



# A Deadlock-Free and Adaptive Prime Perspective Turn Model for 3D-Mesh Based Network-on-Chips

Pradeep Kumar Sharma<sup>1(✉)</sup>, Pinaki Mitra<sup>1</sup>, and Santosh Biswas<sup>1,2</sup>

<sup>1</sup> Indian Institute of Technology (IIT) Guwahati, Assam, India  
cspradeepindia@gmail.com, pinaki@iitg.ac.in

<sup>2</sup> Indian Institute of Technology (IIT), Bhilai, India  
santosh@iitbhilai.ac.in

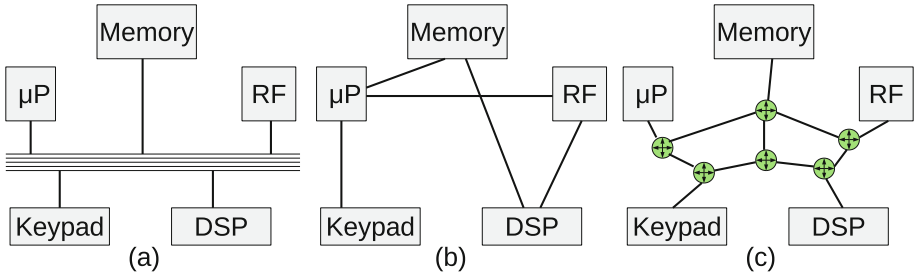
**Abstract.** With the tremendous demands for core-based digital systems, routing algorithms play a key role in improving metrics like performance parameters (delay, throughput) and cost effectiveness of network-on-chips (NoCs). A 3D-mesh-based NoC essentially stacks many layers of 2D-meshes integration to achieve greater performance, a high integration density, shorter interconnects, and lower power consumption. We propose a 3D-mesh routing model that is adaptive, deadlock-free, and that forbids certain turns depending on the prime number of stacks up layer as well as vertical and horizontal directions without virtual channels. The proposed routing model extends the two dimension repeating turn model (RTM-2D) into 3D-mesh by applying certain additional rules based on prime perspective, and it balances the degree of adaptiveness accordingly as prime perspective plane. According to the simulation results, our routing method outperforms in terms of performance metrics to the six typical routing systems in the uniform, transpose, and hotspot cases.

**Keywords:** Network-on-Chip (NoC) · PTM · Turn model · Mesh topology · Deadlock-freedom

## 1 Introduction

The adjacent processing cores communicate with one another through a number of topological routers in a network-on-chip (NoC), which is an organised system of point-to-point connections. The structure of network communication is depicted in Fig. 1 from [2, 18] as follows: (a) traditional bus-based network communication among cores of SoCs where only one core can communicate at a time in the network system, (b) point-to-point interconnection links where each core dedicated to each other via a link, and (c) router-based network communication called NoC where each router either pass the communication to a local core or it passes to the next destined address. NoC overcomes the limitations of standard System-on-Chips (SoCs) bus-based communication architectural system by providing a scalable, reuseable, and parallel communication platform for large applications.

High-performance real-time and application processors, a dedicated graphics core, and programmable logic are all combined into one unit to produce an MPSoC (MultiProcessor System on Chip), which delivers a high-performance processing system [10, 14]. Efficient task mapping [18, 22] and routing model [4, 16, 21, 23] can help greatly to reduce the overall communication energy and communication overheads.



**Fig. 1.** Network communication structures (a) traditional bus, (b) dedicated point-to-point interconnection, and (c) router-based interconnection

To construct a deadlock-free routing algorithm, there are 12 possible ways to prohibit two combinations of turns in a  $2 \times 2$  mesh topology as shown in Fig. 2. Consequently, for a  $10 \times 10$  2D-mesh topology, it has  $(12)^{81}$  possible routing ways because it contains total  $(10-1) \times (10-1) = 81$   $2 \times 2$  subnetworks. To design a deadlock-free and highly adaptive high performance routing algorithm from such a high search space is an NP-hard problem.

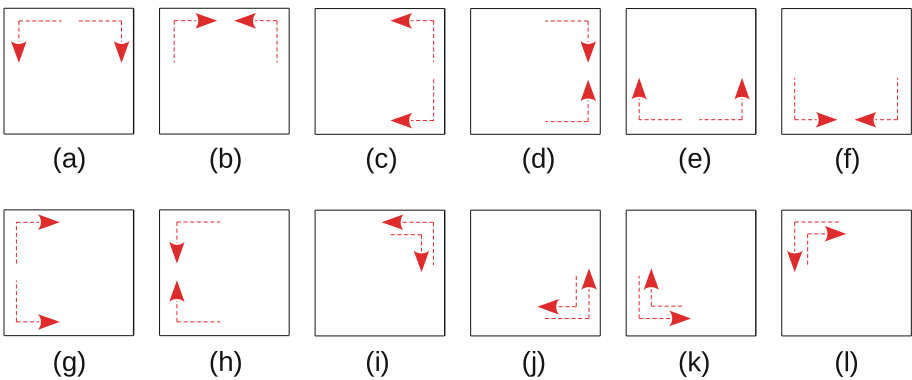
The 2D-mesh-based NoC is the most commonly accepted topology. However, in order to achieve higher performance, recent trends suggest that a network’s routing algorithm is slightly moving to 3D-mesh on integration of stacks over 2D-meshes. The routing method can be implemented in two different ways: table-based (look-up table) and logic-based (combination circuit). In terms of area, power, latency, and throughput; Dimension order Routing (DoR) [8], turn model routing [12], Odd-even turn model [6], balanced-plane odd-even turn model [7], repetitive turn model (RTM-2D) [21], RTM-3D [4] are the most basic examples of logic-based routing algorithms. Table-based approach maintains a look-up table at each switch/router to store routing paths for its every destination pairs. It is implemented to any topology but has poor scalability. SR [17], PM [20], LBDR [11] are examples of table-based routing algorithms.

Since RTM-3D [4] (3D extension of RTM-2D) outperforms over BOE, OE-3D, NF-3D turn models in 3D-mesh network. So, the fundamental objective is to define a new deadlock-free and adaptive routing algorithm that outperforms RTM-3D and the earlier turn models for 3D-mesh networks.

Following is the arrangement of the remaining sections. We provide a brief summary of the associated work for the forbidden turn model in Sect. 2. Prohibited turn model (PTM-3D) routing and an illustrated example are found in Sect. 3. The comparative simulation results of six fundamental routing methods are shown in Sect. 4. Finally, Sect. 5 concludes the paper work.

## 2 Related Work

A topology which has low diameter, low average distance, high bandwidth and low node-degree, doesn't mean that it always performs better for any applications. Routing algorithms classified into several categories based on the nature of application problems [1,2]. The most famous Dimension order Routing (DoR) [8] is a fully deterministic XY routing in which the packet always moves in x-direction first until x-direction vanishes if needed, and then the packet moves in y-direction if needed. The very first paper based on deadlock-free adaptive turn model routing in 2D-mesh architecture proposed by Glass *et al.* in 1992 [12]. The *XY*, *West-first*, *North-last*, and *Negative-first* routing algorithms are deadlock-free and not using any virtual channel or buffer. A new partially adaptive routing technique based on Odd-Even columns turned in a 2D-mesh was proposed by chiu *et al.* in 2000 [6]. It slightly solves the adaptiveness problem of turn model routing and it also works without using virtual channels and buffers in the router design architecture. By proposing two additional modified odd-even rules, Dahir *et al.* developed a balanced degree of adaptiveness odd-even routing for 3D-mesh network in 2012 [7]. PAAD (partially adaptive and deterministic routing) is a switch based routing: *deterministic* when there is no congestion and *partially adaptive* when there is congestion in 2D-mesh based network [15]. The RTM-2D [21] repetitive prohibited turns are based on *column%3* instead of odd-even so that the maximum column repetitive distance increases and the routing pressure [19] decreases. The deadlock-free routing models, prime turn model and



**Fig. 2.** The deadlock-free routing algorithms for 2D-mesh ( $2 \times 2$ ) topology

last first turn model [16], and column/row-partitioning routing model [3] implemented on 2D-mesh topology and their performances are slightly better than odd-even routing turn model. In 3D mesh based NOCs, FT-DyXYZ, an adaptive fault tolerance routing that employs proximity congestion information to balance traffic, may tolerate permanently damaged links which do not require routing tables, or global information about pathways and defects [13]. RTM-3D [4] is basically an extension of repetitive turn model by deploying two addition rules for inter-layer prohibited turn. It shows the performance result over basic turn model routing and odd-even for 3D-mesh network. Adaptive thermal-aware routing (ATAR) [9] is thermal-aware prohibited turn based deadlock-free routing which can also alleviate the peak temperature.

Therefore, we got the motivation as well as set the objective to design a new routing algorithm called, prime perspective turn model (PTM) which has the longest MRD distance, low routing pressure, and better performance metrics (average packet delay (cycles), throughput (flits/cycle/node) to others routing algorithms, RTM-3D and MRD-3D, OE-3D and others 3D routing algorithms.

### 3 Proposed Model

For a 3D-mesh based network topology, a particular node is identified by a three-element-vector  $(x_c, y_c, z_c)$ , where  $x_c$  denotes coordinate along x-axis,  $y_c$  denotes coordinate along y-axis, and  $z_c$  denotes coordinate along z-axis. Each node can be labeled(l) by the following formula:  $l = x_c + y_c * N + z_c * N * N$ . Assume that N is the size of each dimension. Each node  $(x_c, y_c, z_c)$ , therefore meets the condition that  $0 < x_c, y_c, z_c < N$ . Each node in the same plane has the same value of  $z_c$ .

**Definition 1.** A prime number (or prime) is a natural number that is divisible exactly only by one (1) and the number itself.

From the above definition,  $\text{isPrime}(z_c)$  is a function which returns *true* if  $z_c$  is prime, otherwise returns *false*.

**Definition 2.** Routing,  $\langle S, D \rangle$  is the higher-level decision-making process that determines the path for traffic within a network, between networks, or across multiple networks from the source node (S) to the destination node (D) through intermediary network nodes (routers/switches/gateways).

#### 3.1 PTM-3D

PTM-3D (Prime perspective Turn Model for 3D-Mesh), which restricts turns based on prime position of coordinate axes, is the proposed deadlock-free routing algorithm for 3D-mesh based NoCs. PTM-3D is a combination of RTM-3D (which is an extension of RTM-2D), OE-3D, MRD-3D and BOE-3D. A message/packet forwards possibly in six available directions: **Up** ( $z_e > 0$ ), **Down** ( $z_e < 0$ ), **North** ( $y_e > 0$ ), **South** ( $y_e < 0$ ), **East** ( $x_e > 0$ ), and **West**

( $x_e < 0$ ). When a new message is injected into 3D-mesh network then the proposed PTM-3D routing is following turn rules to avoid deadlock in layer to layer communication,

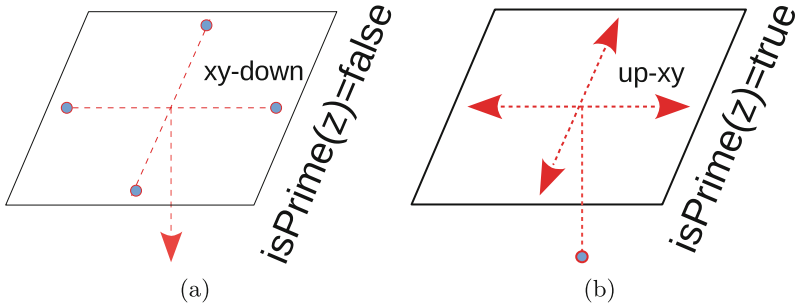


Fig. 3. Prohibited turn: (a) xy-down turn, (b) up-xy turn

- **Rule 1:** A message is not allowed to take a turn at any node  $(x_c, y_c, z_c)$  in xy-down direction, if the current layer  $z_c$  is not a prime, i.e.,  $isPrime(z_c) = false$  as shown in Fig. 3(a)
- **Rule 2:** A message is not allowed to take a turn at any node  $(x_c, y_c, z_c)$  in Up-xy direction, if the layer  $z_c$  is a prime, i.e.,  $isPrime(z_c) = true$  as shown in Fig. 3(b).

To avoid the deadlock, the injected message follows Rules 1 and 2 in inter-layers communication. For example, the injected message is prohibited from *xy-plane-to-down* direction at  $z_c = 1, 4, 5, 6$  by Rule 1, and it is prohibited *Up-to-xy-plane* at  $z_c = 2, 3, 5$  by Rule 2 as shown in Fig. 3.

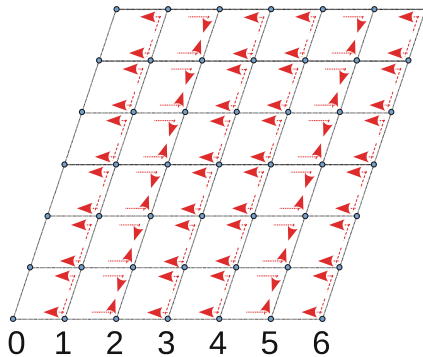


Fig. 4. Prohibited turn: column (x-axis) perspective turn

- **Rule 3:** Any message is not allowed to take a turn at any node  $(x_c, y_c, z_c)$  in North-West (NW) and South-West (SW) turns, if the column  $x_c$  is not a prime number, i.e.,  $isPrime(x_c) = false$  as shown in Fig. 4.
- **Rule 4:** Any message is not allowed to take a turn at any node  $(x_c, y_c, z_c)$  East-North (EN) and East-South (ES) turns, if the coordinate  $x_c$  is a prime number, i.e.,  $isPrime(x_c) = true$  as shown in Fig. 4.

To avoid the deadlock in intra-layer (i.e., within a xy-plane), the injected message follows Rules 3 and 4. For example, the injected message is restricted North/South to West turn at non-prime column,  $x_c = 1, 4, 5, 6$  according to Rule 3, and Rule 4 forbids East to North/South turn at prime column,  $x_c = 2, 3, 5$  as shown in Fig. 4.

The Algorithm 1, PTM-3D divides the routing problem based on plane-wise prohibited turn model along z-direction perspective. Let us assume that current node  $(x_c, y_c, z_c)$ , source node  $(x_s, y_s, z_s)$ , destination node  $(x_d, y_d, z_d)$  are the coordinate points of 3D-mesh having three coordinate axes (x-direction, y-direction, and z-direction).

The Algorithm 1 divides basically into three cases ( $z_e = 0$ ,  $z_e > 0$ ,  $z_e < 0$ ) based on offset(z) along z-axis ( $z_e = z_d - z_c$ ). For the first case ( $z_e = 0$ ), it calls algorithm PTM-2D as shown in Fig. 4. In the second case ( $z_e > 0$ ),

---

#### ALGORITHM 1: PTM-3D

---

**Input** : current node  $(x_c, y_c, z_c)$ , source node  $(x_s, y_s, z_s)$ , destination node  $(x_d, y_d, z_d)$

**Output:** *Avail\_Direction\_Set*

- 1: *Avail\_Direction\_Set* =  $\phi$ ;
  - 2:  $x_e = x_d - x_c$ ;
  - 3:  $y_e = y_d - y_c$ ;
  - 4:  $z_e = z_d - z_c$ ;
  - 5: **if**  $z_e = 0$  **then**
  - 6:     call PTM-2D;
  - 7: **else if**  $z_e > 0$  **then**
  - 8:     **if**  $x_e = 0$  and  $y_e = 0$  **then**
  - 9:         Add **Up** ( $z_c + 1$ ) to *Avail\_Direction\_Set*;
  - 10:     **else**
  - 11:         **if**  $isPrime(z_c) = false$  or  $z_c = z_s$  **then**
  - 12:             call PTM-2D;
  - 13:         **if**  $z_c = 1$  and  $z_d = 3$  **then**
  - 14:             **return** (*Avail\_Direction\_Set*);
  - 15:         **if**  $z_e > 1$  or  $isPrime(z_d) = false$  **then**
  - 16:             Add **Up** ( $z_c + 1$ ) to *Avail\_Direction\_Set*;
  - 17:     **else**
  - 18:         Add **Down** ( $z_c - 1$ ) to *Avail\_Direction\_Set*;
  - 19:         **if**  $isPrime(z_c) = true$  and  $(x_e \neq 0$  or  $y_e \neq 0)$  **then**
  - 20:             call PTM-2D;
  - 21:     **return** (*Avail\_Direction\_Set*);
-

if both offsets along  $x_e$  and  $y_e$  becomes zero then the packet adds route Up ( $z_c + 1$ ) direction in available direction set; if the current plane is non-prime ( $isPrime(z_c)=false$ ) or the current plane is source plane, then it calls PTM-2D, and if a situation becomes  $z_c = 1$  and  $z_d = 3$  then the packet routes to a low congested immediate neighbor from the avail directions set; otherwise the packet also adds Up direction in available direction set if the current offset  $z_e > 1$  or  $isPrime(z_d) = false$ . For the last case ( $z_e < 0$ ), the available direction set adds Down ( $z_c - 1$ ) direction and it also calls PTM-2D, if the current plane is prime whereas ( $x_e \neq 0$  or  $y_e \neq 0$ ). Therefore, The current message firstly analyzes all possible available paths/routes using PTM-3D routing algorithm and forwards to one of direction in *Avail\_Direction\_Set* where network traffic is low and update the current position.

Based on the above cases, Algorithm 2 called by Algorithm 1. Initially Avail\_Direction\_Set is empty( $\phi$ ), and both  $x_e$  and  $y_e$  are the offset between current position and destination node along x-direction and y-direction respectively. If both offsets are zero, that means, the current packet reached to the destination node, hence the packet goes to the local core and return from it. But, for  $x_e = 0$ , the available direction set contains either North ( $y_c + 1$ ) or South ( $y_c - 1$ ) direction, depend on its case,  $y_e > 0$  or not. For  $x_e > 0$ , if offset  $y_e = 0$  then it adds East( $x_c + 1$ ) in available direction set otherwise, for the case  $isPrime(x_c) = false$  or  $x_c = x_s$ , the available direction set contains either North ( $y_c + 1$ ) or South ( $y_c - 1$ ) direction depend on its case,  $y_e > 0$  or not, and if a situation  $x_c = 1$  and  $x_d = 3$  occurred then the packet moves to one of the direction from available direction set, that means  $y_c \rightarrow y_d$ . The East ( $x_c + 1$ ) direction is available, either the current packet is far at least two hops distance ( $x_e > 1$ ) or the destination node is not at prime position along x-direction ( $isPrime(x_d) = false$ ). For the third case,  $x_e < 0$ , the available direction set contains West as well as North/South direction depends on,  $y_e > 0$  becomes true or false. At the end of the algorithm, it returns all available directions set, and among all, any one can took as current path.

The repetitive (prohibited turn) distance for RTM-3D is three(3) but, the repetitive distance for PTM-3D is not repetitive (not constant) under any size of network. Since, higher the maximum column repetitive distance (MRD) has lower the routing pressure observed in [21]. So, we can say that the repetitive distance for PTM-3D is the size of network (e.g.,  $\Delta z$ ) and it has low routing pressure.

**ALGORITHM 2: PTM-2D**


---

**Input** : current node( $x_c, y_c$ ), source node( $x_s, y_s$ ), destination node( $x_d, y_d$ )  
**Output**: *Avail\_Direction\_Set*

- 1: *Avail\_Direction\_Set* =  $\phi$ ;
- 2:  $x_e = x_d - x_c$ ;
- 3:  $y_e = y_d - y_c$ ;
- 4: **if**  $x_e = 0$  **and**  $y_e = 0$  **then**
- 5:   Packet goes to local core and **exit**;
- 6: **if**  $x_e = 0$  **then**
- 7:   **if**  $y_e > 0$  **then**
- 8:     Add **North** ( $y_c + 1$ ) to *Avail\_Direction\_Set*;
- 9:   **else**
- 10:    Add **South** ( $y_c - 1$ ) to *Avail\_Direction\_Set*;
- 11: **else**
- 12:   **if**  $x_e > 0$  **then**
- 13:     **if**  $y_e = 0$  **then**
- 14:       Add **East** ( $x_c + 1$ ) to *Avail\_Direction\_Set*;
- 15:     **else**
- 16:       **if**  $isPrime(x_c) = false$  or  $x_c = x_s$  **then**
- 17:         **if**  $y_e > 0$  **then**
- 18:         Add **North** ( $y_c + 1$ ) to *Avail\_Direction\_Set*;
- 19:         **else**
- 20:         Add **South** ( $y_c - 1$ ) to *Avail\_Direction\_Set*;
- 21:         **if**  $x_c = 1$  **and**  $x_d = 3$  **then**
- 22:         **return** (*Avail\_Direction\_Set*);
- 23:         **if**  $x_e > 1$  or  $isPrime(x_d) = false$  **then**
- 24:         Add **East** ( $x_c + 1$ ) to *Avail\_Direction\_Set*;
- 25:         **else**
- 26:         Add **West** to *Avail\_Direction\_Set*;
- 27:         **if**  $isPrime(x_c) = true$  **then**
- 28:         **if**  $y_e > 0$  **then**
- 29:         Add **North** ( $y_c + 1$ ) to *Avail\_Direction\_Set*;
- 30:         **else**
- 31:         Add **South** ( $y_c - 1$ ) to *Avail\_Direction\_Set*;
- 32: **return** (*Avail\_Direction\_Set*);

---

**3.2 Illustrative Example**

A 3D-mesh ( $7 \times 7 \times 7$ ) topology where each joint (black dot) of mesh topology depicted as router (6 ports) and it is dedicated to a local core, is shown in Fig. 5. Both Rules 1 and 2 are applicable if the packets are communicating from layer to layer (inter-plane) whereas both Rules 3 and 4 are applicable if packet moves in the same plane (intra-plane) where x-coordinate decides prohibited turn model for xy-plane.



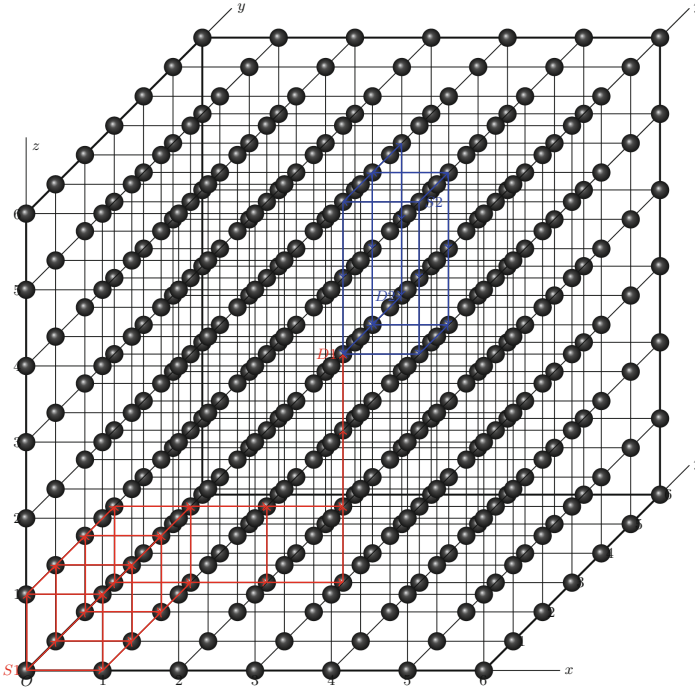


Fig. 5.  $7 \times 7 \times 7$  3D-mesh architecture.

In Fig. 5, two source-destination pairs,  $\langle S1, D1 \rangle$  and  $\langle S2, D2 \rangle$ , are marked with red and blue, respectively, to illustrate the PTM-3D routing algorithm in 3D mesh ( $7 \times 7 \times 7$ ) topology. The first pair (red),  $\langle S1, D1 \rangle$  routes a message from  $S1(0,0,0)$  to  $D1(3,3,3)$ , whereas the second pair (blue),  $\langle S2, D2 \rangle$ , routes a message from  $S2(4,3,5)$  to  $D2(3,5,3)$  and the available direction set for each message/packet is taken as empty, i.e.,  $Avail\_Direction\_Set = \phi$ .

For the first pair (red)  $\langle S1(0,0,0), D1(3,3,3) \rangle$ , initially, the offsets are  $x_e = y_e = z_e = 3$  along x-axis, y-axis, and z-axis respectively. First, the source node  $S1(0,0,0)$  can add a route  $(1,0,0)$  or  $(0,1,0)$  to available direction set by calling PTM-2D since  $(z_c = z_s)$ , or it can add a route to the immediate upper plane arriving at  $(0,0,1)$ , so the  $Avail\_Direction\_Set$  is  $\{(1,0,0), (0,1,0), (0,0,1)\}$ . The current packet is arriving at any of the available directions, thus let's assume that  $(0,0,1)$ . The message at  $(0,0,1)$  cannot be routed to  $(0,0,2)$  since Rule 2 is violated, and hence the message at  $(0,0,1)$  routes either at  $(1,0,1)$  or  $(0,1,1)$  in its plane ( $z_c = 1$ ), and let's assume that the packet is receiving at  $(1,0,1)$ . The message at  $(1,0,1)$  cannot be routed to  $(1,0,2)$  because a situation,  $z_c = 1$  and  $z_d = 3$  occurred, or to  $(2,0,1)$  because again a situation,  $x_c = 1$  and  $x_d = 3$  occurred. Therefore, the message at  $(1,0,1)$  reached to  $(1,1,1)$ , and similarly, from  $(1,1,1)$  to  $(1,2,1)$ , from  $(1,2,1)$  to  $(1,3,1)$ . For the message at  $y_e = 0$  and  $z_c = 1$ , only East direction is available. When the message where  $x_e = 0$  and  $y_e = 0$ , it routes to Up

direction only and reached to the destination  $D1(3,3,3)$ . Therefore, for the first pair(red)  $\langle S1, D1 \rangle$ , the message at source  $S1(0,0,0)$  can travel via 28 alternative paths to reach the destination  $D1(3,3,3)$  using the PTM-3D routing algorithm. One of the routing path is as:  $\langle S1(0,0,0) \rightarrow (0,0,1) \rightarrow (1,0,1) \rightarrow (1,1,1) \rightarrow (1,2,1) \rightarrow (1,3,1) \rightarrow (2,3,1) \rightarrow (3,3,1) \rightarrow (3,3,2) \rightarrow D1(3,3,3) \rangle$ . Likewise, another routing path is as:  $\langle S1(0,0,0) \rightarrow (0,1,0) \rightarrow (0,2,0) \rightarrow (0,3,0) \rightarrow (1,3,0) \rightarrow (2,3,0) \rightarrow (3,3,0) \rightarrow (3,3,1) \rightarrow (3,3,2) \rightarrow D1(3,3,3) \rangle$ .

For the second pair (blue)  $\langle S2(4,3,5), D2(3,5,3) \rangle$ , the packet at source (S2) can find one of the routing path as:  $\langle S2(4,3,5) \rightarrow (4,3,4) \rightarrow (4,3,3) \rightarrow (4,4,3) \rightarrow (3,4,3) \rightarrow D2(3,5,3) \rangle$ ; likewise, another path is as:  $\langle S2(4,3,5) \rightarrow (4,3,4) \rightarrow (4,3,3) \rightarrow (3,3,3) \rightarrow (3,4,3) \rightarrow D2(3,5,3) \rangle$ . Therefore, for the second pair(red)  $\langle S2, D2 \rangle$ , has 08 alternative paths to reach destination  $D2(3,5,3)$ .

### 3.3 Deadlock-Freedom Proof

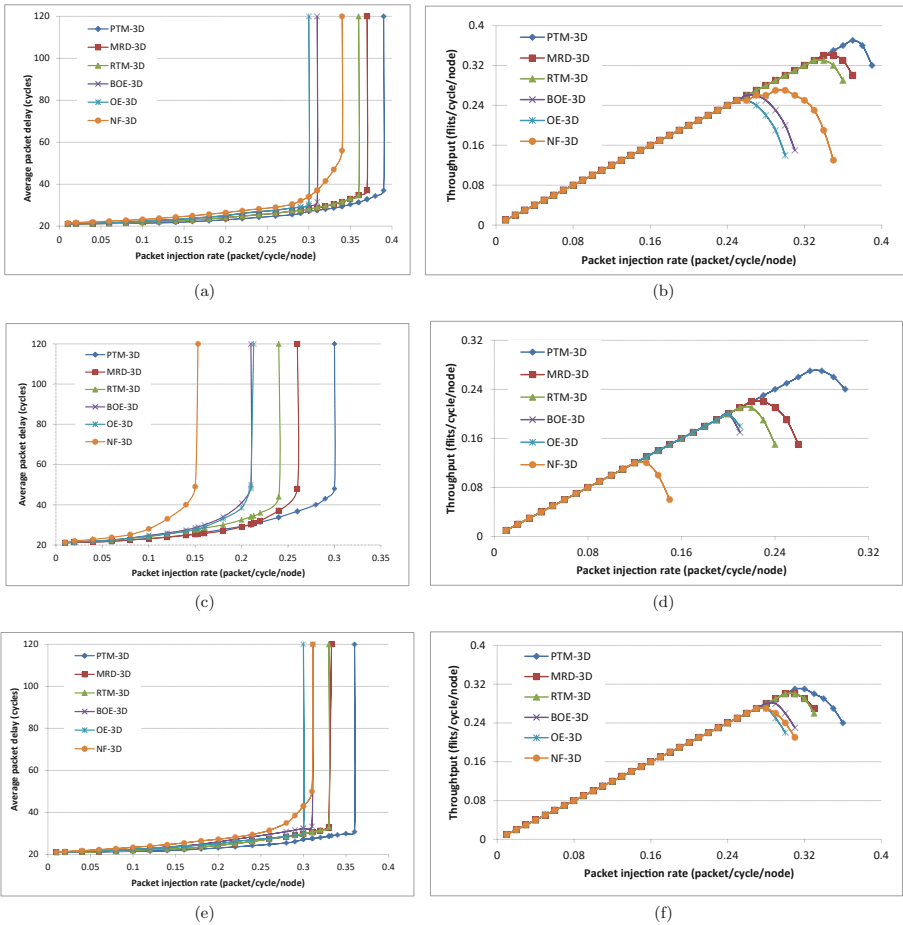
To ensure deadlock-freeness in intra-layer (PTM-2D), the algorithm must obey 12 possible ways of prohibited turns in each  $2 \times 2$  submesh network as shown in Fig. 2, e.g., Rules 3 and 4. In PTM-3D, Rules 1 and 2 are implemented to make sure that no cycle exists between any two layers, which is one of necessary condition for inter-layer deadlock-freeness.

## 4 Experimental Results

In this section, we compare the performance metrics (delay, throughput) of the proposed model, *prime number based turn model routing for 3D-mesh topology (PTM-3D) to maximum column repetitive distance based 3D routing (MRD-3D routing)* [4], *repetitive turn model for 3D routing (RTM-3D)* [4], *balanced odd-even turn model for 3D routing (BOE-3D)* [7], *odd-even turn model for 3D routing(OE-3D)* [6,7] and *negative-first turn model for 3D routing (NF-3D)* [12]. These all are simulated on network simulator, Noxim [5] using virtual cut-through technique for  $5 \times 5 \times 5$  mesh as well as  $5 \times 5 \times 7$  mesh up to 12000 cycles. PTM-3D model is illustrated in detail on  $7 \times 7 \times 7$  3D-mesh to show two complex cases separately so that all four rules can be practiced but it efficiently works also in  $5 \times 5 \times 5$  3D-mesh. Here, we have chosen  $5 \times 5 \times 5$  3D-mesh to show and compare performance parameters with different-different recent routing models in experimental results because most of the routing models (for example, RTM-3D, MRD-3D etc.).

Now, we calculate the performance parameter (delay, throughput) in respect of packet injection rate for each message. We also considered three different typical traffic patterns namely: uniform, transpose and hotspot traffic. In uniform traffic, the source node and the destination node are fully randomly distributed in the network topology. In transpose traffic, both source node  $(i, j, k)$  and destination node  $(j, k, i)$  should be the mirror image of each other along the principal diagonal axis. Now, in hotspot traffic, a particular node chosen as hotspot that receive 4 percent extra packets in addition to the uniform traffic.

PTM-3D routing model works under uniform, transpose, and hotspot cases and compared to RTM-3D Fig. 6 demonstrates the performance metrics(average packet latency and throughput) versus packet injection rate in uniform, transpose, and hotspot cases for  $5 \times 5 \times 5$  3D-meshes. In Fig. 6(a), PTM-3D got saturated (sat) when PIR is 0.36 in uniform traffic. Higher the saturation point offers less congestion over lower saturation point. Figure 6(b), the throughput of PTM-3D also performs better than the previous routing schemes under uniform traffic case. In most of the cases, PTM-3D performed better result than the previous turn models under the transpose traffic (Fig. 6(c) and (d)) as well



**Fig. 6.** Performance metrics in  $5 \times 5 \times 5$  meshes. (a) Average packet delay versus Packet injection rate (PIR) under uniform traffic, (b) Throughput Vs PIR under uniform traffic, (c) Average packet delay Vs PIR under transpose traffic, (d) Throughput Vs PIR under transpose traffic, (e) Average packet delay Vs PIR in hotspot traffic, (f) Throughput Vs PIR in hotspot traffic.

as hotspot traffic (Fig. 6(e) and (f)) having two hotspot positions  $(\mathbf{3}, \mathbf{1}, \mathbf{1})$  and  $(\mathbf{2}, \mathbf{3}, \mathbf{4})$ . PTM-3D also performed better results in case of cuboid 3D-meshes in uniform, transpose, and hotspot cases.

Tables 1 and 2 show more concrete view of average packet delay against PIR and throughput against PIR for the above six mentioned routing strategies under uniform, transpose, hotspot traffics. In most of the cases, they concludes that PTM-3D has reduced average packet delay and increased throughput than the other routing strategies.

**Table 1.** Average packet delay (cycles) against Packet injection rate (packet/cycle/node) in uniform, transpose and hotspot traffics in  $5 \times 5 \times 5$  meshes

PIR	Average packet delay (cycles)																	
	Uniform traffic						Transpose traffic						Hotspot traffic					
	PTM-3D	MRD-3D	RTM-3D	BOE-3D	OE-3D	NF-3D	PTM-3D	MRD-3D	RTM-3D	BOE-3D	OE-3D	NF-3D	PTM-3D	MRD-3D	RTM-3D	BOE-3D	OE-3D	NF-3D
0.04	21.17	21.2	21.2	21.4	21.42	22.05	21.5	21.5	21.9	22	22	22.8	21.17	21.2	21.2	21.4	21.42	21.8
0.08	21.36	21.5	21.5	22	22.04	22.8	22.55	22.5	23.3	23.5	23.5	25.2	21.36	21.5	21.5	22	22.04	22.8
0.12	21.59	22.35	22.35	22.9	22.96	23.75	24	23.85	25.5	26	25.5	33	21.59	22.35	22.35	22.9	22.96	24
0.16	22.2	23	23	23.8	23.88	25	26.25	25.75	28.5	30	29	sat	22.2	22.9	22.9	24.1	23.88	25.5
0.2	23.1	24.1	24.1	25	25.1	26.5	29.25	29	32.5	41	38.5	--	23.1	24.3	24.3	26	25.1	27.25
0.24	24.2	25.5	25.5	26.85	27	28.3	33.75	37	44	sat	sat	--	24.2	26.2	26.2	28.4	27	29.5
0.28	25.5	27.2	27.3	28.7	28.7	30.5	40	sat	sat	--	--	--	25.5	28.4	28.5	30.9	28.7	35
0.32	28.1	29.5	29.65	sat	sat	41.5	sat	--	--	--	--	--	28.1	31.35	31.6	sat	sat	sat
0.36	31.3	34.8	35	--	--	sat	--	--	--	--	--	--	30.8	sat	sat	--	--	--
0.4	sat	sat	sat	--	--	--	--	--	--	--	--	--	sat	--	--	--	--	--

**Table 2.** Throughput (flits/cycle/node) against Packet injection rate (packet/cycle/node) in uniform, transpose and hotspot traffics in  $5 \times 5 \times 5$  meshes

PIR	Throughput (flits/cycle/node)																	
	Uniform traffic						Transpose traffic						Hotspot traffic					
	PTM-3D	MRD-3D	RTM-3D	BOE-3D	OE-3D	NF-3D	PTM-3D	MRD-3D	RTM-3D	BOE-3D	OE-3D	NF-3D	PTM-3D	MRD-3D	RTM-3D	BOE-3D	OE-3D	NF-3D
0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04	0.04
0.08	0.08	0.08	0.08	0.08	0.08	0.08	0.08	0.08	0.08	0.08	0.08	0.08	0.08	0.08	0.08	0.08	0.08	0.08
0.12	0.12	0.12	0.12	0.12	0.12	0.12	0.12	0.12	0.12	0.12	0.12	0.12	0.12	0.12	0.12	0.12	0.12	0.12
0.16	0.16	0.16	0.16	0.16	0.16	0.16	0.16	0.16	0.16	0.16	0.16	sat	0.16	0.16	0.16	0.16	0.16	0.16
0.20	0.20	0.20	0.20	0.20	0.20	0.20	0.20	0.20	0.20	0.20	0.20	--	0.20	0.20	0.20	0.20	0.20	0.20
0.24	0.24	0.24	0.24	0.24	0.24	0.24	0.24	0.21	0.15	sat	sat	--	0.24	0.24	0.24	0.24	0.24	0.24
0.28	0.28	0.28	0.28	0.25	0.22	0.26	0.27	sat	sat	--	--	--	0.28	0.28	0.28	0.28	0.27	0.27
0.32	0.32	0.32	sat	sat	sat	0.25	sat	--	--	--	--	--	0.31	0.29	0.29	sat	sat	sat
0.36	0.36	sat	sat	--	--	sat	--	--	--	--	--	--	0.24	sat	sat	--	--	--
0.4	sat	--	--	--	--	--	--	--	--	--	--	--	sat	--	--	--	--	--

Table 3 shows the PIR saturation point and % improvement of performance parameters (average packet delay, throughput) PTM-3D routing scheme to the rest mentioned routing schemes under uniform, transpose, and hotspot traffics. In this table, PIR saturation point = 0.36 and % improvement = 10.57 for the RTM-3D routing scheme under uniform traffic that means, when the PIR reaches to 0.36 under uniform traffic then RTM-3D routing scheme got fully saturated, did not receive any data more and at that point PTM-3D routing scheme improve the performance over RTM-3D routing scheme by 10.57%.

Table 3 concludes that PTM-3D routing schemes performs better than the previous routing schemes, but when the 3D-mesh size increase, its performance get more enhance.

**Table 3.** Saturation point and % improvement (delay, throughput) of PTM-3D to five different routing strategies under uniform, transpose, and hotspot traffic

Routing strategies	PTM-3D						Throughput (flits/cycle/node)					
	Average packet delay (cycle)											
	Uniform		Transpose		Hotspot		Uniform		Transpose		Hotspot	
PIR saturation point	% improvement	PIR saturation point	% improvement	PIR saturation point	% improvement	PIR saturation point	% improvement	PIR saturation point	% improvement	PIR saturation point	% improvement	
MRD-3D	0.37	11.35	0.26	23.4	0.33	13.1	0.36	09.0	0.26	42.3	0.33	11.1
RTM-3D	0.36	10.57	0.24	23.3	0.33	13.6	0.36	24.13	0.24	37.5	0.33	15.4
BOE-3D	0.31	12.7	0.21	40	0.31	17.3	0.31	66.7	0.21	23.5	0.31	34.8
OE-3D	0.31	11.1	0.21	38.6	0.3	09.3	0.3	87.7	0.21	14.3	0.3	26.6
NF-3D	0.34	47.5	0.15	47.8	0.31	46.9	0.35	2.5x	0.15	2.5x	0.31	47.6

## 5 Conclusions

In this paper, we briefly discussed various prohibited turn routing models to ensure adaptive deadlock-freeness for 2D/3D-mesh networks. The prohibited turn models are working even without virtual channels. A packet can route prime based prohibited turn model using proposed algorithm, PTM-3D (Prime perspective Turn Model for 3D-Mesh) which ensure deadlock-freedom adaptiveness, in 3D-mesh network. To measure the performance, we have shown the comparison based simulation results of the average packet delay (in cycles) as well as throughput (flits/cycle/node) along packet injection rate (packet/cycle/node) among six different routing strategies, PTM-3D, MRD-3D, RTM-3D, BOE-3D, OE-3D, NF-3D under uniform, transpose and hotspot cases cases of  $5 \times 5 \times 5$  3D-meshes. After all, the results conclude that, PTM-3D is performed relatively better than other routing models in 3D-cube mesh or 3D-cuboid mesh in most of the cases (uniform, transpose, or hotspot).

PTM-3D has some limitations as: (a) PTM-3D works efficiently if mesh size is not beyond the  $36 \times 36 \times 36$  3D-mesh, (b) for smaller 3D-meshes (between  $2 \times 2 \times 2$  to  $4 \times 4 \times 4$ ), it gives similar results to RTM-3D, (c) PTM-3D do not have awareness of faults (switch/router) but in near future, we are considering fault-aware cases.

## References

1. Abbas, A., et al.: A survey on energy-efficient methodologies and architectures of network-on-chip. *Comput. Electr. Eng.* **40**(8), 333–347 (2014)
2. Bjerregaard, T., Mahadevan, S.: A survey of research and practices of network-on-chip. *ACM Comput. Surv. (CSUR)* **38**(1), 1 (2006)
3. Cai, Y., Luo, W., Xiang, D.: The column-partition and row-partition turn model. In: 42nd Annual Computer Software and Applications Conference (COMPSAC), vol. 1, pp. 687–694. IEEE (2018)
4. Cai, Y., Xiang, D., Ji, X.: Deadlock-free adaptive routing based on the repetitive turn model for 3d network-on-chip. In: International Conference on Parallel & Distributed Processing with Applications, Ubiquitous Computing & Communications, Big Data & Cloud Computing, Social Computing & Networking, Sustainable Computing & Communications, pp. 722–728. IEEE (2018)
5. Catania, V., Mineo, A., Monteleone, S., Palesi, M., Patti, D.: Noxim: an open, extensible and cycle-accurate network on chip simulator. In: IEEE 26th International Conference on Application-specific Systems, Architectures and Processors (ASAP), pp. 162–163. IEEE (2015)

6. Chiu, G.M.: The odd-even turn model for adaptive routing. *IEEE Trans. Parallel Distrib. Syst.* **11**(7), 729–738 (2000)
7. Dahir, N., Al-Dujaily, R., Yakovlev, A., Missailidis, P., Mak, T.: Deadlock-free and plane-balanced adaptive routing for 3d networks-on-chip. In: *Proceedings of the Fifth International Workshop on Network on Chip Architectures*, pp. 31–36. ACM (2012)
8. Dally, W.J., Towles, B.P.: *Principles and practices of interconnection networks*. Elsevier (2004)
9. Dash, R., Majumdar, A., Pangracious, V., Turuk, A.K., Risco-Martin, J.L.: ATAR: an adaptive thermal-aware routing algorithm for 3-D network-on-chip systems. *IEEE Trans. Components Packaging Manuf. Tech.* **8**(12), 2122–2129 (2018)
10. Delgado, R., Park, J., Choi, B.W.: MPSoC: the low-cost approach to real-time hardware simulations for power and energy systems. *IFAC-PapersOnLine* **52**(4), 57–62 (2019)
11. Flich, J., Duato, J.: Logic-based distributed routing for NOCs. *IEEE Comput. Archit. Lett.* **7**(1), 13–16 (2008)
12. Glass, C.J., Ni, L.M.: The turn model for adaptive routing. *ACM SIG. Comput. Archit. News* **20**(2), 278–287 (1992)
13. Jouybari, H.N., Mohammadi, K.: A low overhead, fault tolerant and congestion aware routing algorithm for 3D mesh-based network-on-chips. *Microprocess. Microsyst.* **38**(8), 991–999 (2014)
14. Kumar, S., et al.: A network on chip architecture and design methodology. In: *Proceedings IEEE Computer Society Annual Symposium on VLSI*, pp. 105–112. IEEE (2002)
15. Manzoor, M., Mir, R.N., et al.: PAAD (Partially adaptive and Deterministic routing): a deadlock free congestion aware hybrid routing for 2D mesh network-on-chips. *Microprocessors and Microsystems*, p. 104551 (2022)
16. Manzoor, M., Mir, R.N., et al.: Prime turn model and first last turn model: an adaptive deadlock free routing for network-on-chips. *Microprocess. Microsyst.* **89**, 104454 (2022)
17. Mejia, A., Flich, J., Duato, J.: On the potentials of segment-based routing for NOCs. In: *2008 37th International Conference on Parallel Processing*, pp. 594–603. IEEE (2008)
18. Sharma, P.K., Biswas, S., Mitra, P.: Energy efficient heuristic application mapping for 2-D mesh-based network-on-chip. *Microprocess. Microsyst.* **64**, 88–100 (2019)
19. Tang, M., Lin, X., Palesi, M.: Routing pressure: a channel-related and traffic-aware metric of routing algorithm. *IEEE Trans. Parallel Distrib. Syst.* **26**(3), 891–901 (2013)
20. Tang, M., Lin, X., Palesi, M.: An offline method for designing adaptive routing based on pressure model. *IEEE Trans. Comput. Aided Des. Integr. Circuits Syst.* **34**(2), 307–320 (2014)
21. Tang, M., Lin, X., Palesi, M.: The repetitive turn model for adaptive routing. *IEEE Trans. on Computers* **66**(1), 138–146 (2016)
22. Tosun, S., Ozturk, O., Ozkan, E., Ozen, M.: Application mapping algorithms for mesh-based network-on-chip architectures. *J. Supercomput.* **71**(3), 995–1017 (2015)
23. Wu, J.: A deterministic fault-tolerant and deadlock-free routing protocol in 2-D meshes based on odd-even turn model. In: *Proceedings of the 16th International Conference on Supercomputing*, pp. 67–76. ACM (2002)