An Efficient Traffic-Based Routing Algorithm for 3D Networks-on-Chip

Hsueh-Wen Tseng¹, Ruei-Yu Wu²*, Wan-Chi Chang¹, Yi-Huo Lin¹, and Dyi-Rong Duh³
¹Department of Computer Science and Engineering, National Chung-Hsing University, Taiwan
²Department of Management Information Systems, Hwa Hsia University of Technology, Taiwan
³Department of Computer Science and Engineering, Hwa Hsia University of Technology, Taiwan

Abstract

Network-on-Chip (NoC) is proposed to solve the communication challenges in multiprocessor System-on-Chip architecture. 3D NoC uses through silicon vias to stack multiple 2D silicon layers and obtain better performance. However, 3D NoC has to design an efficiently adaptive routing algorithm to overcome huge traffic load problems. An adaptive routing algorithm is composed of routing function (RF) and selection function (SF). RF decides a set of deadlock-free candidate paths, and SF chooses a proper routing path from those candidates. In this paper, we propose a novel traffic-based routing algorithm, TBRA, to achieve better performance. TBRA combines some advantages from different routing algorithms and selects adaptive routing algorithms according to the traffic load. Furthermore, it also collects two hops data of networks to determine the better routing path. Simulation results show that the proposed scheme significantly improves the transmission latency, the throughput, and the energy consumption.

Keywords: Network-on-Chip, 3D IC, Congestion, Selection Function, Routing Algorithm.

1. Introduction

Nowadays, multiprocessor System-on-Chip (MP-SoC) integrates many functional blocks onto a single die [1] and the communication becomes bottleneck between two processor elements (PEs). Network-on-chip (NoC) is proposed to provide a scalable packet switched interconnection architecture for on-chip communication solutions. However, 2D chip architecture significantly increases wire delay and power consumption in the deep submicron regime [2], [3]. 3D NoC vertically stacks many 2D dies by using through-silicon-vias (TSVs) [4], [5] and has shorter global interconnection length, higher performance, and better scalability.

However, 3D NoC generates more traffic load than conventional planar ICs [6]. Higher traffic load easily results in traffic congestion. Conventional planar routing algorithms never suffer such huge traffic load and cannot handle the huge traffic load problem. Typical routing algorithms are classified as deterministic routing algorithms and adaptive routing algorithms based on the path selection process [7]. Deterministic routing algorithms (e.g., XY routing algorithm [8]) use a simple router architecture and predetermine a routing path with deadlock free. Although they obtain lower transmission latency in light traffic load, they cannot avoid the occurrence of hotspots and congestion paths as the packet injection rate (PIR) increases.

On the contrary, adaptive routing algorithms consider the traffic variations of NoC and dynamically calculate proper routing paths to avoid the hotspots and congestion areas [9]. The selection of routing path is determined from the current status of NoC (e.g., congestion paths, hotspot regions, and blocking time). Although adaptive routing algorithms pay slightly higher transmission latency than deterministic routing algorithms in low traffic load due to the hardware overhead of routing implementation, they work well in high traffic load.

Previous works for selection function (SF) only focused on a routing path decision based on the buffer information of adjacent routers or blocking time [10]. However, most of them only took into account how to avoid the adjacent congestion areas when they forwarded packets to destination nodes [11], [12]. These papers only used a partial network information to choose a proper routing path, so packets easily blocked during transmission.

In a general adaptive routing algorithm, a PE generates a packet and the packet is sent to a router for transmitting to a destination. The router uses RF to compute a number of routing paths to forward the packet, and then SF chooses an appropriate output channel (OC) according to network information (i.e., the buffer information of adjacent routers). Once RF selects incorrect routing paths to SF, SF cannot determine the suitable routing path. As a result, it results in traffic congestion problem.

In this paper, we propose a traffic-based routing algorithm (TBRA) to dynamically select suitable adaptive routing algorithms according to the network situations. TBRA consists of a feedback traffic-based routing function (FTBRF) and a traffic-based selection function (TBSF). TBSF uses network information of two hops to precisely determine the proper output port. When TBSF detects the bad situations about the input buffers of neighbors with two hops, TBSF informs FTBRF to dynamically switch routing algorithms depending

*The corresponding author. E-mail: d92007@csie.ntu.edu.tw
on the network situations.

This paper is organized as follows. In Section 2, we describe the related work. Section 3 describes the proposed traffic-based routing algorithm, TBRA. Then, we show our simulation results in Section 4. The conclusions are shown in Section 5.

2. Related work

In this section, we introduce the conventional adaptive routing algorithms and selection function schemes, respectively. The adaptive routing algorithms are classified as fully adaptive routing algorithms and partially adaptive routing algorithms based on the number of selected paths between the source node (S) and the destination node (D). A fully adaptive routing algorithm is presented that a packet is transmitted to use all possible paths from S to D. It has the highest adaptiveness but yields the risk of deadlock.

On the contrary, partially adaptive routing algorithms only use a subset of possible paths to route packets. They have lower adaptiveness and deadlock-free. The turn model [13] and the odd even model [9] are famous partially adaptive routing algorithms without virtual channels. Turn model algorithm limits the minimum number of turns to avoid deadlock, and OE routing is proven to have higher even adaptiveness than the turn model routing. However, OE results in imbalance traffic and then incurs congestion problems in different routing paths. Thus, it is an important issue to design a good selection function to assist adaptive routing algorithms in NoC.

Most of conventional selection functions referred to congestion information, such as Dynamic XY (DyXY) [12], Neighbors-on-Path (NoP) [11], and BARP [13]. They used congestion-aware algorithms to analysis the congestion status of the current router and adjacent routers. According to the obtained network information, each router determines a suitable OC to transmit a packet. DyAD adopted the information of buffer capacity to alleviate traffic congestion among neighboring routers [15]. Although DyAD dynamically switched deterministic and adaptive routing algorithms to alleviate traffic congestion, it only considered the adjacent network information and easily caused misjudgment problems. In literature [16], the authors both consider run time contention and bandwidth-aware selection function in NoC. In literature [17], they proposed a destination-based selection strategy (DBSS) to integrate local and global network information and avoid network congestion. However, these schemes required specific hardware design and only used in 2D NoC.

Briefly, most of previous selection schemes only considered the buffer information of adjacent routers while S sent packets to D. However, these schemes may determine an incorrect routing path and suffer traffic congestion during transmission as shown in Fig. 1.

![Fig. 1: Misjudgment problems of conventional selection function schemes.](image-url)

Fig. 1 illustrates misjudgment problems in a mesh NoC. We assume the design of the router uses wormhole scheme, and the input buffer size is set to 4 flits without virtual channel (VC). When S wants to transmit packets, it considers the buffer information of the adjacent routers. There are two candidate paths: path 1 and path 2. Path 1 uses router 4 to forward packets, and path 2 uses router 8 to forward packets. The used input queue lengths (UIQLs) of router 4 and router 8 are 2 and 3. Hence, S selects path 1 to forward packets. There is a similar procedure in router 4. Router 4 finds the UIQLs of router 1 and router 5 are 4 and 3. Hence, router 4 forwards packets to router 5. However, router 5 suffers congestion, and packets are blocked in router 5. Although the situation of path 2 seems to be not a good selection from the UIQLs of neighbor routers, path 2 is a better selection than path 1 because router 9 has shorter UIQL and quickly forwards packets of router 8.

Hence, we should use the UIQLs of adjacent routers (one hop) and more information from neighbors of adjacent routers (two hops) to reduce the misjudgment problem.

The researches of previous 2D NoC only took into account single RF or single SF mechanism. Previous works for 3D NoC exploited hybrid routing algorithms to balance traffic and avoid throttled routers in the chip [18]. However, these schemes were only based on throttling information and should consider congestion situations to effectively transmit packets.

3. Traffic-based routing algorithm

In this paper, we propose a traffic-based routing algorithm (TBRA) to dynamically select suitable adaptive routing algorithms according to the network situations and achieve maximize $T_{PE}$, minimize $L_{packet}$, and $E_{system}$. $T_{PE}$ is the average throughput per processor element (PE), $L_{packet}$ is the average latency per packet, and $E_{system}$ is the total power consumption of the system.

TBRA consists of a routing function (FTBRF) and a selection function (TBSF). FTBRF combines the features of two adaptive routing algorithms (i.e., WX and OX routing...
algorithms. Most adaptive routing algorithms usually extend from OE and turn mode (i.e., WF routing). OXD and WXD [18] are typical hybrid routing algorithms based on OE [9] and WF [13] routing algorithms, respectively.

TBSF uses a packet forwarding direction table (PFDT) to record the UIQL of each router. The larger UIQL is presented that the router has heavier traffic load and higher routing cost (e.g., larger transmission latency). According to PFDT, TBSF selects the minimum cost routing path (MCRP) to forward packets. MCRP is the least congested routing path for forwarding packets from S to D. Once TBSF obtains the bad routing paths from the original routing algorithm of FTBRA, TBSF sends a feedback to FTBRF. FTBRF dynamically changes to other routing algorithm to obtain better routing paths to transmit the following packets.

Conventional adaptive routing algorithms dynamically determine the proper routing paths and balance traffic in 3D NoC, especially for high traffic load. To observe the features of different adaptive routing algorithms with different traffic loads and traffic patterns in 3D NoC, we efficiently use the features of different routing algorithms (i.e., OXD and WXD) to transmit packets. OXD is based on OE routing, and a router first uses OE routing. If OE routing algorithm cannot be used in a router because of some directional limitations, the router switches to XY routing algorithm to find out other routing paths. If the router still cannot use XY routing algorithm to forward packets, packets are transmitted to below layer by downward routing algorithm. WXD also has the similar routing processes to OXD.

Fig. 2 and Fig. 3 show the features of OXD and WXD including average latency, throughput, and total energy in two most popular synthetic traffic patterns: random traffic and transpose1 traffic [9], [13], [15] by using AccessNoxim [19]. The topology size is set to $8 \times 8 \times 4$, and the packet length is fixed to be 8 flits to neglect the impact of different packet lengths. In random traffic, each source has the same probability to send a packet to each destination.

In transpose1 traffic, the coordinates of a destination node $(mesh_x-s.y-1, mesh_y-s.x-1, s.z)$ transpose x coordinate and y coordinate of the source node $(s.x, s.y, s.z)$. $mesh_x$ and $mesh_y$ are the size values of x coordinate and y coordinate from the topology size. Our simulation is from 0.001 to 0.039 flits per cycle with each scale 0.002.

When the PIR is less or equal to 0.02 flits per cycle, we define the situation as 'low' traffic load. On the other hand, when the PIR is higher than 0.02 flits per cycle, the situation is defined as 'high' traffic load.

As shown in Fig. 2 (a), (b), and (c), the performance of WXD is better than that of OXD in random traffic. Since WXD is based on WF, WF has better performance in random traffic [9]. In Fig. 4 (a), (b), and (c), OXD has lower latency and higher throughput when the PIR is less than 0.029 flits per cycle. As the PIR increases, WXD obtains lower average latency and higher throughput. WXD also has lower energy consumption between two routing algorithms.

From Fig. 3 and Fig. 4, we find that OXD and WXD appear different performance in different traffic loads and traffic patterns. Thus, we should choose the suitable routing algorithms according to the network situations to obtain better performance in 3D NoC. Routing algorithm (RA) can be expressed as the functions of $L_{packet}$, $T_{PE}$, and $E_{system}$ by

$$L_{packet} = \frac{\sum_{i=1}^{p_{num}} L_i(RA)}{p_{num}},$$

$$T_{PE} = \frac{\sum_{i=1}^{Com\_PE_{num}} T_i(RA)}{Com\_PE_{num} \times Total\_cycles},$$

$$E_{system} = \frac{\sum_{i=1}^{Com\_PE_{num}} E_i(RA)}{Total\_cycles},$$

where $L_i(RA)$, $T_i(RA)$, and $E_i(RA)$ are the latency function, throughput function, and energy function of RA, respectively. $p_{num}$ is the number of the transmitted packets. $Com\_PE_{num}$ is the number of received packets by PE.
Fig. 4: The switching problem occurs in even columns when RFs of routers switch routing algorithms between OE and WF.

Fig. 5: Traffic-based routing algorithm (TBRA).

which is communicating to other ones. Total cycles is the total simulation cycles. In subsection 3.1, we present the proposed FTBREF. TBSF and PFDT are described in subsection 3.2.

3.1 Feedback Traffic-based Routing Algorithm (FTBRA)

From Fig. 2 and Fig. 3, OXD appears better performance in low traffic load, and WXD is suitably used in high traffic load. According to the features of Fig. 2 and Fig. 3, a router should select a suitable routing algorithm based on different traffic loads and traffic patterns. However, two issues should be considered. First, we need to obtain the traffic status of a router. Second, we have to decide the switch conditions between the routing algorithms.

Fig. 4 shows the switching problems in even columns when RF of current router changes routing algorithms between OE and WF. The gray blocks show that OE routes packets in some directions but WF does not use these directions. The packets are dropped seriously when routers change from OXD routing to WXD routing, especially for high traffic load. Similarly, RF of current router changes from WXD to OXD, it still encounters the switching problem.

Fig. 5 is the illustration of TBRA. TBRA uses FTBREF to forward packets, and TBSF decides MCRP from a set of candidate output ports which are selected by using FTBREF. We use a congestion counter (CC) in the arbiter to show the situations of network congestion. Initially, CC is set to 0. We also use a traffic threshold (bufferth) to define the traffic status of a router in TBSF. bufferth presents that the UIQL of a router is X flits. In this paper, the value of X is set to 4. In addition, we use a switching threshold (δ) to switch different routing algorithms of a router. Initially, δ is set to 0.

FTBREF first checks the value of CC in a router. If CC is less than or equals δ, the router uses OXD routing. Once CC is larger than δ, the router switches to WXD. As shown in Fig. 5, we add an arbiter in each router, and the arbiter detects the buffer information of adjacent routers (the neighbors of the current router (CN)) and neighbors of next routers (the neighbors of next routers (NN)) from PFDT.

If both the UIQLs of CN and NN are less than the threshold (bufferth), CC is set to 0. Contrarily, the value of CC increases one and notifies RF. To compare with the value of CC, RF of the current router dynamically uses OXD or WXD, and TBSF selects MCRP according to the buffer information of PFDT to forward packets.

Fig. 6 is the flow chart of TBRA, and it consists of two blocks. The first block is illustrated the operations of FTBREF, and the second block is shown the operations of TBSF. As shown in Fig. 6, a router first checks the value of CC. If CC is less than or equals δ, the router uses OXD routing. Once CC is larger than δ, the router switches to WXD. As shown in Fig. 6, we add an arbiter in each router, and the arbiter detects the buffer information of adjacent routers (the neighbors of the current router (CN)) and neighbors of next routers (the neighbors of next routers (NN)) from PFDT.

If both the UIQLs of CN and NN are less than the threshold (bufferth), CC is set to 0. Contrarily, the value of CC increases one and notifies RF. To compare with the value of CC, RF of the current router dynamically uses OXD or WXD, and TBSF selects MCRP according to the buffer information of PFDT to forward packets.

Fig. 6 is the flow chart of TBRA, and it consists of two blocks. The first block is illustrated the operations of FTBREF, and the second block is shown the operations of TBSF. As shown in Fig. 6, a router first checks the value of CC. If CC is less than or equals δ, the router uses OXD routing and TBSF finds MCRP according to the information of PFDT. Once CC is larger than δ, the routing path appears traffic congestion. Hence, the router should use WXD routing to obtain better performance from Fig. 2 and Fig. 3. Our FTBREF dynamically switches the routing algorithms between OXD routing and WXD routing algorithms according to the traffic situations. Then, TBSF selects a proper routing path with MCRP.
calculates a proper routing path with MCRP. The decision checks the buffer information of two hops from PFDT. TBSF progresses [21]. Improved to use the technology of routing table minimization increases. In the future, the scalability problems will be the scalability of PFDT is not good when the topology size [20]. The warm-up time is to collect PFDT information. Of NoC system is 1000 cycles based on default configuration [20]. The warm-up time is to collect PFDT information.

As shown in Fig. 7(b), there are n columns, each column is \( \lceil \log_2(\text{buffer size}) \rceil + 1 \) bits, where buffer size is the input buffer size. The column is denoted the flit number of each output port in a router. Normally, a router has seven output ports in 3D NoC: north port (NP), east port (EP), south port (SP), west port (WP), up port (UP), down port (DP), and local port (LP). Each row is denoted ID of each PE in 3D NoC. The topology size of 3D NoC is set to \( dx \times dy \times dz \) (8 \( \times 8 \times 4 \)), the number of PEs is \( dx \times dy \times dz \), and the size of PFDT is \( \lceil \log_2(\text{buffer size}) \rceil + 1 \) ndx\( \times dy \times dz \) bits. However, the scalability of PFDT is not good when the topology size increases. In the future, the scalability problems will be improved to use the technology of routing table minimization progresses [21].

When S wants to transmit packets to D, TBSF first checks the buffer information of two hops from PFDT. TBSF calculates a proper routing path with MCRP. The decision function of the best forwarding direction \( f_{\text{Best Direction}} \) is given by \( \text{Cost}_{\text{current}} + \text{Cost}_{\text{next node}} \), where \( \text{Cost}_{\text{current}} \) is the routing cost of the current router and \( \text{Cost}_{\text{next node}} \) is the routing cost of next routers. \( \text{Cost}_{\text{current}} \) and \( \text{Cost}_{\text{next node}} \) are calculated based on the buffer information of PFDT. As S sends packets, S uses TBSF to calculate MCRP and alleviate the misjudgment problems based on more accurate network information. However, if there is congestion occurrence with two hops, the selected MCRP may be bad. When CC is larger than \( \delta \), TBSF notifies FTBRF to choose another routing algorithm. FTBRF switches OXD routing to WXD routing to find other better routing paths.

### 4. Experiment Results

We use AccessNoxim [19] to show the performance of TBRA, and the simulation parameters refer to literature [22], [23], [24]. AccessNoxim consists of Noxim [20] and Hotspot [25] as well as models the 2D NoC system in literature [26]. The router uses the design of wormhole switching without virtual channel, and \( \delta \) is set to 0. To evaluate the performance, two major synthetic traffic patterns are used: random traffic and transpose1 traffic [9], [13], [15]. The primary performance metrics are average latency, throughput, and total energy consumption in the system.

We design two experiments to verify the performance of TBRA. First, we compare the average latency, throughput, and total energy consumption with OXD-NoP in subsection 4.1. OXD is based on OE, and OE is proven to be the best routing algorithm in 2D NoC with non-uniform traffic [9]. NoP is proven to be the best selection function in 2D NoC [11]. In subsection 4.2, we improve the switching problem by adjusting the switching threshold \( \delta \) in different traffic loads and traffic patterns.

#### 4.1 The Performance of TBRA

As shown in Fig. 8(a), (b), and (c), the performance (average latency, throughput, and total energy) of TBRA sharply decreases as PIR increases. TBRA uses OXD routing in light traffic load and then switches to WXD routing in high traffic load. When routers switch to other routing algorithm in even columns, different directional limitations between OE and WF result in performance degradation in high traffic load.

Fig. 9 divides into two parts (the PIR is below 0.027 flits per cycle and above 0.027 flits per cycle) to discuss the performance of TBRA. When the PIR is below 0.027 flits per cycle, TBRA has a little higher average latency and higher total energy than OXD-NoP. The reason is that TBRA uses more complex hardware design to collect more network information. As the PIR is above 0.027 flits per cycle, TBRA improves 17.62% average latency and 0.57% throughput. However, TBRA has a little higher total energy (0.27%) because of the complex hardware design.

#### 4.2 The switch threshold of FTBRA

We adjust the switching thresholds \( \delta \) to improve the switching problem. Fig. 10 and Fig. 11 show the impact of different switching thresholds \( \delta \) for TBRA performance in.

---

**Fig. 7: The packet forwarding direction table (PFDT).**

**3.2 Traffic-based Selection Function (TBSF)**

Most of conventional selection strategies only took into account the buffer information of adjacent routers to choose a proper routing path but they resulted in the misjudgment problems (see Fig. 1). However, TBSF finds MCRP according to the information of PFDT. PFDT collects more information to select a more suitable routing path and improve the misjudgment problems.

Fig. 7 is the illustration of PFDT. In Fig. 7(a), we use the hardware design of NoP [11] to gather the buffer information of neighbors with two hops, and then we record the information of each router on PFDT. The warm-up time of NoC system is 1000 cycles based on default configuration [20]. The warm-up time is to collect PFDT information.

As shown in Fig. 7(b), there are n columns, each column is \( \lceil \log_2(\text{buffer size}) \rceil + 1 \) bits, where buffer size is the input buffer size. The column is denoted the flit number of each output port in a router. Normally, a router has seven output ports in 3D NoC: north port (NP), east port (EP), south port (SP), west port (WP), up port (UP), down port (DP), and local port (LP). Each row is denoted ID of each PE in 3D NoC. The topology size of 3D NoC is set to \( dx \times dy \times dz \) (8 \( \times 8 \times 4 \)), the number of PEs is \( dx \times dy \times dz \), and the size of PFDT is \( \lceil \log_2(\text{buffer size}) \rceil + 1 \) ndx\( \times dy \times dz \) bits. However, the scalability of PFDT is not good when the topology size increases. In the future, the scalability problems will be improved to use the technology of routing table minimization progresses [21].

When S wants to transmit packets to D, TBSF first checks the buffer information of two hops from PFDT. TBSF calculates a proper routing path with MCRP. The decision function of the best forwarding direction \( f_{\text{Best Direction}} \) is given by \( \text{Cost}_{\text{current}} + \text{Cost}_{\text{next node}} \), where \( \text{Cost}_{\text{current}} \) is the routing cost of the current router and \( \text{Cost}_{\text{next node}} \) is the routing cost of next routers. \( \text{Cost}_{\text{current}} \) and \( \text{Cost}_{\text{next node}} \) are calculated based on the buffer information of PFDT. As S sends packets, S uses TBSF to calculate MCRP and alleviate the misjudgment problems based on more accurate network information. However, if there is congestion occurrence with two hops, the selected MCRP may be bad. When CC is larger than \( \delta \), TBSF notifies FTBRF to choose another routing algorithm. FTBRF switches OXD routing to WXD routing to find other better routing paths.
random traffic and transpose1 traffic. The $i_{th}$-TBRA denotes that the value of CC is larger than $i$ to change routing algorithm from OXD routing to WXD routing. The larger value of CC is present that the duration of traffic congestion has continued CC cycles.

Our TBRA uses the features of OXD and WXD to improve the system performance. In Fig. 10, we find that the larger switching threshold reduces the number of switch algorithm from OXD routing to WXD routing. When we use a larger $\delta$, the performance of TBRA approaches that of OXD-NoP.

From Fig. 4, if two conditions are satisfied simultaneously, the switching problem occurs. One is that the current router switches to other routing algorithm. The other is that the switching position appears in even columns. From Fig. 10, we find that the $10_{th}$-TBRA appears unstable states. When the PIRs are equal to 0.027, 0.033 and 0.035 flits per cycle, we find that the duration of traffic congestion is larger than 10 cycles, and many hotspots appear in even columns. In addition, a router easily changes from OXD routing to WXD routing due to the smaller threshold (i.e., $\delta=10$). Hence, the switching problem occurs, and the $10_{th}$-TBRA suffers serious throughput degradation.

The $0_{th}$-TBRA and the $10_{th}$-TBRA are not the proper switch thresholds because most duration of traffic congestion is larger than 10 cycles. When the PIR is below 0.029 flits per cycle, the $30_{th}$-TBRA has a little higher average latency and higher total energy than OXD-NoP. The reason is that TBRA uses more complex hardware design to collect more network information. When the PIR is above 0.029 flits per cycle, the $30_{th}$-TBRA improves 6.11% average latency. However, the $30_{th}$-TBRA has a little lower throughput (-0.17%) and consumes lower total energy (0.16%). As $\delta$ equals 40, the performance of the $40_{th}$-TBRA is the same as the $30_{th}$-TBRA. Fig. 10 shows that the better value of $\delta$ is set to 30 in random traffic.

In Fig. 11, we can find that the $0_{th}$-TBRA has the lowest average latency in transpose1 traffic, and the performance of the $30_{th}$-TBRA approaches to that of OXD-NoP. Summarily, TBRA exploits OXD routing in light traffic load and then switches to WXD in high traffic load. It obtains better performance than that of OXD-NoP in transpose1 traffic with high traffic load. Although TBRA with $\delta=0$ has worse performance in random traffic, the traffic distribution of the chip rarely appears random traffic in realistic 3D NoC. We also improve the performance in random traffic by adjusting the switching threshold. The better value of $\delta$ is set to 30 in random traffic, and $\delta=0$ is a better choice in transpose1 traffic.

5. Conclusion

In this paper, we propose a TBRA to combine FTBRF with TBSF. FTBRF dynamically uses OXD or WXD routing depending on traffic load. TBSF exploits more accurate network information to improve the network congestion problems.

We use TBRA to obtain the advantages of hybrid routing algorithms and address better performance in different
traffic loads and traffic patterns. According to the simulation results, we find that TBRA show the better performance in transpose 1 traffic with high traffic load. We adjust switching threshold to improve the switching problem. The better value of $\delta$ is set to 30 in random traffic, and $\delta = 0$ is a better choice in transpose 1 traffic.

Therefore, the designers use different routing algorithms and different switching thresholds to achieve better performance in different traffic loads and traffic patterns. TBRA is a better hybrid routing algorithm for enhancing performance in 3D NoC.

Acknowledgment

The authors would like to thank the Ministry of Science and Technology of the Republic of China, Taiwan, for financially supporting this research under Contract No. MOST 104-2221-E-005-011 and 104-2221-E-146-003-. We also thank to Access Lab for sharing the code of Access-Noxim simulator.

References


