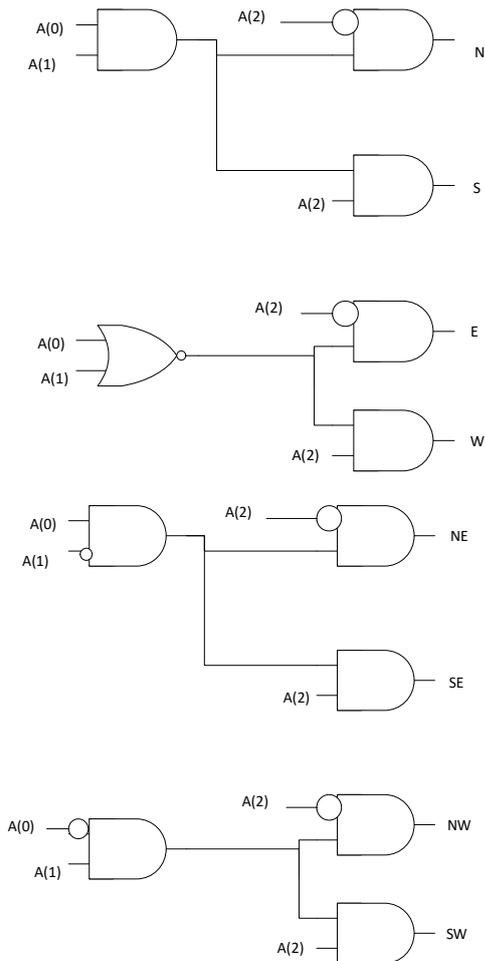


Figure 3. Circuits for determining the position of fault node

3.2- Cost aware fault tolerance routing algorithm (CAFT):

As mentioned before, if value of the signal is 1, that node is defective. Otherwise, the node is healthy and packet can be routed from this node. By this information, node is notified of the fault location and chooses the path based on it. One of the goals of presenting the proposed algorithm is tolerating faults with using minimum path but under the conditions mentioned in the algorithm which non-minimal routing is used. We start by defining nodes and number of their connections, as shown in fig 4. Each node is assigned a label in 2-connection, 3i-connection and 4-connection. For example, node number 2 is 2-connection, CS node, node number 3 is 3-connection, BS node, and node number 4 is 4-connection node.

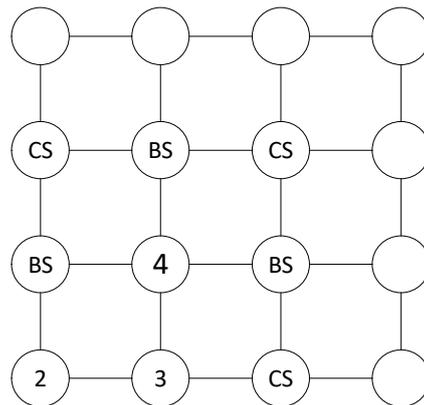


Figure 4. Nodes and number of their connections

In next step, algorithm is shown in fig 5 describe function of the nodes When the fault occurs. Let us explain operation of algorithm, each node depending on its location , number of connections, destination location and fault location, chooses a path that in addition to being short, does not face fault node again and there is no backtrack.

CAFT Routing Algorithm

Input: current and destination node address, fault location

Output: selected out port.

Procedure:

If no fault in min path

XY routing algorithm

If one fault in BS node and in minimal path

Select other minimal path direction

If fault (BS and CS)

If current node = 2-connection

Select other minimal path direction

If current node = 3-connection

Select other available direction in non-minimal path

If current node = 4-connection

Select reverse available direction in minimal path

Figure 5 . CAFT routing algorithm

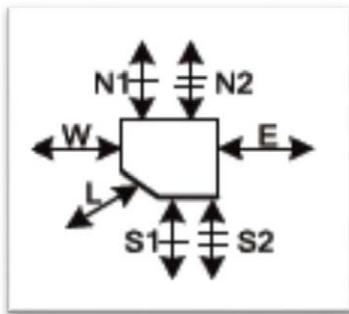
3.3- The turn model and deadlock freedom:

In adaptive routing algorithm the possibility of happening deadlock may increase, two solutions are presented to avoid deadlock. First, virtual channel is added in each dimensions, which will

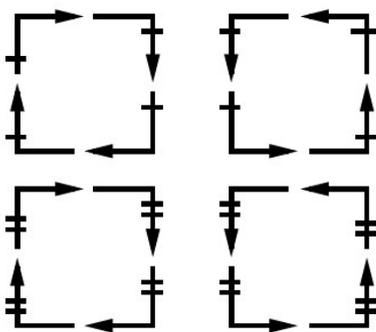
impose extra cost because induce extra buffer area and complex control logics. Second, we can use turn model, this method restricts a number of turn to break all the cycles but reduces additivity. In order to improve additivity and reduction in costs we use both solutions and present a routing algorithm that tolerates faults with minimum number of VCs and unnecessary restrictions on the routing turn concurrent. The idea of this article is developed by 2 VCs in Y dimension which creates seven pairs of channels as shown in fig 6-a and four cycles formed in network (fig 6-b), to avoid deadlock one turn is prohibited from each cycle (fig 6-c).

In summary, to avoid deadlock and backtrack (live lock) the following turns are prohibited:

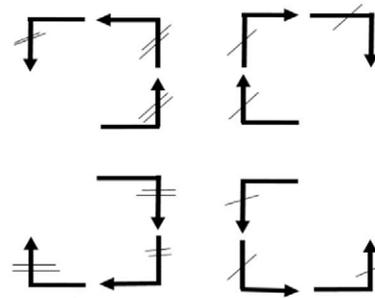
1. It prohibits four 90-degree turns (N1-W, N2-E, S1-W,S2-E)
2. It also prohibits two 0-degree turns (S2-S1, N2-N1)
3. Since it is a minimal routing algorithm, it does not permit any 180 –degree turn.
- 4.



(a)



(b)



(c)

Figure 6. (a) A switch in a double-Y network. (b) four cycles formed by the sixteen turns in a 2D mesh with double Y channels

IV. SIMULATION AND EVALUATION PROPOSED FAULT TOLERANCE ROUTING

In order to evaluate the performance, reliability and overhead of the proposed fault-tolerant routing scheme, we implemented our proposed algorithm in Noxim simulator. The Noxim simulator is developed using SystemC, a system description language based on C++ programming language [16].

We use the common 4*4 2D Mesh NoCs. For all the routers, the data width is set to 32 bits. Each input buffer can accommodate four flits in each channel. Moreover, the minimum packet length is 8 flits. The request rate is defined as the ratio of the successful packet injection into the network over the total number of injection attempts. As a performance metric, we use latency defined as the number of cycles between the initiation of a packet issued by a processing element and the time when the packet is completely delivered to the destination. For performance analysis, the simulator is warmed up for 10,000 cycles .In warm-up option you can set the start time after the simulator starts to collect statistics.

TABLE I
Simulation configuration.

Parameters/System			
buffer size of each channel of the router	4 flits	Packet Injection Rate	0.01 [0.01-0.1]
min_packet_size	8 flits	evaluation	throughput, delay power consumption
warm-up	1000 cycles	topology	mesh
clock cycles that have to be simulated	100000 cycles	Simulation_time	10000 cycles
Switching	Wormhole	reset_time	1000

According to the simulation result in Figure 8, a 4×4 2D- mesh, with faults under random and shuffle traffics have been selected. Based on simulation results, the proposed method shows 5% improvement because of the minimal data overhead and the use of one virtual channel in Y dimension and logical base routing, which selects the shortest path and does not return backwards.

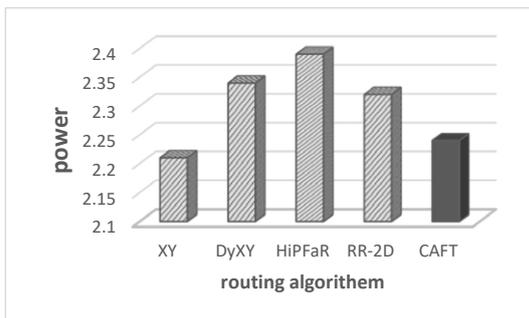
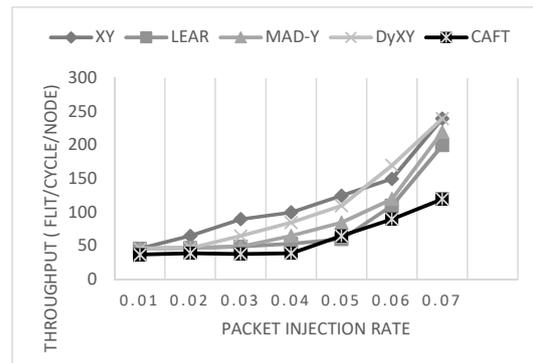
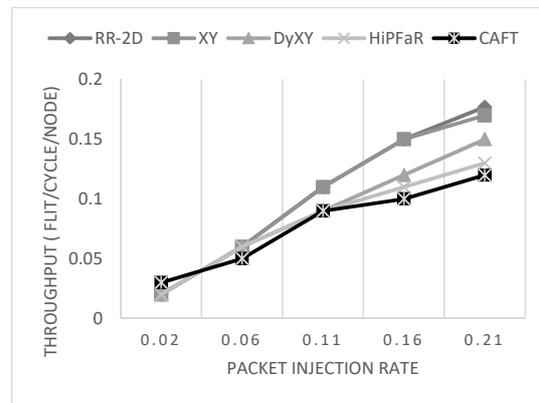


Figure 7. Power under random traffic



(a)

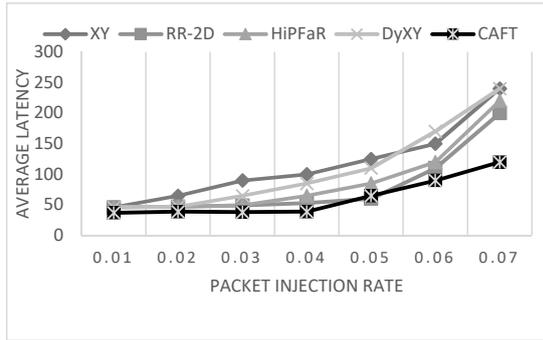


(b)

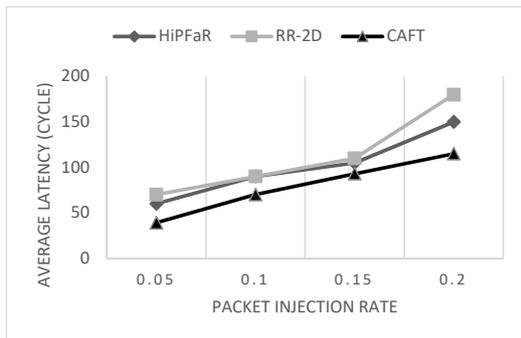
Figure 8. (a) throughput under random traffic (b) throughput under shuffle traffic

Also due to the fact that there is no backtrack in the proposed routing based on low local information, the throughput about 30% improvement Compared with other suggested fault tolerance routing in shuffle traffic as show in figure 8.

Also as show in figure 9, average latency in deferent packet injection rate we have more than 40% improvement Compared with other suggested fault tolerance routing in random traffic and 24% improvement in shuffle traffic. This improvement is because of the suggested fault tolerance routing selected shortest possible route based on local fault information without back tracking.



(a)



(b)

Figure 9. (a) average latency under random traffic (b) average latency under shuffle traffic

V. CONCLUSION

In this paper, a two-dimensional mesh-based routing algorithm is proposed based on the turn-model and the addition of a virtual channel, which is at least a hardware overhead. This algorithm uses minimal routing when there is not faulty node or link in this path while, non-minimized routing is used if there is a minimum path or congestion path. This algorithm has reduced the cost of hardware by adding a minimum of hardware overhead and also eliminates the circulation constraints using the virtual channel.

The use of two-hop-neighbor fault information and minimum number of VC impose minimum overhead on the system

REFERENCES

1. Moadeli, M., Shahrabi, A., Vanderbauwhede, W. and Ould-Khaoua, M., 2007, May. An analytical performance model for the Spidergon NoC. In *Advanced Information Networking and Applications, 2007. AINA'07. 21st International Conference on* (pp. 1014-1021). IEEE.
2. Yaghini, P.M., Eghbal, A. and Bagherzadeh, N., 2015. On the design of hybrid routing mechanism for mesh-based network-on-chip. *INTEGRATION, the VLSI journal*, 50, pp.183-192.
3. Wu, Z., Fu, F., Lu, Y. and Wang, J., 2015. A role-changeable fault-tolerant management strategy towards resilient NoC-based manycore systems. *Microelectronics Journal*, 46(12), pp.1371-1379.
4. Ahmed, A.B. and Abdallah, A.B., 2014. Graceful deadlock-free fault-tolerant routing algorithm for 3D Network-on-Chip architectures. *Journal of Parallel and Distributed Computing*, 74(4), pp.2229-2240.
5. Duato, J., 1995. A necessary and sufficient condition for deadlock-free adaptive routing in wormhole networks. *IEEE Transactions on Parallel and Distributed Systems*, 6(10), pp.1055-1067.
6. Glass, C.J. and Ni, L.M., 1998, August. The turn model for adaptive routing. In *25 years of the international symposia on Computer architecture (selected papers)* (pp. 441-450). ACM.
7. Xie, R., Cai, J. and Xin, X., 2016. Simple fault-tolerant method to balance load in network-on-chip. *Electronics letters*, 52(10), pp.814-816.
8. Xie, R., Cai, J. and Xin, X., 2016. Simple fault-tolerant method to balance load in network-on-chip. *Electronics letters*, 52(10), pp.814-816.
9. Glass, C.J. and Ni, L.M., 1996. Fault-tolerant wormhole routing in meshes without virtual channels. *IEEE transactions on parallel and distributed systems*, 7(6), pp.620-636.
10. Dally, W.J. and Seitz, C.L., 1988. Deadlock-free message routing in multiprocessor interconnection networks.
11. Valinataj, M., Mohammadi, S., Plosila, J., Liljeberg, P. and Tenhunen, H., 2011. A reconfigurable and adaptive routing method for fault-tolerant mesh-based networks-on-chip. *AEU-International Journal of Electronics and Communications*, 65(7), pp.630-640.
12. Zhou, J., Li, H., Wang, T. and Li, X., 2016. LOFT: A low-overhead fault-tolerant routing scheme for 3D NoCs. *Integration, the VLSI Journal*, 52, pp.41-50.
13. Huang, L., Zhang, X., Ebrahimi, M. and Li, G., 2016. Tolerating transient illegal turn faults in NoCs. *Microprocessors and Microsystems*, 43, pp.104-115.
14. Kumar, M., Laxmi, V., Gaur, M.S., Daneshtalab, M., Ebrahimi, M. and Zwolinski, M., 2014, October. Fault tolerant and highly adaptive routing for 2D NoCs. In *Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT)*, 2014 IEEE International Symposium on (pp. 104-109). IEEE.
15. Ebrahimi, M., Daneshtalab, M. and Plosila, J., 2013, February. High performance fault-tolerant routing algorithm for NoC-based many-core systems. In *Parallel, Distributed and Network-Based Processing (PDP), 2013 21st Euromicro International Conference on* (pp. 462-469). IEEE.
16. Valinataj, M., Mohammadi, S., Plosila, J., Liljeberg, P. and Tenhunen, H., 2011. A reconfigurable and adaptive routing method for fault-tolerant mesh-based networks-on-chip. *AEU-International Journal of Electronics and Communications*, 65(7), pp.630-640.
17. Sinha, D., Roy, A., Kumar, K.V., Kulkarni, P. and Soumya, J., 2018, March. D n-FTR: Fault-tolerant routing algorithm for Mesh based network-on-chip. In *2018 4th International Conference on Recent Advances in Information Technology (RAIT)* (pp. 1-5). IEEE.
18. Xie, R., Cai, J., Xin, X. and Yang, B., 2017. Low-cost adaptive and fault-tolerant routing method for 2D Network-on-Chip. *IEICE TRANSACTIONS on Information and Systems*, 100(4), pp.910-913.
19. M. Palesi, D. Patti and F. Fazzino, "http://www.noxim.org/," University of Catania, 2005-2010. [Online].