
Bio-Inspired Solutions and Its Impact on Real-World Problems: A Network on Chip (NoC) Perspective

Muhammad Athar Javed Sethi, Momil Ijaz,
Huma Urooj and Fawnizu Azmadi Hussin

Additional information is available at the end of the chapter

<http://dx.doi.org/10.5772/intechopen.83229>

Abstract

Bio-inspired solutions are used to solve the real-world problems as they are able to resolve the complex issues. Already existing bio-inspired solutions are reviewed in this chapter which solved the complex engineering problems. Moreover, this chapter also discusses the implementation of biological brain mechanism in Network on Chip to address the fault-tolerant issues. Network on Chip (NoC) is a communication framework for System on Chip (SoC). Due to routers and interconnect failure, NoC suffers from faults. Therefore, bio-inspired solutions help us to recover from these faults. The techniques from the biological brain were implemented in NoC as the brain is fault tolerant and highly adaptive. Results showed that bio-inspired techniques are performing well compared to the traditional fault-tolerant algorithms.

Keywords: bio-inspired solutions, Systems on Chip, Network on Chip, fault-tolerant algorithms, synapse, neuron

1. Introduction

It is more than 20 years now that scientists and researchers are taking inspiration from biological solutions to build dynamic, real-time and robust applications and systems. The characteristics of adaptability, robustness and resilience to handle failure make biological solutions ideal for solving the real-world complex engineering problems [1]. Based on the literature, the bio-inspired research is basically divided into three main components, that is, bio-inspired computation, bio-inspired systems and bio-inspired networking. In bio-inspired computation, those algorithms are considered which focused on optimization and efficient computing;

however, distributive systems are studied under bio-inspired systems. Bio-inspired networking deals with scalable networking techniques for autonomic organizations and systems. A biological concept should be similar to the real-world problem in order to implement it. It should be realistic to implement it, and the methodology should be clearly defined how to convert it to the real-world system [2].

Biological brain is highly robust and fault tolerant. Synaptogenesis, sprouting, dynamic brain, synapse rearrangement, the long-term potential (LTP) and long-term depression (LTD), nerve growth factor (NGF) and redundant neuron are few fault-tolerant techniques of the biological brain. Synaptogenesis and sprouting algorithms are implemented in Network on Chip (NoC) to make it fault tolerant. **Figure 1** shows the basic structure of 4×4 NoC having mesh topology. NoC is scalable and can be connected in various topologies, which include torus, star, ring, honeycomb and tree. The figure shows NoC having 16 routers and PEs; moreover, PEs can be homogeneous or heterogeneous resources. PEs can be a processor (P), memory (M), cache (C), reconfigurable block (re), digital signal processing (DSP) core or any other intellectual property (IP) cores as shown in **Figure 2**.

There are four broad categories of NoC fault-tolerant techniques, which are deterministic, stochastic and fully adaptive and partial adaptive routing algorithms [3, 4]. Deterministic routing algorithm routes the flit (flow digits; NoC messages are divided into packets, while packets contain flits) in a particular fix direction. XY and YX are types of deterministic routing algorithms. In XY and YX, flits are transferred in the horizontal direction (x-axis) towards destination, PE; later the flits traverse the vertical direction (y-axis), so that flits would reach

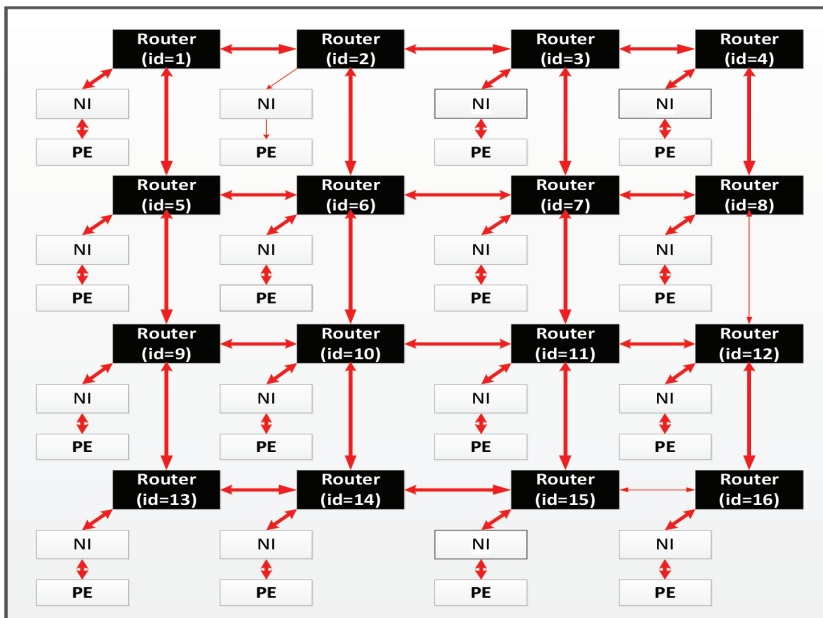


Figure 1. Network on Chip (NoC).

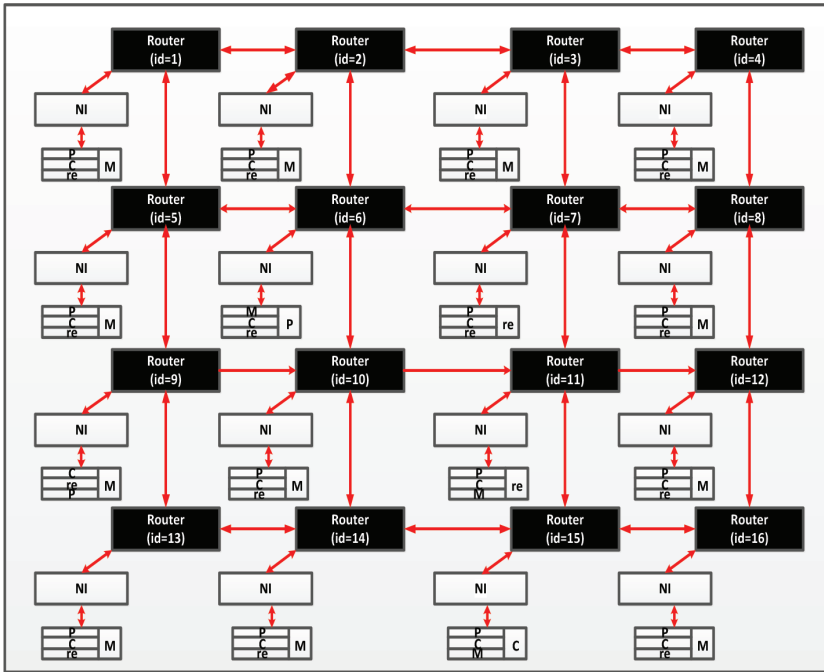


Figure 2. NoC with heterogeneous resources.

the destination PE. The same is the case for YX routing algorithm, in which, flits first cover the vertical direction towards destination PE and later flit traverses the horizontal direction to reach destination PE. Deterministic routing algorithms are not fault tolerant and do not contain adaptivity.

Similarly, stochastic routing algorithm achieves fault tolerance by broadcasting the flits in all directions, until flits reach the destination PE. Stochastic routing algorithms utilize more bandwidth and power. Fully adaptive routing algorithms are adaptive and fault tolerant. However, updating the routing table consumes high power, bandwidth and time. Partial adaptive routing algorithms have restrictions on certain turns in NoC although they are fault tolerant and consume less power. Odd-even is a famous example of the partial adaptive routing algorithm. Odd-even puts a restriction that there will be no flit turn from east to north and from east to south at even NoC columns; moreover, at odd NoC column, no flit turn is allowed from south to west and north to west direction.

Majority of the traditional fault-tolerant algorithms do not completely address the faults in the NoC and have the drawbacks of high packet network latency (time taken from source PE to destination PE) and inter-flit arrival time (time taken between two flits at destination PE), low bandwidth utilization and less throughput. This makes the NoC communication unreliable, and the NoC architecture is not fault tolerant. To overcome the drawbacks of deterministic, stochastic and fully adaptive and partially adaptive routing algorithms, novel biological-inspired

fault-tolerant algorithms were proposed. Bio-inspired algorithms are able to address the limitations of fault-tolerant algorithms by mimicking the fault-tolerant and robust mechanism of the brain based on the algorithms called 'synaptogenesis' and 'sprouting'. Novel bio-inspired NoC fault-tolerant algorithms have been inspired by biological brain algorithms. The bio-inspired NoC fault-tolerant algorithms have the following characteristics:

- Bio-inspired algorithms do not follow any fixed path towards a destination compared to a deterministic routing algorithm.
- The proposed algorithms do not broadcast the packet in any directions, which efficiently utilized the bandwidth compared to stochastic routing algorithms.
- There is no routing table in these algorithms, yet they are highly adaptive and robust compared to fully adaptive algorithms.
- There is no restriction on turns in the NoC; rather it takes two hops (nodes) neighbor information before making decisions on the turn it takes compared to partial adaptive algorithms.

2. Bio-inspired solutions

In literature, there are a number of examples that have solved the real-world problems using biological inspiration.

2.1. Ant colony optimization

Ant colony optimization (ACO) algorithm is basically based on the foraging behaviour of the ants. ACO is adopted in the organization of parallel distributed systems. In the biological world, ants collaborate with each other with the releasing special liquid (pheromone) when they find the food. Later, ants group together and follow this shortest path constructed because of the release of this liquid [5].

2.2. Artificial immune system

Artificial immune system (AIS) is inspired by the biological immune system. AIS is used to detect the environmental changes from the normal system and memorize these characteristics automatically [6].

2.3. Autonomic Network on Chip (NoC) using the biological immune system

The robustness and dynamic nature of biological solutions is being used in the NoC. The self-configuration, self-healing and self-optimization are the few dynamic natural characteristics of the immune system used in the NoC to make it autonomic. The bio-inspired NoC (BNoC) behaves and reacts like an immune system as it detects the pathogens (application behaviour and system state change), which enters in the body (system) and delivers a response to heal (adapts to change) it. In BNoC, self-configuration, self-healing and self-optimization are performed in the application, communication and architecture layers. BNoC could react like an immune system

against pathogens that have entered the body. It detects the infection (application behaviour or system state changes) and delivers a response to eliminate it (i.e. it adapts to changes) [7].

2.4. Fault-tolerant NoC using biological brain techniques

A neuron is the basic building unit of the biological brain. Neuron rearranges itself in case of damage or due to the broken synapse. Synaptogenesis and sprouting are biological brain techniques being adopted in Network on Chip to make it fault tolerant in case of a router or interconnect failure [8–10].

2.5. Epidemic spreading

The dissemination of information in wireless ad hoc networks is inspired by the epidemic spreading technique. The distribution of information particles or the spread of viruses on the web are linked with the dissemination of information [11].

3. Bio-inspired network on Chip

Bio-inspired NoC fault-tolerant algorithms gain inspiration from biological brain fault-tolerant algorithms to make NoC communication reliable. The biological brain is a complex organ with a network of neurons interconnected by synapses. It works faster than any supercomputer. The neuron is the basic building unit of the brain. There are around 80–120 billion neurons in the human brain. **Figure 3** shows the biological neuron [12]. Two biological brain algorithms, namely, synaptogenesis and sprouting, are adopted and implemented to make NoC fault tolerant. These are self-adapting and self-healing mechanisms. These algorithms help the brain to recover when neurons and synapses get damaged. These bio-inspired NoC fault-tolerant algorithms are implemented using two connection setups or communications services. The two communication services are guaranteed-throughput (GT) and best-effort (BE) services.

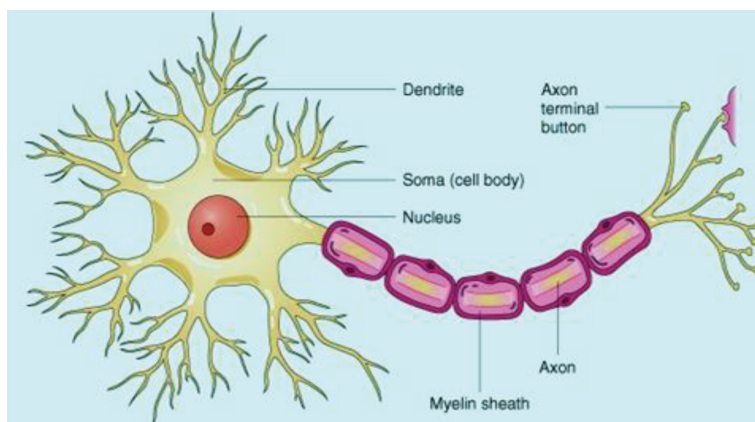


Figure 3. Biological neuron.

3.1. Synaptogenesis

Synaptogenesis is a self-adapting mechanism in the human brain where two neurons attempt to connect and communicate with each other. In this phenomenon, the growth cone (having lamellipodium and filopodia) at the top of the axon and dendrite terminals finds a path to a target neuron. The filopodia actually find the path for a connection with the target neuron. The chemical attractant is released by the target neuron to attract the growth cone. A synapse is formed between the source and target neuron using this method [13]. The synaptogenesis process is shown in **Figure 4**.

3.2. Sprouting

Sprouting is the self-healing mechanism of the brain. Neurons get damage because of hypertension and other different reasons. A new synapse or sprout is created from the already connected source neuron (using chemical adhesion material (CAM)) by establishing a new connection with the target neuron. With the help of this sprout is connected with the target neuron as shown in **Figure 5**. After various experiments on fishes and frogs (amphibians), it is concluded that this feature is not present in humans [13].

3.3. Synaptogenesis-based NoC

The best path is established between the source and destination PE with the help of Synaptogenesis algorithm. Using the algorithm faulty interconnects are detected, and new synapse originates from the neighbor router, which makes the communication reliable and fault tolerant. A credit method is adopted in NoC to detect the faults. Three ns are required to detect the fault as the router does not receive the credit packet from the neighbor router. The concept of credit is shown in **Figure 6**. The flow of flits between routers and PEs is managed by incrementing and decrementing a credit counter. When a port sends a flit, it decrements the credit counter. This specifies

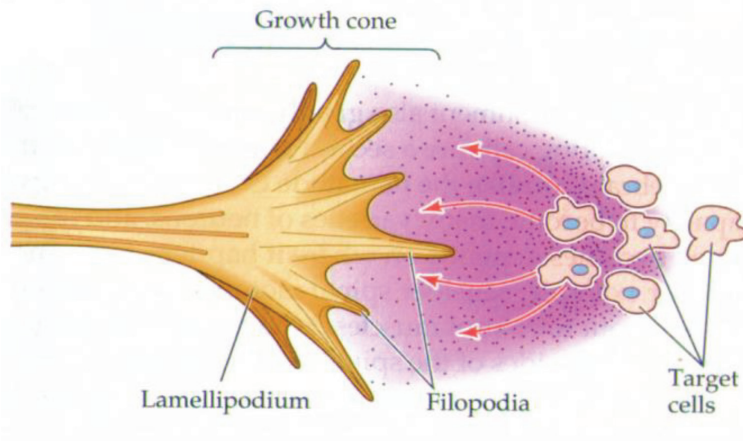


Figure 4. Synaptogenesis.

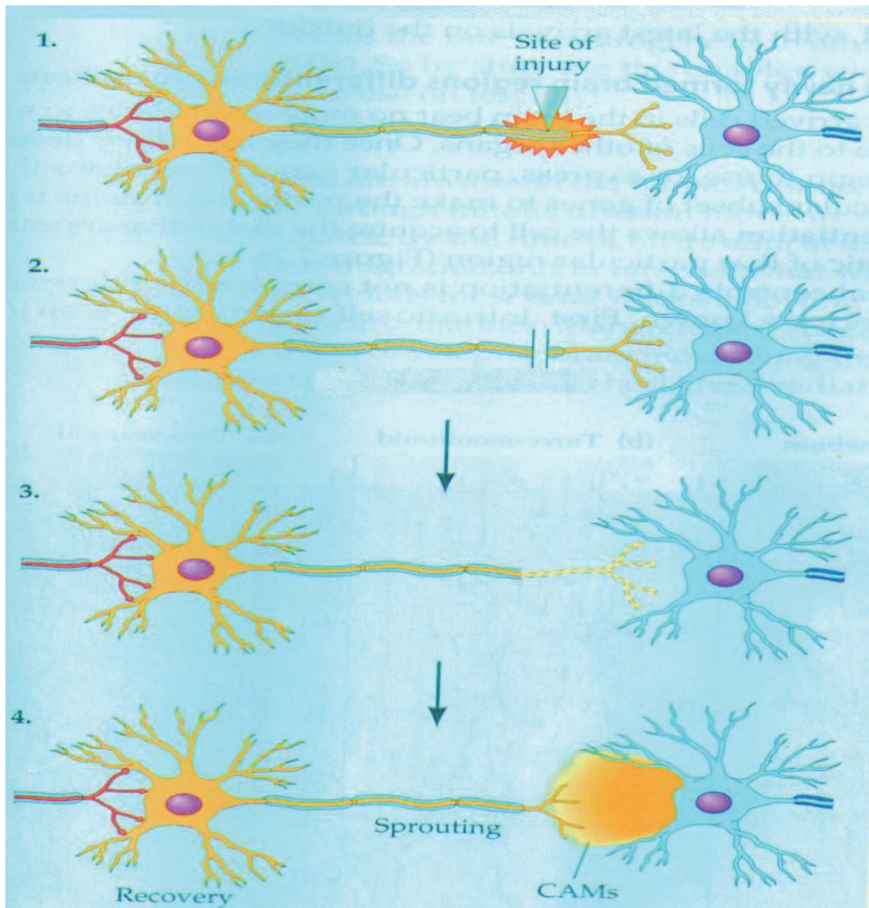


Figure 5. Sprout.

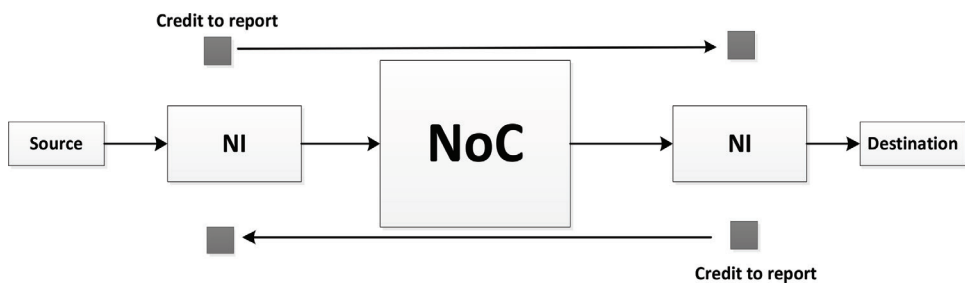


Figure 6. Credit mechanism.

that the output port is busy as it just sends the flit. When the adjacent router receives the flit, it sends back the credit packet to the router so that it increases the credit counter. When the credit counter of the port reaches zero, it cannot send more flits at the particular output port [14, 15].

To improve the fault detection and performance of the NoC, a sprouting algorithm was proposed.

3.4. Sprouting-based NoC

A sprout (or synapse) is originated from the synapse already connected with source PE during the recovery from faults (bypass the fault). During the NoC communication, the faults are detected using sprouting algorithm. As compared to synaptogenesis algorithm, sprouting algorithm does not use the credit mechanism. This helps the algorithm to quickly create a work-around synapse compared to synaptogenesis. The sprout emerges as the destination address is saved in every flit. Therefore, the overall performance of the sprouting algorithm is better than the synaptogenesis algorithm.

The algorithms recover from the static and runtime faulty interconnection by creating a work-around synapse. Multiple synapses are formed between source PE and destination PE which efficiently utilizes the bandwidth, maximizes the throughput and recovers from the faults quickly.

3.5. Best effort (BE) services in NoC

In BE services, no resources (routers, interconnects) are reserved. BE services does not provide any guarantee of the bandwidth. Any source (PE) can send the packet based on the availability of an output port. The decision to send the packet in a particular direction is made on a router-to-router basis based on the destination address. The scheduler of the port detects the fault. In BE services, credit-based flow control is used to control the flow of flits between routers and PEs. Examples of BE connections are cache updates. This shows that BE connections are preferred for noncritical traffic. One of the major drawbacks of the BE services is its unpredictability [16].

3.6. Guaranteed throughput (GT) services in NoC

In GT services, resources are reserved for a particular time period between specific source and destination pair. In GT services, the bandwidth is reserved for particular connections between source and destination pairs. GT connections may underutilize network resources at certain times. This makes the GT connections an expensive option. At certain times, routers and PEs send a burst of data on these GT connections, and then it remains silent for a certain period of time. This leads to the underutilization of network resources. That is why BE services complement the GT services by utilizing the unused bandwidth. Video processing is an example of a GT connection. This implies that GT connections are usually preferred for real-time critical traffic applications [17].

3.7. BE + GT services in NoC

To efficiently utilize the bandwidth of NoC, BE services are provided along with GT services [18]. The flits are saved in best effort queue (BQ) or guaranteed throughput queue (GQ) depending on the source of the flit. Round-robin slot mechanism is used in BE + GT services, whereas GT flits have the highest priority compared to BE flits [19, 20]. The port architecture for BE + GT is shown in **Figure 7**.

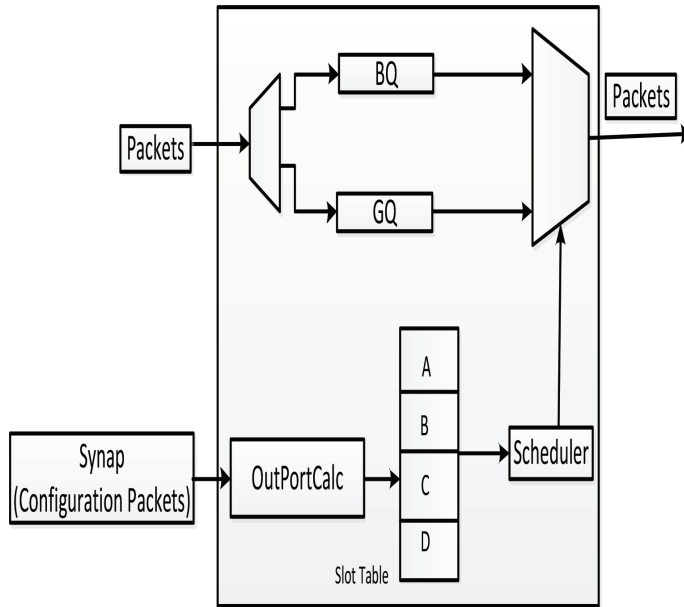


Figure 7. Port architecture.

3.8. Bio-inspired algorithm implementation

The synapse packet is initiated from the source PE and traverses towards the destination using synaptogenesis algorithm. The path is saved in every flit after the creation of synapse to make NoC fault tolerant. During the establishment of synapse, if there is no fault, then flits will traverse on the path. However, if during the communication fault occurs, then neighbor router scheduler detects the fault. The port architecture is shown in **Figure 8**. A new sprout (synapse) emerges from the neighbor router 'inPort' to bypass the faulty interconnect. The port of the router connected with that particular faulty interconnect is blocked through 'Scheduler'. The flits are received at the 'inPort', while the Scheduler is used to send the flit to the neighbor router. The 'OutPortCalc' is used to find the best path for synapse connection.

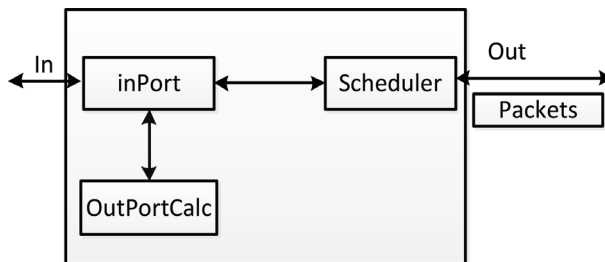


Figure 8. NoC port architecture.

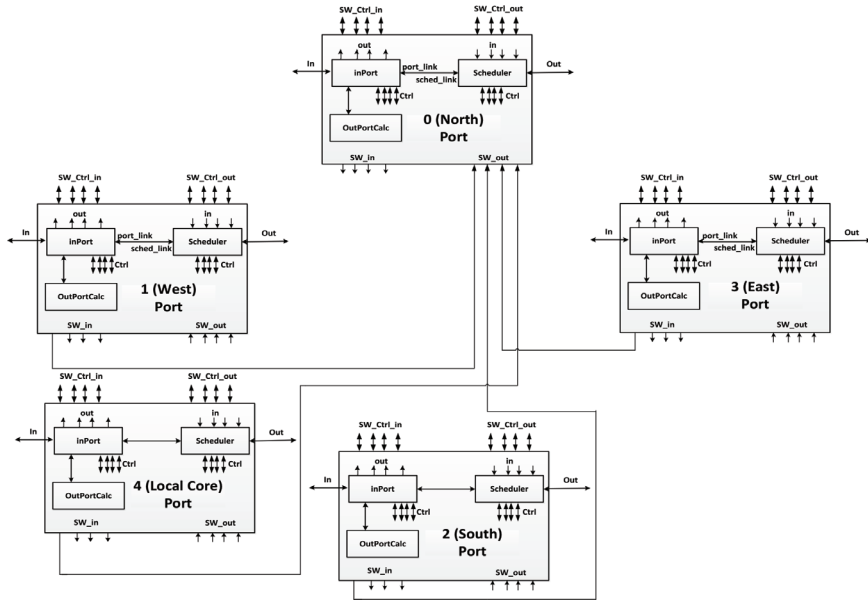


Figure 9. Port path for data flits.

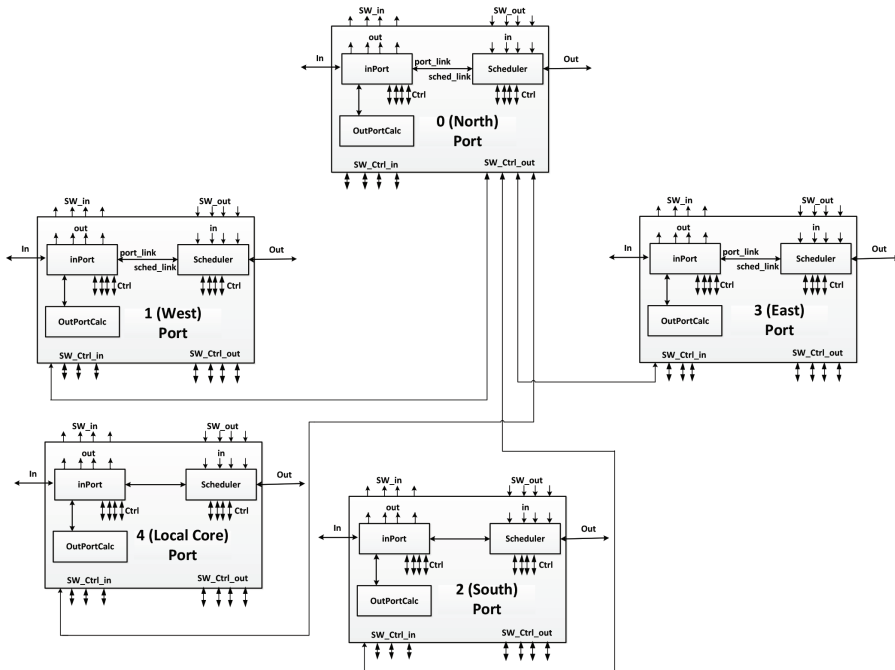


Figure 10. Port path for control flits.

Inside the router, the five ports are connected together using crossbar interconnection. There are separate paths for data and control flits between the ports of the router as shown in **Figures 9** and **10**, respectively. The 'SW_in', 'SW_out', 'SW_Ctrl_in' and 'SW_Ctrl_out' are connected together between different ports of the router. In **Figure 9**, four pins of 'SW_out' of port 0 (north) are connected with the one 'SW_in' pin of port 1 (west), port 2 (south), port 3 (East) and port 4 (local core), respectively. Similarly, four pins of 'SW_Ctrl_out' of port 0 are connected with each 'SW_Ctrl_in' of ports 1, 2, 3, and 4, respectively as shown in **Figure 10**. The same mechanism is used for other ports of a router. The data flits traverse from 'SW_in' of one router port to 'SW_out' of the other port of the same router, whereas control packets traverse on the 'SW_Ctrl_in' and 'SW_Ctrl_out' [21].

With the help of these two bio-inspired algorithms, the NoC becomes robust and fault tolerant and is able to efficiently utilize the bandwidth and maximize the throughput and has less end-to-end latency and inter-flit arrival time than the literature techniques.

4. Results and discussions

In **Figures 11–14**, the bio-inspired algorithm having BE + GT services is compared with literature techniques of XY and odd-even algorithms. As shown in **Figure 11**, the runtime throughput (bits per second) of the bio-inspired algorithm is better than the XY and odd-even by 66 and 58%, respectively.

Moreover, the bandwidth utilization (megabyte per second) by the bio-inspired algorithm is maximum compared to XY and odd-even by 45 and 39%, respectively, as shown in **Figure 12**. Similarly, packet (having flits) network latency (in nanoseconds) of bio-inspired algorithm is less than XY and odd-even as shown in **Figure 13**. **Figure 14** shows that the interflit arrival time of XY and odd-even is more than BE + GT-based bio-inspired algorithm by 49 and 62%, respectively.

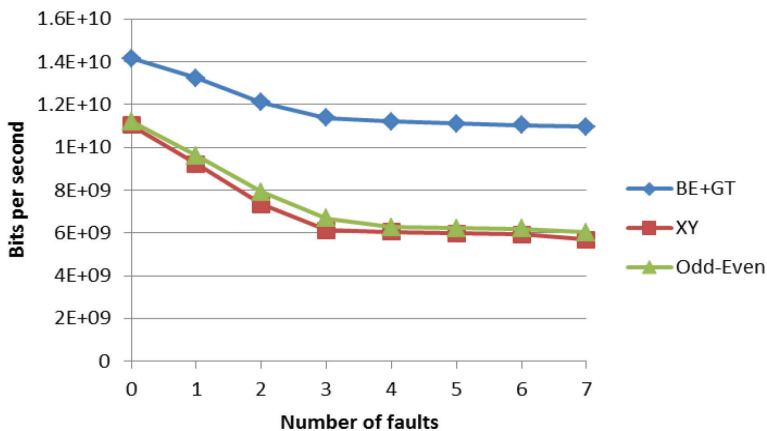


Figure 11. Runtime throughput vs. number of faults.

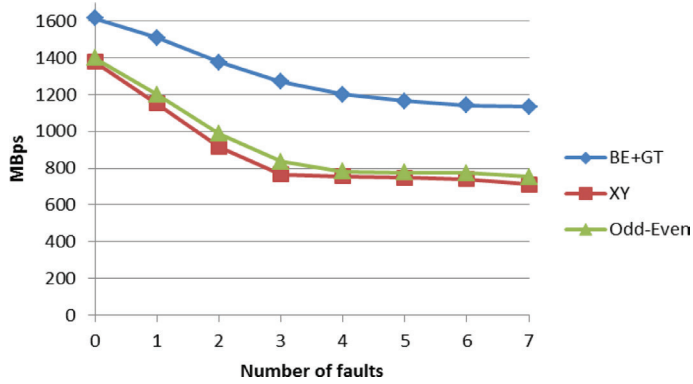


Figure 12. NoC bandwidth vs. number of faults.

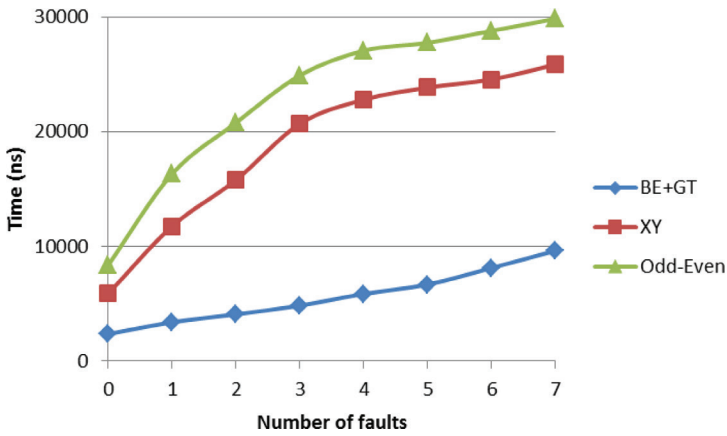


Figure 13. Packet network latency vs. number of faults.

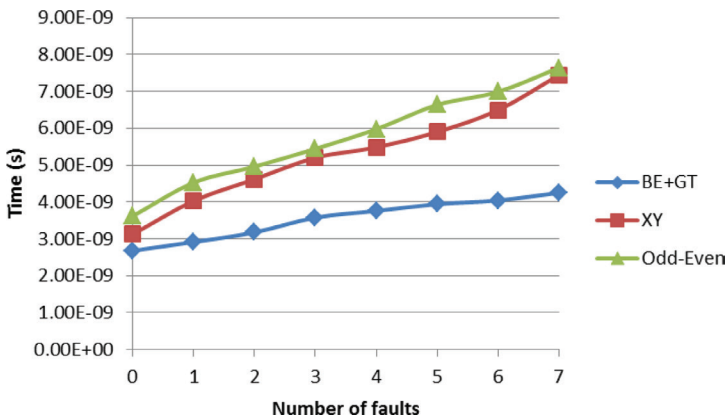


Figure 14. Interflit arrival time vs. number of faults.

5. Conclusion

In this chapter, various biological-inspired solutions are discussed. It is quite wonderful to see how natural phenomenon is able to solve the complex real-world problems. Different biological brain fault-tolerant techniques are presented, which are implemented in NoC to make it fault tolerant. The bio-inspired techniques having BE + GT services are performing better than the literature techniques.

Author details

Muhammad Athar Javed Sethi^{1*}, Momil Ijaz¹, Huma Urooj¹ and Fawnizu Azmadi Hussin²

*Address all correspondence to: atharsethi@uetpeshawar.edu.pk

¹ University of Engineering and Technology, Peshawar, Pakistan

² Universiti Teknologi PETRONAS, Malaysia

References

- [1] Dressler F, Akan OB. A survey on bio-inspired networking. *Computer Networks*. 2010; **54**:881-900
- [2] Dressler F, Akan OB. Bio-inspired networking: From theory to practice. *IEEE Communications Magazine*. 2010; **48**:176-183
- [3] Sethi MAJ, Hussin FA, Hamid NH. Survey of network on chip architectures. *Science International (Lahore)*. 2015; **5**:4133-4144
- [4] Athar Javed Sethi M, Azmadi Hussin F, Hisham Hamid N. Review of network on chip architectures. *Recent Advances in Electrical & Electronic Engineering (Formerly Recent Patents on Electrical & Electronic Engineering)*. 2017; **10**:4-29
- [5] Drigo M. The ant system: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics-Part B*. 1996; **26**:1-13
- [6] Castro LN, De Castro LN, Timmis J. *Artificial Immune Systems: A New Computational Intelligence Approach*. London: Springer Science & Business Media; 2002
- [7] Bakhouya M. Towards a bio-inspired architecture for autonomic network-on-chip. In: 2010 International Conference on High Performance Computing and Simulation (HPCS). USA: IEEE; 2010. pp. 491-497
- [8] Sethi MAJ, Hussin FA, Hamid NH. Synaptogenesis based bio-inspired NoC fault tolerant interconnect. In: 2013 IEEE International Conference on Control System, Computing and Engineering (ICCSCE). USA: IEEE; 2013. pp. 46-51

- [9] Sethi MAJ, Hussin FA, Hamid NH. Implementation of biological sprouting algorithm for NoC fault tolerance. In: 2013 IEEE International Conference on Circuits and Systems (ICCAS). USA: IEEE; 2013. pp. 39-44
- [10] Sethi MAJ, Hussin FA, Hamid NH. Bio-inspired NoC fault tolerant techniques. In: 2014 5th International Conference on Intelligent and Advanced Systems (ICIAS). USA: IEEE; 2014. pp. 1-6
- [11] Vogels W, Van Renesse R, Birman K. The power of epidemics: Robust communication for large-scale distributed systems. *ACM SIGCOMM Computer Communication Review*. 2003;**33**:131-135
- [12] Hashmi A, Berry H, Temam O, Lipasti M. Automatic abstraction and fault tolerance in cortical microarchitectures. *ACM SIGARCH Computer Architecture News*. 2011:1-10
- [13] Rosenzweig MR, Breedlove SM, Leiman AL. *Biological Psychology: An Introduction to Behavioral, Cognitive, and Clinical Neuroscience*. USA: Sinauer Associates; 2002
- [14] Hansson A, Goossens K, Rădulescu A. Avoiding message-dependent deadlock in network-based systems on chip. *VLSI Design*. 2007;**2007**
- [15] Sethi MAJ, Hussin FA, Hamid NH. Implementation and analysis of biological synaptogenesis technique on nodes and interconnects for NoC fault tolerance. Maxwell Scientific Publication Corp. Submitted: September. 2016;**2015**:483-489
- [16] Sethi MAJ, Hussin FA, Hamid NH. Bio-inspired fault tolerant network on chip. *Integration, the VLSI Journal*. 2017;**58**:155-166
- [17] Sethi MAJ, Hussin FA, Hamid NH. Biologically inspired network on chip fault tolerant algorithm using time division multiplexing. In: 2016 6th International Conference on Intelligent and Advanced Systems (ICIAS). USA: IEEE; 2016. pp. 1-6
- [18] Rijpkema E, Goossens K, Radulescu A, Dielissen J, van Meerbergen J, Wielage P, et al. Trade-offs in the design of a router with both guaranteed and best-effort services for networks on chip. *IEEE Proceedings-Computers and Digital Techniques*. 2003;**150**:294
- [19] Sethi MAJ, Hussin FA, Hamid NH. Bio-inspired NoC fault tolerant techniques using guaranteed throughput and best effort services. *Integration, the VLSI Journal*. 2016;**54**:65-96
- [20] Sethi MAJ, Hussin FA, Hamid NH. Bio-inspired network on chip having both guaranteed throughput and best effort services using fault-tolerant algorithm. *IEEJ Transactions on Electrical and Electronic Engineering*. 2018;**13**(8):1153-1162
- [21] Ben-Itzhak Y, Zahavi E, Cidon I, Kolodny A. HNOCS: Modular open-source simulator for heterogeneous NoCs. In: 2012 International Conference on Embedded Computer Systems (SAMOS). USA: IEEE; 2012. pp. 51-57