



## Research paper

# Int-TAR: An Intelligent Thermal-Aware Packet Routing Algorithm for 3D NoCs

Z. Shirmohammadi<sup>1,\*</sup>, M.J. Mahmoudi<sup>2</sup>, M. Rostamnejad<sup>3</sup>

<sup>1</sup>Department of Computer Engineering, Shahid Rajaei Teacher Training University, Tehran, Iran.

<sup>2</sup>Department of Computer Engineering, Khajeh Nasir Toosi University of Technology, Tehran, Iran.

<sup>3</sup>Department of Computer Engineering, Sharif University of Technology, Tehran, Iran.

## Article Info

### Article History:

Received 30 January 2021  
Reviewed 28 February 2021  
Revised 28 March 2021  
Accepted 10 April 2021

### Keywords:

Reliability  
Thermal  
3D NoC  
Routing algorithm

\*Corresponding Author's Email Address:  
[shirmohammadi@sru.ac.ir](mailto:shirmohammadi@sru.ac.ir)

## Abstract

**Background and Objectives:** Thermal problem is one of the main challenges in 3D on-chip networks. Inappropriate traffic distribution, poor heat dissipation, cooling restriction for layers away from the chip heatsink are the main reasons for this problem.

**Methods:** This paper proposes a new intelligent routing algorithm called (Int-TAR) to solve these problems. Int-TAR applies a routing for managing the heat in 3D on-chip networks dynamically. The main idea behind Int-TAR is to save the past states of the system and, according to these states, predict the future behavior of the network and perform routing dynamically. It is done by the threshold of routers dynamically based on the current status of the routers.

**Results:** The simulation results show that Int-TAR decreases the temperature of the network by 13% and improves performance efficiently.

**Conclusion:** The proposed idea shows the better benchmark for the thermal problem in 3D on-chip network. Also, the higher memory for storing the past state of the network can make accurate and the further performance for the network.

©2022 JECEI. All rights reserved.

## Introduction

3D Network On Chips (NoCs) are proposed as an efficient solution to solve the complexity of an on-chip network design and communication problems. These networks combine on-chip networks and 3D chips to eliminate the limitation of 2D chips [1], [2]. In the architecture of 3D on-chip networks, layers are vertically stacked using through-silicon vias (TSVs). Using this architecture leads to a significant reduction in the length of connections [3], [4], [2]. Also, due to the reducing the size of wires, these networks have short latency and lower power consumption in data transmission. As a result, 3D networks have higher performance and lower power consumption for 2D ones, accordingly [5].

In 3D on-chip networks, the power consumption of data transfer is the central portion of power consumption in the chip and causes intense heat production [6]. On-chip networks are highly active for delivering packets through the small area of the chip. Hence, there is more power density and heat-generating compared with processing elements (PE) [7], [8].

The design of cooling systems is based on the analysis of the worst temperature state. With the increase in thermal density and complexity of the systems, these analyses become more complex and even impossible. It is complicated and infeasible to design a cooling system [6]. Furthermore, the power consumed by the cooling system is half to one per watt in computation.

Many recent studies attempted to reduce chip temperature by implementing routing algorithms that eliminate and control the temperature proactively. Another way is to throttle the routers. In these kinds of routing algorithms, if the routers' temperature is higher than a threshold, there is a throttling mechanism to lower the routers' temperature. When routing is throttled, the routing algorithm must be reactive to deliver packets to the destination on time.

However, Reactive routing algorithms are activated only when the temperature exceeds the threshold. Thus, these algorithms solve the heat problem by spending the performance penalty. But throttling routing algorithms balance the temperature before the threshold. Therefore, these algorithms are more efficient to achieve better performance.

The main drawbacks of routing mechanisms are as follows: 1- Existing routing mechanisms are performed based on the current status of routers by gathering the current temperature of routers. This data transfer in the router increase performance of a network, especially in critical situations. 2- The threshold of routers is constant and they do not perform dynamically based on the current status of the routers. It reduces the accuracy of the hot region that routing algorithm should not conduct packets.

To solve these drawbacks in this paper, an intelligent routing algorithm called Int-TAR is proposed. Int-TAR controls the temperature of 3D on-chip networks proactively and tries to increase the accuracy of hot regions dynamically. This is done by predicting the temperature of routers based on their past history status and determining the threshold of routers dynamically. In other words in the Int-TAR routing algorithm: 1-in, the first step based on the router status history predictor determines the threshold of routers. 2-Then the hot region is determined dynamically and, the routing algorithm is performed to avoid packets entering the hot area. If the destination node is not lower than the source node, if the source layer is not routable, it routes through the lowest layer, which has no definite router and, then goes to the destination. But if the destination node is lower than the source node, it first tries to reach the corresponding destination node in the source layer. If routing was not possible on the source layer or a router was interrupted during routing, it will send the packet to a lower layer and continue routing on that layer. This process can continue until the bottom layer is reached.

The simulation results show that in this method, the simulation results show that Int-TAR decreases the temperature of the network by 13% and improves performance efficiently.

The main contributions of this paper include:

1. Prediction of the temperature of routers based on the history of routers.
2. Propose a temperature-aware routing algorithm based on the dynamic threshold of routers that let's define the hot region dynamically.

The rest of the paper is organized as follows. In the second section, the concepts of 3D on-chip networks and the challenges of their reliability are discussed. Prior methods on temperature management in 3D on-chip networks are reviewed in section three. In the fourth section, the proposed method described and, the evaluation results of this method are presented in the fifth section. Finally, the conclusion and future work are given in the sixth section.

### Motivation and Backgrounds: Crosstalk Fault

On-chip networks connect processing elements (PEs), memory systems, Application-Specific Integration Circuits (ASICs), and Field Programmable Gate Arrays (FPGAs) through routers and facilitate communications [2]. Shorter connections in 3D NoCs and fewer global signals reduce latency and power dissipation. Fig. 1 shows the architecture and geometry of a 3x3 TSV mesh in 3D NoCs. In [9] and [10], shortening the length of the wires can reduce the delay by up to 30%. Furthermore, reducing the size of cables, repeaters, and repeating latches reduces parasitic capacitors and reduces power consumption. [9] shows that a 3D stacked mathematical unit has 46% less power consumption than a 2D model. Besides, 3D NoCs take up less space due to compact placement in each layer [5], [2].

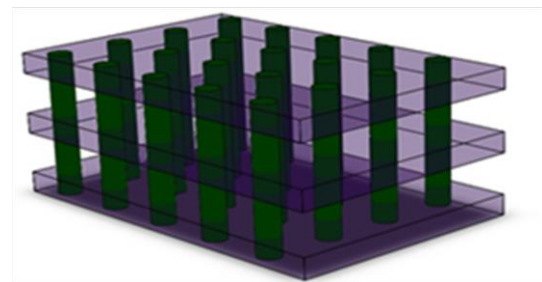


Fig. 1: Architecture of 3D NoC.

Besides the advantages of 3D NoCs, some challenges threaten the reliability. Along with the challenges of 2D chips, we are facing new challenges in 3D networks. For example, the heat challenges in 3D NoCs are more severe than 2D ones because of the different layers on top of each other. Heat problem in 3D NoCs is more potent due to the longer path for heat transfer from the upper layer to the heat sink, more active routers, and more power density [11], [12]. As the temperature increases, the leakage current of the transistor increases, and the power consumption increases accordingly [13].

An increase in power also leads to a quick rise in temperature, causing the elements of the circuit to have intolerable latency and to fail temporarily. If the latency increases so much that it is impossible to perform a calculation or transfer data throughout the clock cycle, that part of the circuit does not operate accurately [14].

The main drawback of high temperature in 3D NoCs is the hot region. As shown in Fig. 2, in 3D NoC, the traffic at the focal point of each layer is more blocked than the other side of that layer. That is because the sides at the middle are responsible for handing off different hub packets between north to south, east to west, and start to finish. Due to the lower number of neighbors for

horizontal hubs, the number of transmitted packets is less. Therefore, focal seats produce more heat, and in each layer, the central part has a higher temperature than the side hubs. Secondly, the side hubs are more relaxed due to better heat transfer. Research shows that if there are hot spots on the chip, packets are not delivered on time. Therefore, network reliability is reduced in the presence of hotspots research shows that if there are hotspots in the network, the rate of packets arriving at the destination decreases. This confirms that if the latency increases, the packages will not transfer on time. Therefore, network reliability is reduced in the presence of hotspots.

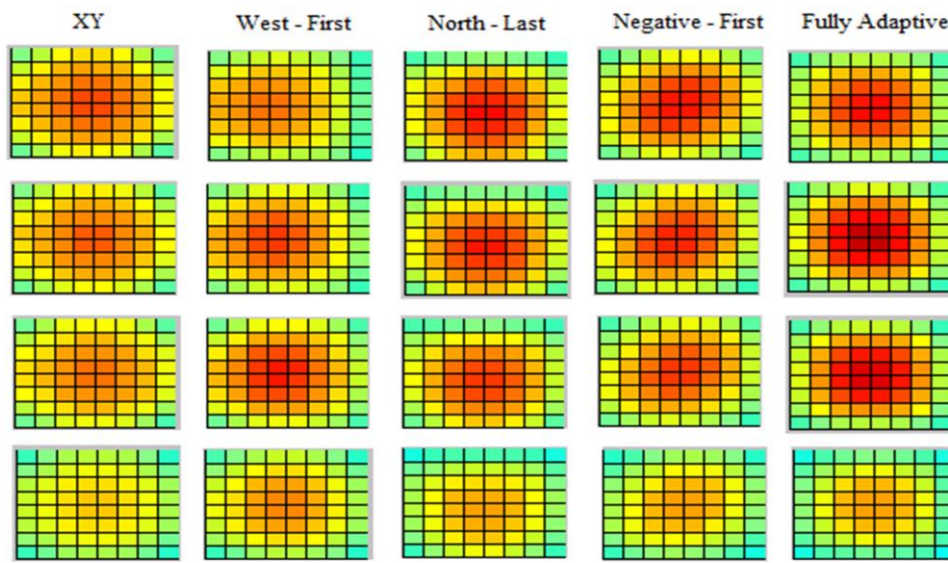


Fig. 2 : Hot regions of a 3D NoC in different routing algorithms under the uniform traffic pattern [26].

## Related Work

Some papers work on 3D NoCs [29]-[33]. There are various methods for solving the thermal problem in 3D NoC that try to manage the network temperature in proactive and reactive manners. Aggressive strategies control the network temperature before reaching the allowed threshold, while reactive methods try to reduce the temperature of hotspots and the timely delivery of packets.

Many recent kinds of research in temperature management have attempted to reduce and control the chip temperature by representing a routing algorithm. [15] presents a traffic-aware routing algorithm (TADW) that tries to bring the 2D layer in which routing is taking closer to the heatsink. In [16], an edge routing algorithm is proposed.

The minimal path between source and destination also provides an edge path that bypasses the minimal path area, often a busy one.

If the router temperature is higher than the threshold, a mechanism cuts it off to reduce the temperature. The most obvious way to disconnect routers is to disconnect all routers if there is even one hot router. Although the method reduces the network temperature faster, it dramatically affects performance. [17] suggests a distributed cut-off method that cuts only hot routers. This method has better performance, but it takes longer to cool the routers. Another technique called temperature-aware vertical cutting is presented in [15], in which the hot router and underneath routers vertically create a channel to conducts heat to the heatsink. There is a higher rate of cooling the routers in this method than a distributed cut-off method, and it is more efficient than the global disconnection method.

When routing is cut-off, the routing algorithm must be reactive to deliver packets on time. In [15], downward routing is presented as reactive routing. In this method, if a separate router is in the routing path, routing is done

through the undermost layer with no different router. Thus the traffic of the lowest layer increases, and efficiency reduces. Therefore, in [18], using the information of other disconnections in the same row and column as a disconnected router, this router bypass quickly in the same layer to reduce the traffic of the undermost layer. However, due to the use of the even-odd rotation model [19], this method can increase the traffic of some columns.

Since reactive routing algorithms are activated only when the temperature exceeds the threshold, they give an efficiency penalty to solve the temperature problem. But pre-routing algorithms dynamically balance the temperature before reaching the point. For this reason, they are a more efficient way to achieve average performance.

In [26], to have a better distribution of heat and traffic in network layers, a method represented that considers the hot areas with static shapes in each layer and routes the packet in such a way as to reduce the rate of entry into the hot zone.

This method introduces a temperature-aware measure that indicates the heat of each router. This method can distribute heat and traffic uniformly in layers.

Moreover, it has more efficiency and reliability compare with prior 3D methods. However, the shape of the hot region is shown. These methods have a large area and memory overhead for the system and cannot provide a uniform distribution of traffic and heat in the network.

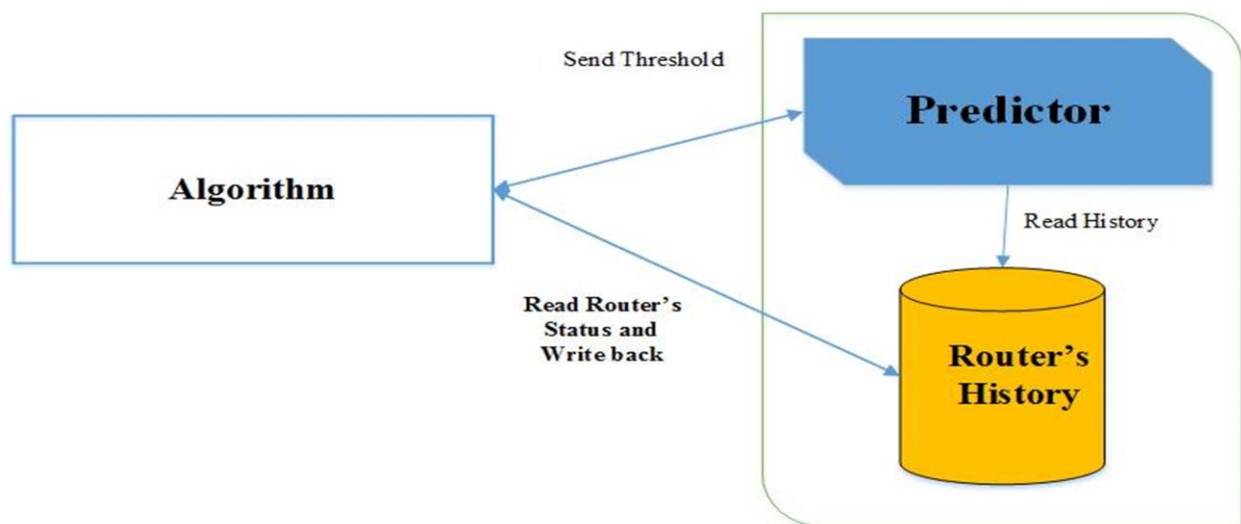


Fig. 3: Main modules of Int-TAR routing algorithm.

### Proposed Intelligent Thermal-Aware Routing Algorithm

Increasing the temperature in the on-chip network is the worst possible reason for eliminating network efficiency. In this case, proactive methods are much more effective than reactive methods. The significant reason is that in these methods, network performance never decreases at once. However, most of the proactive routing algorithms are performed based on the current status of routers by gathering the current temperature of these routers. This data transfer in the router increases the performance of the network, especially in critical situations. Also, the routers' threshold is constant and calculated dynamically based on the current status of the routers.

This reduces the accuracy of hot region calculation, a region where routing algorithms should not conduct packets.

The paper's main idea is to propose an intelligent routing algorithm called Intelligent Thermal-Aware Routing Algorithm (Int-TAR). This idea is the proactive algorithms to reduce the temperature of routers. In general, this idea wants to make an intelligent prediction of the router's temperature status in the future. This routing algorithm increases efficiency in critical situations, accelerates the cooling of routers, and ultimately distributes routing between different routers, resulting in uniform heat distribution throughout the network.

In the ultimate idea of smartening, a predictive module is added to the algorithm, and this module keeps the history of the routers' temperature. At specified intervals based on the temperature history of the routers, threshold values are determined and sent as input to the routing algorithm. The main idea of Int-TAR is shown in Fig. 3.

```

S <- Source;
D <- Destination;
Initialization;
Use History to define hotspot
if D and S in same layer then
    if routable from S to D then
        Intera-Layer routing from S to D ;
    else
        down to the bottom layer;
        Intera-Layer routing to Dj;
        Intera-Layer routing from Dj to D;
    end
else
    if routable from S to Di then
        step1:
        Intera-Layer routing to Di;
        if Di arrival then
            goto step4;
        else
            if routable from C to Di then
                goto step1;
            else
                step2:
                down to the next layer;
                i++;
                if i=n then
                    Intera-Layer routing from Dj to D;
                step3:
                elseif routable frm Ci to Di then
                    Intera-Layer routing to Di;
                    update Ci;
                    if Di arrival then
                        goto step4;
                    else
                        goto step3;
                    end
                else
                    goto step2;
                end
            end
        end
    end
    else
        down to next layer;
    end
step4:
Intera-Layer routing from Di to D;
end

```

Fig. 4: The pseudocode of INT-TAR routing algorithm.



The idea is to create a general module called Predictor, which consists of two parts:

1. Router Status History: this section is the storage of each router's temperature history. It can be said that this module is supposed to save the temperature state of  $N$  at the last moment of each router.
2. Predictor: this part is the central part of the idea. In this subsystem, at certain intervals, the history of  $N$ 's previous states is read from storage. Then the average of these states is calculated and multiplied with a performance factor. The final result is sent as a threshold to the algorithm.

The important point is the efficiency factor at a high temperature of the network. Therefore, networks cool down more slowly, and the performance reduces fairly. The higher the efficiency factor, the faster the cooling of the network. Thus, the efficiency factor is the latency that occurs in routing to select the alternative path. It is done to cool the directions which have routers with high temperature.

The pseudocode of the proposed routing algorithm has shown in Fig. 4, where  $S$  and  $D$  are the source and destination nodes, respectively,  $D_i$  is the corresponding destination (the node that has the same  $X$  and  $Y$  coordinates as the destination node) in layer  $i$ , and  $C_i$  is the current node in layer  $i$ . The routing algorithm comprises two parts: fully adaptive intra-layer routing in the horizontal layer and downward routing in the vertical layer. To avoid deadlock, the routing algorithm firstly routes the packet in the flat layer until the corresponding destination node and then to the destination node in the vertical direction. The turns from up to the four horizontal directions are prohibited. Suppose the packet can't be routed in the current layer. In that case, it is transferred down to the next layer and routed to the corresponding destination again by the intra-layer routing algorithm. This process is repeatedly executed until it is down to the bottom layer. Since the routers in the bottom layer are not throttled, there is at least one path to the corresponding destination.

Finally, the packet can be routed from the corresponding destination in the bottom layer to the destination node. We employ a throttling information collection mechanism to transfer the throttling state to the four neighbor nodes in the horizontal layer. The local router is informed of the node status in two hops. According to the second assumption, there is no need to collect the throttling information in the vertical direction. In each router, there is a 12-bit register for storing the throttling information of the neighbor nodes. The throttling information is represented in a one-bit signal, with "1" representing "throttle" and "0" as "normal."

### Deadlock freeness of Int-TAR

Cyclic waiting for a channel in wormhole switching causes deadlock [20], [21]. There will be a kind of dependency between packets if a packet requests the track taken up by another one. Deadlock occurs when cyclic dependency appears between four requests and for channels. There are two kinds of dependency for each design in a mesh-based network, clockwise and counterclockwise.

To breaking the cyclic dependency, one of the turnings in each cycle should become restricted. As a result, provided the combination of the remaining turns cannot remake the forbidden turn. To experiment this, and prevent never-ending waiting, deadlock-free routing algorithms broadly apply two techniques: 1) using extra virtual channels [22], [23]. 2) Using turn models to prohibit some turns [24], [25].

Besides, to release from deadlock, virtual channels improve the system performance. However, using extra buffer leads to remarkable hardware costs [24]. The proposed routing algorithm employs maximum turning to increase path diversity. In the routing algorithm, due to breaking channel dependency in the vertical direction, the up-south, up-north, up-west, and up-east turns are forbidden. In the horizontal directions, as shown in Fig. 5, the north-east and south-east turns are prohibited. Inconsequence, the routing algorithm is deadlock-free, including no cyclic dependency may occur.

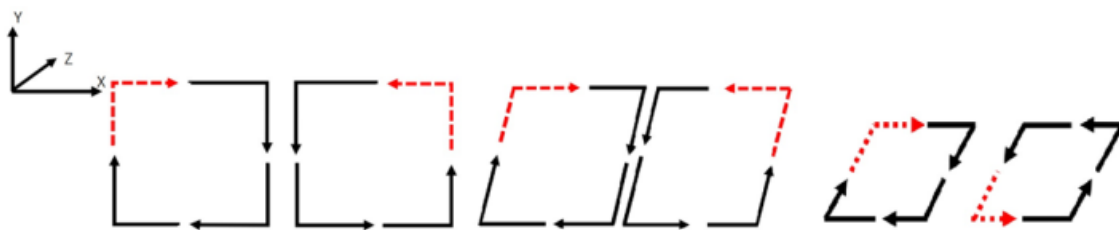


Fig. 5: Turn models [28].

## Results and Discussion

For simulating the Int-TAR, we use the Access NocSim simulator [27]. This is the 3D network simulator, and network parameters can be adjusted for simulations. For implementing the intelligent algorithm, there is a four-thread that works concurrently. The simulation is similar to the existing network. There are many messages based on the functionality of the network that should transfer from source to destination. While network elements are busy and routing the packets, the temperature rises rapidly. It is needed to have a module that monitors and reduces the temperature of the network when the network is busy. This must be done without any significant disruption in the functionality of the network.

To experiment with the real effect, we implemented the idea of Int-TAR algorithm in a heat challenge situation. We ran the algorithm for four hours. Then we simulated the algorithm in a  $4 \times 4 \times 4$  network that routes 100 messages. For fair comparisons, we compare this idea with [26]. In the network, when the packets are sending and receiving concurrently, a module monitors the activities. In the simulation, the network, with a thread for proactive routing algorithm, is busy routing and sending the packets. Another thread called sampler is sampling the status of routers to store their temperature, the third thread reads the stored status and specifies the threshold to determine the cooling rate of the network.

According to the results, when the predicted value for the threshold is high, the routing time increases. In comparison with the basic proactive algorithm, the following points can be stated:

As reported before, the great advantage of a proactive algorithm is the heat distribution over the network. The idea of this paper calculates thresholds dynamically and improves the basic algorithm.

By optimizing the prediction of the threshold, we keep the network throughput high and increase the reliability of the network. This idea shows that the more intelligent algorithm is, the higher reliability the network has.

It can be concluded that the idea of an intelligent algorithm has no destructive effect on proactive algorithms. The intelligent algorithm does not change the main idea of the aggressive algorithms and makes the algorithm more efficient with dynamically calculate the heat threshold. From the heat point of view, we compare the new idea with a proactive algorithm in Fig. 6. In this figure red bar shows [26], and our concept is shown in blue. The figure shows the temperature is to increase about in the same time at two approaches, but the new idea always is in the lower level of the temperature in comparison of the [26]. The time axis shows the duration of the execution of each algorithm. This figure shows that the [26] starts to increase temperature only by a few delay, but in large period causes more temperature than our idea. Fig. 7 shows that proposed algorithm has lower increase in temperature during the certain time. This experiment shows that when in higher thresholds, the routing time increases slightly compared to the original algorithm but has no destructive effect on temperature. Based on our results, the average temperature is of the chip is increased by 13%. We have run our simulation and consider 0.07 packet/cycle. The result of this experiment is shown in Fig. 8. As the results show, the global average delay of the proposed routing algorithm scales with network size and is better than the [26] in different network sizes. To evaluate the area and power overheads of the additional hardware required by the INT-TAR routing algorithm, we use Verilog hardware description language for describing additional hardware and Synopsys R Design Compiler for synthesizing them using CMOS 45 nm technology.

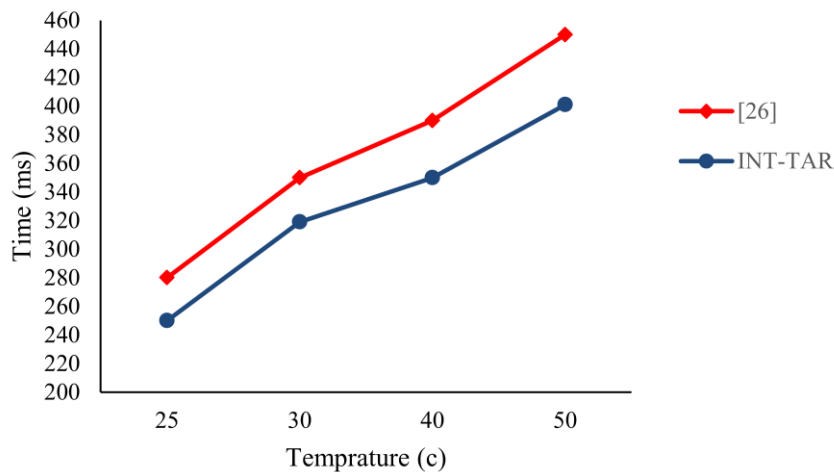


Fig. 6: The temperature of routers with respect to the algorithm of [26].

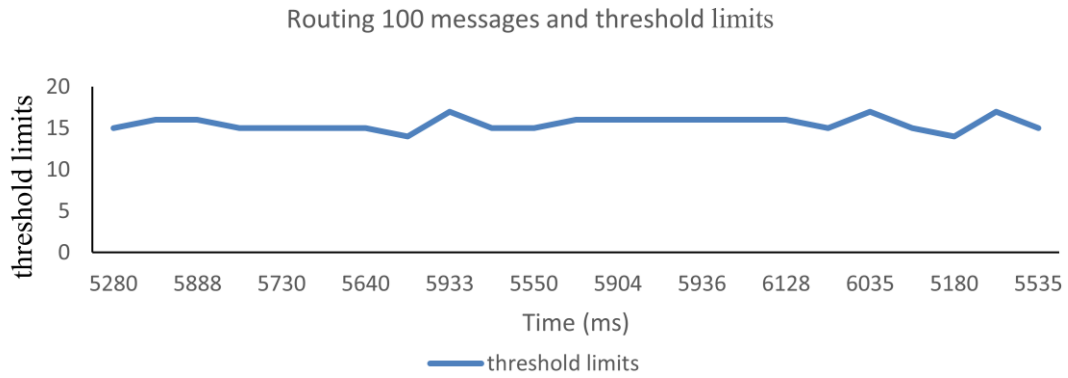


Fig. 7: Threshold limits.

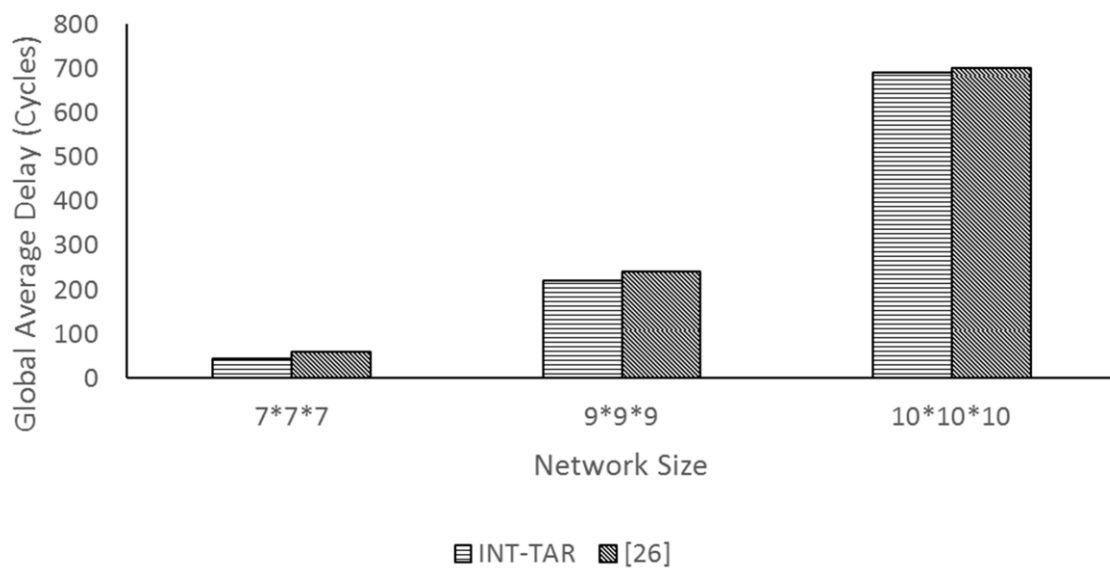


Fig. 8: Global average delay (Cycles) in different network size.

In a typical sized network, for example, an  $8 \times 8 \times 4$  3D NoC, INT-TAR adds register into each router to save the hot region of the layer Table 1 summarizes the results of the area and power consumption for INT-TAR in a 3D router.

As shown, the scope of additional hardware is less than  $15 \mu m^2$ , which imposes less than 0.18% area overhead to an ordinary 3D router with seven ports with a reported location of  $13093 \mu m^2$ .

The power consumption overhead of the additional hardware is at the order of nW, which is negligible compared to the mW order of a 3D router power consumption.

Table 1: Hardware of INT-Tar

	Area ( $\mu m^2$ )	Power (nw)
Additional Hardware	>15	>80
Router $8 \times 8 \times 4$	13093	--
Overhead	0.18	--

## Conclusion

This paper has proposed a routing algorithm called Int-TAR for dynamic thermal management in 3D NoCs. The asymmetric property of mesh topology causes these networks to face thermal difficulties as a significant issue.



In Int-TAR, a prediction of the temperature of routers based on the history of routers is made. We are also proposing a temperature-aware routing algorithm based on the dynamic threshold of routers that lets us define the hot region dynamically.

### Author Contributions

Zahra Shirmohammadi and MohammadJavad Mahmoudi contributed to the idea, simulate and, writing the paper. Maede Rostamnejad contributed for writing the paper.

### Acknowledgment

This work was supported by Shahid Rajaee Teacher Training University.

### Conflict of Interest

The authors declare no potential conflict of interest regarding the publication of this work. In addition, the ethical issues including plagiarism, informed consent, misconduct, data fabrication and, or falsification, double publication and, or submission, and redundancy have been completely witnessed by the authors.

### Abbreviations

NoC	Network On the Chip
PE	Processing Elements
TSV	Through-Silicon Via
ASIC	Application-Specific Integration Circuits
FPGA	Field Programmable Gate Arrays

### References

- [1] V.F. Pavlidis, E.G. Friedman, "3-D topologies for networks-on-chip," *IEEE Trans. Very Large Scale Integr. VLSI Syst.*, 15(10): 1081-1090, 2007.
- [2] R.K. Dash, J.L. Risco-Martin, A.K. Turuk, J.L. Ayala, "A thermal driven genetic algorithm for three-dimensional network-on-chip systems," in *Proc. Summer Computer Simulation Conf.*, 47: 1-8, 2016.
- [3] Z. Zhu, V. Chaturvedi, A. Singh, W. Zhang, Y. Cui, "Two-stage thermal-aware scheduling of task graphs on 3D multi-cores exploiting application and architecture characteristics," in *Proc. IEEE Asia and South Pacific Design Automation Conf. (ASP-DAC)*: 324-329, 2017.
- [4] L. Shen, N. Wu, G. Yan, J. Zhang, F. Zhou, "CTTA: a cluster-based thermal-aware task allocation algorithm for 3D NoC," in *Proc. World Congress on Engineering and Computer Science*: 1-5, 2016.
- [5] R. Salamat, M. Khayambashi, M. Ebrahimi, N. Bagherzadeh, "A resilient routing algorithm with formal reliability analysis for partially connected 3D-NoCs," *IEEE Trans. Comput.*, 65(11): 3265-3279, 2016.
- [6] R. Al-Dujaily, N. Dahir, T. Mak, F. Xia, A. Yakovlev, "Dynamic programming-based runtime thermal management (DPRTM): an online thermal control strategy for 3D-NoC systems," *ACM Trans. Des. Autom. Electron. Syst.*, 19(1): 1-27, 2013.
- [7] C.H. Chao, K.C. Chen, T.C. Yin, S.Y. Lin, A. Wu, "Transport-layer-assisted routing for runtime thermal management of 3D NoC Systems," *ACM Trans. Embedded Comput. Syst.*, (TECS), 13(1): 1:22, 2013.
- [8] G. Yan, N. Wu, F. Ge, H. Xiao, F. Zhou, "Collaborative fuzzy-based partially-throttling dynamic thermal management scheme for three-dimensional networks-on-chip," *IET Comput. Digital Tech.*, 11(1): 24-32, 2016.
- [9] J. Zhao, Q. Zou, Y. Xie, "Overview of 3D architecture design opportunities and techniques," *IEEE Des. Test*, 34(4): 60-68, 2017.
- [10] J. Ouyang, G. Sun, Y. Chen, L. Duan, T. Zhang, Y. Xie, M. Irwin, "Arithmetic unit design using 180nm TSV-based 3D stacking technology," in *Proc. IEEE International Conf. on 3D System Integration*: 1-4, 2009.
- [11] C. Chou, Y. Lin, K. Chiang, K. Chen, "Dynamic buffer allocation for thermal-aware 3D network-on-chip systems," in *Proc. IEEE International Conf. on Consumer Electronics-Taiwan (ICCE-TW)*: 65-66, 2017.
- [12] E. Taheri, A. Patooghy, K. Mohammadi, "XYZ-ZYX: a minimal routing algorithm for dynamic thermal management in 3D NoCs," in *Proc. Iranian Conf. on Electrical Engineering (ICEE)*: 1539-1544, 2016.
- [13] X. Jiang, X. Lei, L. Zeng, T. Watanabe, "Fully adaptive thermal-aware routing for runtime thermal management of 3D network-on-chip," in *Proc. International MultiConference of Engineers and Computer Scientists*: 659-664, 2016.
- [14] K. Banerjee, A. Mehrotra, A.S. Vincentelli, C. Hu, "On thermal effects in deep sub-micron VLSI interconnects," in *Proc. Design Automation Conf. (DAC)*: 885-891, 1999.
- [15] C.H. Chao, K.Y. Jheng, H.Y. Wang, J.C. Wu, A. Wu, "Traffic-and thermal-aware run-time thermal management scheme for 3D NoC system," in *Proc. ACM/IEEE International Symposium on Networks-on-Chip (NOCS)*: 223-230, 2010.
- [16] K. Chen, Sh. Lin, A. Wu, "Design of thermal management unit with vertical throttling scheme for proactive thermal-aware 3D NoC systems," in *Proc. IEEE International Symposium on VLSI Design, Automation, and Test (VLSI-DAT)*: 1-4, 2013.
- [17] L. Shang, L. Peh, A. Kumar, "Thermal modeling, characterization and management of on-chip networks," in *Proc. IEEE/ACM International Symposium on Microarchitecture (MICRO)*: 67-78, 2004.
- [18] S.H. Lin, T.C. Yin, H.Y. Wang, A. Wu, "Traffic-and thermal-aware routing for throttled three-dimensional network-on-chip systems," in *Proc. IEEE International Symposium on VLSI Design, Automation and Test (VLSI-DAT)*: 1-4, 2011.
- [19] G.M. Chiu, "The odd-even turn model for adaptive routing," *IEEE Trans. Parallel Distrib. Syst. (TPDS)*, 11(7): 729-738, 2000.
- [20] C.J. Glass, L.M. Ni, "The turn model for adaptive routing," *J. ACM*, 20: 278-287, 1992.
- [21] J. Duato, "A new theory of deadlock-free adaptive routing in wormhole networks," *IEEE Trans. Parallel Distrib. Syst.*, 4(12): 1320-1331, 1993.
- [22] H.N. Jouybari, K. Mohammadi, "A low overhead, fault tolerant and congestion aware routing algorithm for 3d mesh-based network-on-chips," *Microprocess. Microsyst.*, 38(8): 991-999, 2014.
- [23] F. Liu, H. Gu, Y. Yang, "Dtbr: A dynamic thermalbalance routing algorithm for network-on-chip," *Comput. Electr. Eng.*, 38(2): 270-281, 2012.
- [24] K.C. Chen, S.Y. Lin, H.S. Hung, A.Y.A. Wu, "Topology-aware adaptive routing for nonstationary irregular mesh in throttled 3D NoC systems," *IEEE Trans. Parallel Distrib. Syst.*, 24(10): 2109-2120, 2013.
- [25] A. Patooghy, H. Sarbazi-Azad, "Analytical performance modelling of partially adaptive routing in wormhole hypercubes," in *Proc. Parallel and Distributed Processing Symposium*, 7-13, 2006.
- [26] M. Safari, Z. Shirmohammadi, N. Rohbani, H. Farbeh, "WiP: floating XY-YX: an efficient thermal management routing

algorithm for 3D NoCs," 2018 IEEE 16th Intl Conf on Dependable, Autonomic and Secure Computing, 16th Intl Conf on Pervasive Intelligence and Computing, 4th Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress(DASC/PiCom/DataCom/CyberSciTech): 736-741, 2018.

- [27] K.Y. Jheng, C.H. Chao, H.Y. Wang, A.Y. Wu, "Traffic-thermal mutual-coupling co-simulation platform for three-dimensional network-on-chip," in Proc. IEEE International Symposium on VLSI Design Automation and Test (VLSIDAT): 135-13, 2010.
- [28] J. Duato, S. Yalamanchili, L. Ni, "Interconnection networks: an engineering approach," IEEE CS Press, Los Alamitos, Calif., 1997.
- [29] Z. Shirmohammadi, S.G. Miremadi, "S2ap: An efficient numerical-based crosstalk avoidance code for reliable data transfer of NoCs," in Proc. 10th IEEE International Symposium on Reconfigurable Communication-Centric Systems-on-Chip (ReCoSoC): 1-6, 2015.
- [30] Z. Shirmohammadi, S.G. Miremadi, "On designing an efficient numerical-based forbidden pattern free crosstalk avoidance codec for reliable data transfer of NoCs," Microelectron Reliab., 63: 304-313, 2016.
- [31] Z. Shirmohammadi, F. Mozafari, S.G. Miremadi, "An efficient numerical-based crosstalk avoidance codec design for NoCs," Microprocess Microsyst., 50: 127-137, 2017.
- [32] Z. Shirmohammadi, "Op-fibo: an efficient forbidden pattern free CAC design," Integration, 65: 104-109, 2019.
- [33] Z. Shirmohammadi, Z. Mahdavi, "An efficient and low power one-lambda crosstalk avoidance code design for network on chips," Microprocess Microsyst., 63: 36-45, 2018.

## Biographies



**Zahra Shirmohammadi** received M.Sc. and Ph.D degrees in computer engineering from Sharif University of Technology in 2011 and 2017 respectively. Her current research interests include dependability of System-on-Chip (SoC) and Network-on-Chip (NoC) design and high-performance computer architecture.



**MohammadJavad Mahmoudi** received B. Sc. and M.Sc. degrees in computer engineering from Azad University South Unit and Khajeh Nasir University of Technology (KNTU) in 2009 and 2020 respectively. His current research interests include dependability of System-on-Chip (SoC) and Network-on-Chip (NoC) design and high-performance computer architecture.



**Maede Rostamnejad** received the B. Sc. degree in computer engineering from Shahid Beheshti University in 2016. She is an M. Sc. graduate student in Computer Architecture from Sharif University of Technology (Tehran, Iran). She is currently a researcher in the computer architecture, NoC design, and high performance

computing area.

### Copyrights

©2022 The author(s). This is an open access article distributed under the terms of the Creative Commons Attribution (CC BY 4.0), which permits unrestricted use, distribution, and reproduction in any medium, as long as the original authors and source are cited. No permission is required from the authors or the publishers.



### How to cite this paper:

Z. Shirmohammadi, M.J. Mahmoudi, M. Rostamnejad, "Int-TAR: An intelligent thermal-aware packet routing algorithm for 3D NoCs," J. Electr. Comput. Eng. Innovations, 10(1): 47-56, 2022.

**DOI:** 10.22061/JECEI.2021.7750.428

**URL:** [https://jecei.sru.ac.ir/article\\_1550.html](https://jecei.sru.ac.ir/article_1550.html)

