PAPER
# A New Multiple-Round Dimension-Order Routing for Networks-on-Chip

Binzhang FU[†,††], *Student Member*, Yinhe HAN[†,††a)], Huawei LI[†,††], *Members*, *and* Xiaowei LI[†,††], *Nonmember*

**SUMMARY**    The *Network-on-Chip* (NoC) is limited by the reliability constraint, which impels us to exploit the fault-tolerant routing. Generally, there are two main design objectives: tolerating more faults and achieving high network performance. To this end, we propose a new multiple-round dimension-order routing (NMR-DOR). Unlike existing solutions, besides the intermediate nodes inter virtual channels (VCs), some turn-legally intermediate nodes inside each VC are also utilized. Hence, more faults are tolerated by those new introduced intermediate nodes without adding extra VCs. Furthermore, unlike the previous solutions where some VCs are prioritized, the NMR-DOR provides a more flexible manner to evenly distribute packets among different VCs. With extensive simulations, we prove that the NMR-DOR maximally saves more than 90% unreachable node pairs blocked by faults in previous solutions, and significantly reduces the packet latency compared with existing solutions.

*key words:* network-on-chip (NoC), fault-tolerant routing, multiple round dimension-order routing, turn model

## 1. Introduction

As VLSI technology continues to advance, abundant transistors are utilized to build many-core chips, where the Network-on-Chip (NoC) instead of the bus services the on-chip communication [1]. Unfortunately, as Furber estimated [2], although each individual router/link is robust, NoC still faces a high failure possibility due to the rapidly increased network size.

In general, faults can be categorized into transient and permanent faults. For transient faults, they are usually addressed by the Error-Detection-Code and the retransmission mechanism. For permanent faults, the solutions are more complicated.

Some proposals focus on improving the reliability of individual router/link by sharing or redundancy [25]. The fault-tolerant routing, meanwhile, is utilized to provide communications for remaining fault-free nodes.

Fault-tolerant routing has a long history. Initially, it was designed for the multicomputer or multiprocessor systems. Recently, with the pervasive emergence of NoC, many of them are inherited and modified to meet the special requirements of NoC.

For example, although the resources within a chip are

abundant, the part allocated to each individual router is still limited. Hence, the VCs that were viewed as cheap and abundant, now become expensive and limited [23]. Furthermore, in NoC, failed components cannot be replaced by fault-free ones anymore. Hence, in the presence of faults, the network performance should be kept as high as possible.

Multiple-round routing algorithm has been proved to be effective to provide fault-tolerant communication due to several advantages [17], [20]. For example, it facilitates the design of fast routers by keeping the routing function simple, and also significantly reduces the complexity of design verification. Based on the observation of the effectiveness of multiple-round routing algorithms, this paper proposes a new multiple-round dimension-order routing (NMR-DOR). Compared with existing multiple-round routing algorithms [17], [20], the NMR-DOR strongly reduces the number of unreachable node pairs and further improves the system performance in the presence of faults.

To achieve the first goal, we propose to use turn-legally intermediate nodes, where the turn made by two consecutive routing phases is allowed by the turn model [5]. Thus, there is no need for adding a separate VC to keep the network deadlock free. In other words, given the number of VCs, the NMR-DOR could utilize more intermediate nodes. Hence, more faults can be tolerated, which also means much fewer fault-free node pairs will be unreachable.

In [17] and [20], a packet accesses to an VC only if it finishes the routing phases in all lower indexed VCs. Since not every packet needs the higher routing phases, the lower indexed VCs will carry more traffic and be saturated first. To address this problem, NMR-DOR allows packets to arbitrarily select VCs. This makes the traffic more balanced.

The main contribution of this paper is that we propose a new multiple-round dimension-order routing that has the following advantages:

1. it exploits the turn-legally intermediate nodes to reduce the number of VCs and unreachable node pairs,
2. it flexibly treats the packets utilizing turn-legally intermediate nodes to balance traffic.

The rest of the paper is organized as follows: Sect. 2 discusses the background and related work. Section 3 defines the turn-legally intermediate nodes. Section 4 describes the proposed routing algorithm. Section 5 proposes three complementary techniques. Section 6 evaluates the proposed NMR-DOR. Section 7 gives a discussion about the NMR-DOR. Section 8 concludes this paper.

## 2. Background and Related Work

One of the main challenges of fault-tolerant routing is to keep the network deadlock-free in the presence of faults. In Sect. 2.1, we summarize the principles to design deadlock-free routing algorithms. In Sect. 2.2, we review the existing fault-tolerant routing algorithms, especially the multi-round routing algorithms, and distinguish the proposed NMR-DOR from them.

### 2.1 To Keep a Network Deadlock Free

In general, with wormhole or virtual channel flow control, a deadlock happens when several packets cannot advance due to a dependence cycle among themselves. Hence, there are two major methods to keep a network deadlock-free: preventing the formation of the dependence cycle (deadlock avoidance) and breaking the cycle when it is formed (deadlock recovery).

To date, most NoCs use deadlock avoidance techniques, and there are 3 principles in common use: the Dally and seitz's theory [3], the Duato's theory [4], and the turn models [5], [6]. We review each of these in turn.

#### 2.1.1 Dally and Seitz's Theory

The main idea of Dally and Seitz's theory [3] is to establish a channel dependence graph, and make sure that no cycle is formed.

Specifically, an interconnection network, $I$, is defined as a strongly connected directed graph, $I = G(N, C)$, where the vertices, $N$, represent the set of nodes, and the edges, $C$, are the set of channels. The routing function, $R : C \times N \to C$, is responsible for mapping the current channel, $c_c$, and the destination node, $n_d$, to the next channel $c_n$, i.e., $R(c_c, n_d) = c_n$.

Given the interconnection network and routing function, the channel dependence graph, $D$, is defined as a directed graph, $D = G(C, E)$, where vertices are the channels and edges are the pairs of channels connected by $R$: $E = \{(c_i, c_j) | R(c_i, n) = c_j \text{ for some } n \in N\}$.

Therefore, a routing function, $R$, is deadlock free, iff there is no cycle in its channel dependence graph, $D$.

#### 2.1.2 Duato's Theory

With Duato's theory [4], an adaptive routing algorithm is deadlock free if we can find a routing subfunction with an acyclic extended channel dependence graph.

Specifically, a routing function, $R$, is defined as $R : N \times N \to \mathcal{P}(C)$, and the routing subfunction, $R_1$, is defined as $R_1 : N \times N \to \mathcal{P}(C_1) / R_1(x, y) = R(x, y) \cap C_1$, $\forall x, y \in N$, where $C_1 \subseteq C$.

Given an interconnection network and a routing subfunction, an extended channel dependence graph $D_E$ is a directed graph, $D_E = G(C_1, E_E)$. The vertices of $D_E$ are the channels that define the routing subfunction. The edges are the pairs of channels $(c_i, c_j)$ that have a direct or indirect dependence. A direct dependence indicates that $c_i$ and $c_j$ are connected by $R_1$. An indirect dependence means that they are connected by several channels using the original routing function $R$.

Therefore, an adaptive routing function, $R$, is deadlock free, if there is a subset of channels, $C_1 \subseteq C$, that defines a routing subfunction, $R_1$, which has an acyclic extended channel dependence graph.

#### 2.1.3 The Turn Models

Turn models significantly reduce the complexity of breaking all cycles by prohibiting one turn in each kind of abstract cycles [5]. For 2D meshes, there are two kinds of abstract cycles and four turns in each. Thus, there are totally 16 different ways to prohibit two turns, but only 12 are legal, as shown in Fig. 1. We list all 12 legal ways here, instead of the 3 unique ones in [5], because each of them can tolerate different kinds of faults as will be discussed in Sect. 3.

However, Chiu [6] found that the uneven adaptiveness of original turn model may hurt the network performance, especially under some non-uniform traffic patterns. Hence, he proposed the odd-even turn model. As shown in Fig. 2, the $ES(east \to south)$ and $EN$ $(east \to north)$ turns are forbidden in odd columns, and the $NW$ $(north \to west)$ and $SW$ $(south \to west)$ turns are forbidden in even columns.

### 2.2 The Fault-Tolerant Routing Algorithms

In the following of this subsection, we will review the fault-tolerant routing algorithms based on which principle they adopt to keep the network deadlock free.

With Dally and Seitz's theory, one possible implementation is to analyze the channel dependence graph and break all cycles. Exhausting cycles, however, is a hard job especially for large-scale networks. Segment-based routing [21]
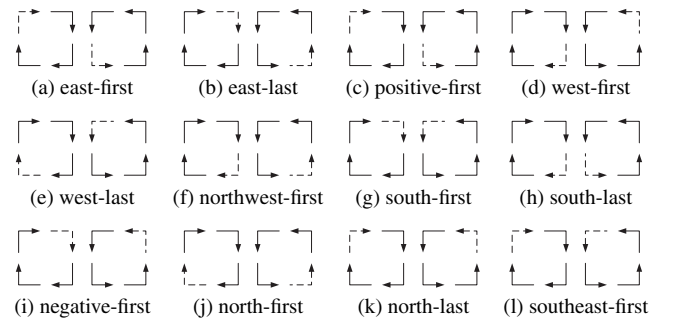


**Fig. 1** Partially adaptive routing algorithms based on turn model. Forbidden turns are shown as dashed lines.
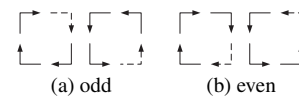


**Fig. 2** Odd-even turn model.

reduces this job to finding different segments. Rodrigo et.al. [29] utilized the segment-based routing and extended the LBDR [28] to support non-minimal paths.

Another implementation is to classify packets into several classes based on packet directions [8], [9], [11], [12], [15], [19] or routing phases [17]. To avoid deadlock, each class of packets is allocated a separate VC. The routing algorithms guarantee that the channel dependence graph within each class is acyclic and the transition among different classes is in a strictly increasing or decreasing order.

For example, Linder and Harden [8] proposed to utilize maximally $2^{n-1} + 1$ VCs to tolerate faults for n-dimensional meshes. Chien and Kim [9] proposed the planar-adaptive routing, and reduced the number of VCs to 3. Boppana and Chalasani [11] proposed to include faults into convex faulty-blocks, i.e., the rectangular faulty-blocks with respect to 2D meshes. Maximally, four VCs are required. Later, they proposed an algorithm to tolerate solid faults (the orthogonal convex faulty-blocks) to save the fault-free nodes included into the rectangular faulty-blocks [12]. To further reduce the number of VCs, Chen and Chiu [15] proposed an algorithm with 3 VCs per physical channel. Xiang et.al. [19] proposed a planar fault model to further reduce the disabled fault-free nodes.

Ho and Stockmeyer [17] proposed a multiple-round dimension-order routing algorithm. This algorithm, which is the baseline routing algorithm of this work, is separated into two routing phases. The routing phase from the source node to the intermediate node is the first phase, and the second phase routes packets from the intermediate node to the destination. If more than one intermediate node is utilized, more routing phases are needed. Note that within each routing phase the dimension-order routing is adopted, and two routing phases are connected in a pipeline manner, i.e., the intermediate node forwards a flit as soon as possible. Since each routing phase is allocated a separate VC and a routing phase depends on another in a strictly increasing order, the network is deadlock free. Note that if two nodes cannot communicate even with intermediate nodes, one of them should be declared as a lamb, which can be used to forward but not to send and receive packets.

With Duato's theory, there is usually an escape VC provided to the routing subfunction [12], [18], [20]. For example, Chalasani and Boppana also proposed an adaptive version of their fault-tolerant routing in [12]. Three of the four VCs are adaptive, and the other is the e-cube channel. Puente et.al. [18] proposed the Immunet that avoids deadlock by providing a safe virtual network visited under dimension-order routing. Gómez et.al. [20] also proposed a multiple-round routing, which is another related work of this paper, allows to utilize adaptive routing within each routing phase. All routing phases share an adaptive VC, and each of them is allocated a separate escape channel.

Turn models are usually utilized to design fault-tolerant routing algorithms for networks without VCs [10], [13], [16], [22]–[24], [26], [27]. Glass and Ni [10] first extended the original negative-first routing to tolerate one fault. Chen

and Chiu [13] (corrected by Holsmark and Kumar [22]) proposed a routing algorithm to tolerate convex faults. Later, this algorithm is extended by Fukushima et.al. [27] to disable fewer fault-free nodes. Zhang et.al. [23] proposed a reconfigurable router that could tolerate one fault or one convex faulty-block. Fick et.al. [24] proposed a distributed algorithm to update the routing table, but their method does not guarantee the deadlock-freeness. In [26], we extended [17] and proposed the idea of turn-legally intermediate nodes.

It is important to distinguish our work from previous solutions. To the best of our knowledge, there are two popular multiple-round routing algorithms proposed by Ho and Stockmeyer [17] and Gómez et.al. [20] respectively.

The major common feature of [17] and [20] is that inserting an intermediate node always means adding VCs. With the proposed NMR-DOR, however, turn-legally intermediate nodes can be inserted without adding VCs. For example, with a single intermediate node, [17] requires two VCs, [20] requires three, and the NMR-DOR does not need VCs if the intermediate node is turn-legally. In other words, given the number of VCs, the NMR-DOR could utilize more intermediate nodes to reduce the number of unreachable nodes.

Another common feature of [17] and [20] is that virtual channels are accessed in a static order. For example, with the algorithm proposed in [17], packets could not start the $i_{th}$ routing phase in $VC_i$ unless they have finished all routing phases in $VC_j$, where $j < i$. For [20], the adaptive VC is shared, but the escape VCs are also accessed in a strictly increasing order. This kind of restriction leads to an unbalanced traffic, i.e., lower indexed VCs carry more traffic, because many packets do not have the higher routing phases. With the proposed NMR-DOR, However, packets that only use turn-legally intermediate nodes can arbitrarily select VCs. As validated in Sect. 6, this difference significantly improves the NoC performance.

## 3. The Turn-Legally Intermediate Nodes

A node $I$ is a turn-legally intermediate node between source, $S$, and destination, $D$, iff the turn made by two dimension-order routing phases, from $S$ to $I$ and from $I$ to $D$, is allowed by the adopted turn model. However, if a node $I$ is an *available* turn-legally intermediate node, it should meet the following three conditions:

- $I$ is reachable from $S$ using dimension-order routing,
- $I$ could reach $D$ using dimension-order routing,
- $I$ is turn-legally for $S$ and $D$.

In general, finding a turn legally intermediate node partially depends on the adopted turn model. For example, the $SW$ turn is forbidden by the west-first routing, but is allowed by the negative-first and north-last routing [5]. Therefore, the first step of the proposed technique is to determine which turn model is adopted. We summarize the rule as:

- In NMR-DOR, the adopted turn model doesn't forbid

any turn taken by the dimension-order routing.

The above rule means that any allowable turn of dimension-order routing should also be allowed by the adopted turn model. For example, if the dimension-order routing is XY routing, the turn model cannot be the negative-first routing. The reason is that XY routing allows the *ES* turn, which is forbidden by the negative-first routing.

Given the dimension-order routing and adopted turn model, whether an intermediate node is turn-legally is determined by the relative position between source node, $S$, intermediate node, $I$, and the destination, $D$. As shown in Fig. 3, if node $I$ is selected, 2-rounds of dimension-order routing makes an $SE$ turn on it. According to west-first routing, $SE$ is an allowable turn, so node $I$ is said turn-legally. If node $F$ is selected, however, a forbidden turn, $SW$, is introduced. Hence, node $F$ is turn-illegally.

By analyzing all possible relative positions of nodes $S$, $I$ and $D$, we summarize all turn-legally intermediate nodes for dimension-order routing in Table 1, where $x$ and $y$ denote the coordinates of turn-legally intermediate node in x- and y-dimension respectively. $S_x$, $S_y$, $D_x$ and $D_y$ are the coordinates of nodes $S$ and $D$, and $n_x$ and $n_y$ are the width of network in x- and y-dimension respectively. For each node pair $(i, j)$, we utilize a $N$-length vector, $T$, to represent the turn-legally intermediate nodes as shown in Eq. (1), where $N$ is the network size.

$$T_{i,j}(t) = \begin{cases} 1, & node\ t\ is\ turn\ legally\ or\ t \in \{i, j\}; \\ 0, & otherwise. \end{cases} \quad (1)$$

If more than one intermediate node is inserted, whether a node is turn-legally partially depends on who is the preceding intermediate node, and this requires the connectivity information between nodes. Two nodes are connected if
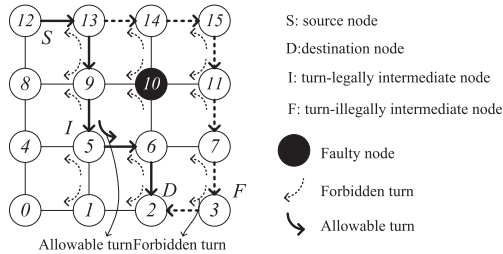
there is at least one fault-free path between them, and we name them as a reachable pair. Otherwise, they are defined as unreachable pair.

If the maximal number of intermediate nodes is zero, NMR-DOR routing is equivalent with dimension-order routing. Under dimension-order routing (we assume the XY routing here), whether two nodes are connected depends on whether there are faults lying in the XY routing path between source and destination. As shown in Eq. (2), we use a $N \times N$ matrix $R$, to represent the connectivity from $node_i$ to $node_j$ under dimension-order routing, where $F$ refers to the faults set.

$$R_{i,j} = \begin{cases} 0, & \exists\ f,\ where\ f \in F\ and \\ & (f_x = j_x\ and\ \min\{i_y, j_y\} \le f_y \le \max\{i_y, j_y\})\ or \\ & (f_y = i_y\ and\ \min\{i_x, j_x\} \le f_x \le \max\{i_x, j_x\}), \\ 1, & otherwise. \end{cases}$$

$$(2)$$

**Example 1:** Let us reconsider the network shown in Fig. 3. Dimension-order routing is XY routing, turn model is west-first routing. The connectivity matrix is shown in Eq. (3). Particularly, $R_{*,10}$ and $R_{10,*}$ are always 0 because $node_{10}$ is faulty. Assume the source is $node_0$, only the XY routing path to $node_{14}$ is blocked by the fault. Thus, $R_{0,14} = 0$. Note that $R_{i,i} = 1$, because any node can reach itself.

$$R = \begin{array}{c} \begin{array}{cccccccccccccccc} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 \end{array} \\ \begin{array}{c} 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \\ 10 \\ 11 \\ 12 \\ 13 \\ 14 \\ 15 \end{array} \left( \begin{array}{cccccccccccccccc} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \end{array} \right) \end{array}$$

$$(3)$$

If the maximal number of intermediate nodes is one, which means that NMR-DOR could route packet first to a turn legally intermediate node. In this situation, we extend the connectivity matrix to be $R^{(2)}$, where $R_{i,j}^{(2)} = 1$ indicates that there is a turn-legally intermediate node, through which



**Fig. 3** Example of turn legally and turn illegally intermediate nodes: DOR is assumed as XY routing, west-first is adopted as the turn model.

**Table 1** Example of turn legally intermediate nodes in 2D meshes.

| DOR | Turn Model | Turn legally intermediate nodes |
|---|---|---|
| XY routing | East-first | $\{(x,y)|D_x \le x \le n_x - 1,\ 0 \le y \le (n_y - 1),\ x\ and\ y\ are\ integers\}$ |
| | West-first | $\{(x,y)|0 \le x \le D_x,\ 0 \le y \le (n_y - 1),\ x\ and\ y\ are\ integers\}$ |
| | Sout-last | $\{(x,y)|0 \le x \le (n_x - 1),\ S_y \le y \le (n_y - 1),\ x\ and\ y\ are\ integers\}$ |
| | North-last | $\{(x,y)|0 \le x < (n_x - 1),\ 0 \le y \le S_y,\ x\ and\ y\ are\ integers\}$ |
| YX routing | South-first | $\{(x,y)|0 \le x < (n_x - 1),\ 0 \le y \le D_y,\ x\ and\ y\ are\ integers\}$ |
| | North-first | $\{(x,y)|0 \le x \le (n_x - 1),\ D_y \le y \le (n_y - 1),\ x\ and\ y\ are\ integers\}$ |
| | East-last | $\{(x,y)|0 \le x \le S_x,\ 0 \le y \le (n_y - 1),\ x\ and\ y\ are\ integers\}$ |
| | West-last | $\{(x,y)|S_x \le x \le (n_x - 1),\ 0 \le y \le (n_y - 1),\ x\ and\ y\ are\ integers\}$ |

$i$ could reach $j$. Mathematically, we calculate $R^{(2)}$ as shown in Eq. (4).

$$R_{i,j}^{(2)} = (R_{i,*} \cap T_{i,j}) \times R_{*,j} \qquad (4)$$

The equation (4) is reasonable since its right side indicates that there is a node $t$, where

- node $t$ is reachable from $i$ ($R_{i,t} = 1$),
- node $t$ is a turn-legally intermediate node for $(i, j)$ ($T_{i,j}(t) = 1$),
- node $t$ could reach $j$ ($R_{t,j} = 1$).

**Example 2:** Let us reconsider the above example. The 2-round connectivity matrix is shown in Eq. (5). The results show that all fault-free nodes, except $node_{11}$, can reach all other nodes with a turn-legally intermediate node. For example $node_0$ cannot reach $node_{14}$ under XY routing. However, with a turn-legally intermediate node, e.g., $node_{12}$, $node_0$ can reach $node_{14}$, i.e., $R_{0,14}^{(2)} = 1$. Unfortunately, $node_{11}$ still cannot reach some nodes, e.g., $node_0$, because its west neighbor is faulty and the turn model is west-first. In Sect. 5, we will discuss the way to handle this kind of problem.

$$R^{(2)} = \begin{array}{c} \\ 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \\ 10 \\ 11 \\ 12 \\ 13 \\ 14 \\ 15 \end{array} \begin{pmatrix} \begin{array}{cccccccccccccccc} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 1 & 1 & 1 \end{array} \end{pmatrix}$$

$$(5)$$

Now, we consider the situation when the maximal number of intermediate nodes is two. Without loss of generality, we assume the first turn-legally intermediate node is $k$, and the second intermediate node is $t$. Node $t$ is turn-legally iff it meets the following four conditions:

- node $k$ is reachable from $i$ ($R_{i,k} = 1$),
- node $k$ is a turn legally intermediate node for $(i,t)$ ($T_{i,t}(k) = 1$),
- node $k$ could reach $t$ ($R_{k,t} = 1$),
- node $t$ is a turn-legally intermediate node for $(k, j)$ ($T_{k,j}(t) = 1$).

Where, the first three conditions guarantee that using node $t$ as the second intermediate node makes a legal turn on its preceding node, $k$. The last condition guarantees that the turn on node $t$ is also legal. Therefore, all turns, which are incurred by using node $t$ as the second intermediate node, are all legal. Mathematically, we extend $T_{i,j}$ to be $T_{i,j}^{(2)}$ as shown in Eq. (6).

$$T_{i,j}^{(2)}(t) = \bigcup_{k=1}^{N} R_{i,k} \cdot T_{i,t}(k) \cdot R_{k,t} \cdot T_{k,j}(t) \qquad (6)$$

With vector $T_{i,j}^{(2)}$, the connectivity matrix for 3-rounds of dimension-order routing, $R_{i,j}^{(3)}$, could be calculated in the similar way as $R_{i,j}^{(2)}$ as shown in Eq. (7). Note that $T_{i,j}^{(2)}(t) = 1$ implies that $R_{i,t}^{(2)} = 1$ according to Eq. (6). The reason why we also include $R_{i,*}^{(2)}$ into Eq. (7) is to keep compatible with Eq. (4).

$$R_{i,j}^{(3)} = (R_{i,*}^{(2)} \cap T_{i,j}^{(2)}) \times R_{*,j} \qquad (7)$$

Finally, we generalize the $T$-vector to the situation that $n$ turn legally intermediate nodes are utilized as shown in Eq. (8), and then generalize the $R$-matrix to the situation that $n$-rounds of dimension-order routing is implemented as shown in Eq. (9).

$$T_{i,j}^{(n)}(t) = \bigcup_{k=1}^{N} R_{i,k}^{(n-1)} \cdot T_{i,t}^{(n-1)}(k) \cdot R_{k,t} \cdot T_{k,j}(t) \qquad (8)$$

$$R_{i,j}^{(n)} = (R_{i,*}^{(n-1)} \cap T_{i,j}^{(n-1)}) \times R_{*,j} \qquad (9)$$

## 4. The NMR-DOR Routing

As described in [20], with a multiple-round routing algorithm, packets should carry the information of intermediate nodes. At each node, the router applies the dimension-order routing with the coordinates of itself and the destination (maybe an intermediate node) as parameters.

Once the output port is determined, the node should select an VC belonging to that port. Since all intermediate nodes are determined based on a same turn model, the corresponding channel dependence graph is absolutely acyclic. Thus, VCs can be selected with any existing technique. Generally, the selection can be random or based on the VC status [30].

The NMR-DOR routing is shown in Algorithm 1. Particularly, the function *get-dest* is utilized to get the coordinates of current destination. This destination can be the next intermediate node or the final destination. The function first compares the coordinates of current node and the first intermediate node indicated by the flit. If the comparison does not match, then the first intermediate node is the current destination. Otherwise, the first intermediate node is shifted out of the flit. If the flit carries other intermediate nodes, then the next intermediate node is the current destination. Otherwise, the destination is the final destination.

With the coordinates of current node and destination node, getting output port and VC is straightforward. This routing algorithm is deadlock free guaranteed by the turn model. Livelock is also impossible with finite intermediate nodes. We prove them as the following theorem.

**Theorem 1:** NMR-DOR routing is deadlock and livelock free.

---

**Algorithm 1** Pseudo code of NMR-DOR routing

**Input:** *cur*: current node; *f*: input flit; *input_vc*: input virtual channel; *sel()*: select function.
**Output:** *output*: output port; *output_vc*: output virtual channel.
 1: *dst* = get-dest(*cur*,*f*)
 2: *output* = DOR(*cur*, *dst*)
 3: **if** *output*=local **then**
 4:     send flit to local core using any available VC;
 5: **else**
 6:     *output_vc*=*sel()*;
 7: **end if**

---

**Proof:**

**Deadlock Freeness.** Without loss of generality, we assume the adopted turn model is west-first routing and the dimension-order routing is XY routing. Turn-legally intermediate nodes never introduces NW and SW turns into the network, because the turn model (west-first routing) forbids them. Furthermore, XY routing does not introduce them neither. Thus, NW and SW turns never be used by the NMR-DOR routing. As proved in [5], a network without NW and SW turns is deadlock free. Thus, NMR-DOR is deadlock free.

**Livelock Freeness.** NMR-DOR always utilizes a finite number of intermediate nodes, resulting into a finite number of routing phases. Since each routing phase adopts the XY routing, which is livelock free, the NMR-DOR is livelock free.   ■

## 5. Complementary Mechanisms

As discussed in Example 2, given a turn model, not every node pair can find an available turn-legally intermediate node. In the following of this section, we give three kinds of complementary mechanisms to reduce this kind of node pairs.
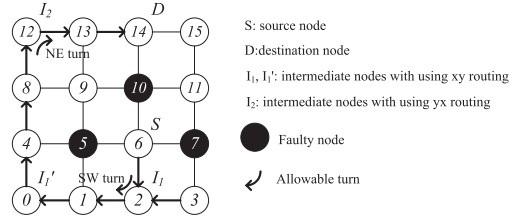
### 5.1 Different Turn Models

Let us reconsider the Example 2, the adopted turn model is the west-first routing. Thus, according to Table 1, all turn-legally intermediate nodes are never on the east of the destination. If, unfortunately, the source node, e.g. $node_{11}$, is on the east of the destination, e.g., $node_0$, and has a faulty west neighbor, e.g., $node_{10}$. Then, for any turn-legally intermediate node, $T$, where $T_x \le D_x \le S_x$, the routing path from $S$ to $T$ is blocked by $node_{10}$. However, this does not mean ($node_{11}$, $node_0$) is unreachable. We redraw Fig. 3 here as shown in Fig. 4. If we select node, $I$, as the intermediate node, then a $SW$ turn is introduced on $I$. According to west-first routing, of course, $I$ is a turn-illegally intermediate node. However, if we adopt a different turn model, e.g., the east-first routing, then $I$ is turn-legally.

This observation motivates us to utilize different turn models for different node pairs. However, with different turn models, we need another VC to avoid deadlock. Generally,
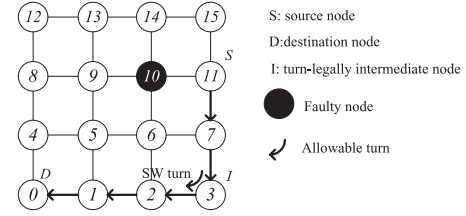


**Fig. 4**   Using other turn models instead of the west-first.



**Fig. 5**   Using different DOR routing.

packets utilizing different turn models are allocated to separate virtual channels. Therefore, the network is still deadlock free, because 1) the channel dependence graph within each VC is acyclic and 2) packets belonging to different VCs do not depend on each other. For some node pairs, they maybe reachable with different turn models. In this situation, they could select VC based on the network status. However, once a packet selects an VC, it does not change it until it reaches the destination.

### 5.2 Different Dimension-Order Routing

For some node pairs, they maybe unreachable with all turn models that are compatible with XY routing. For example, as shown in Fig. 5, the neighbors of source node are all faulty except the south one. There is a fault-free path from $S$ to $D$ with 2 intermediate nodes. However, if we assume the turn model is west-first or south-last routing, then the first intermediate node, $I_1$, is turn-illegally due to the introduced $SW$ turn. With the east-first or north-last routing, however, the second intermediate node, $I_2$, is turn-illegally due to the introduced $NE$ turn.

To solve this problem, we propose to utilize different dimension-order routing within different VCs. For example, if the YX routing is adopted in each routing phase, $S$ can reach $D$ with only one intermediate node, say $I_1'$. Moreover, if the south-first or east-last routing is adopted as the turn model, $I_1'$ is turn-legally.

With different dimension-order routing, packets are routed in a similar way as that with different turn models. In general, packets with different dimension-order routing are allocated to separate VCs, and packets are not allowed to change VC along the routing path. The network is deadlock free because 1) within each VC, network is deadlock free and 2) packets belonging to different VCs do not depend on each other. Also, for node pairs, which are reachable with both dimension-order routing algorithms, they can
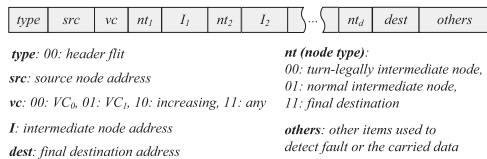
select VC based on the network status.

## 5.3 With Normal Intermediate Nodes

The third complementary mechanism is a direct extension of the first two. As discussed above, implementing the first two complementary mechanisms requires adding an extra VC, resulting in totally two VCs per physical channel in the network. Therefore, we could treat the node where packets change the VC as an intermediate node as [17] and [20] did. Since this kind of intermediate node is not required to be turn-legally, we name them as normal intermediate nodes to distinguish them from turn-legally ones.

To avoid deadlock, we request the packets utilizing normal intermediate nodes to access VCs in a monotonically increasing order. That is to say, once the source node detects a packets using normal intermediate nodes, it sends this packets to the first VC. This packet is transferred in the first VC, maybe through several turn-legally intermediate nodes, until reaches the first normal intermediate node where it starts the routing phase in the second VC. As discussed above, packets utilizing normal intermediate nodes tend to lead unbalanced traffic load. Thus, they should only be used to connect the node pairs that are unreachable with turn-legally intermediate nodes. In next section, we will prove that the number of this kind of node pairs is very small.

## 5.4 The Packet Format

To support the complementary mechanisms, we should define a new packet format. As shown in Fig. 6, the header flit is identified by the first two bits, *type*. Each node address, e.g., source node, *src*, intermediate node, *I*, and destination node, *dest*, occupies $\log_2 N$ bits, where $N$ is the network size. The third field, *vc*, tells us which VC should be used. Particularly, "*vc=10*" means that this packet will utilize normal intermediate nodes, and should access VCs in a strictly increasing order. For example, the packet is transmitted in $VC_0$ until it reaches the first normal intermediate node, then it is switched to $VC_1$. During the phases from one normal intermediate node to another, packets may encounter some turn-legally intermediate nodes, but they do not switch VC at them. If *vc=11*, that means the packets could utilize any VC based on the network status. Finally, for each node address, except the source, a two bits field, *nt* (node type), is added. It tells us which kind of node that address points to, e.g., a turn-legally intermediate node, a normal intermediate node, or the final destination.

| type | src | vc | nt₁ | I₁ | nt₂ | I₂ | ... | ntₐ | dest | others |
|------|-----|-----|-----|-----|-----|-----|-----|-----|------|--------|

**type**: 00: header flit
**src**: source node address
**vc**: 00: $VC_0$, 01: $VC_1$, 10: increasing, 11: any
**I**: intermediate node address
**dest**: final destination address

**nt (node type)**:
00: turn-legally intermediate node,
01: normal intermediate node,
11: final destination

**others**: other items used to detect fault or the carried data

**Fig. 6** The format of header flit of NMR-DOR routing with all complementary mechanisms.

## 6. Evaluation

In this section, we will evaluate the proposed NMR-DOR by comparing it with existing multiple-round routing algorithms, i.e., [17] and [20]. Since NMR-DOR differs with them only in the way of utilizing intermediate nodes, thus the router architecture will be similar. Therefore, we omit the comparison about router area and frequency, and focus on two system metrics, the number of unreachable node pairs and the packet latency, which can highlight the differences between NMR-DOR and prior work.

### 6.1 The Number of Unreachable Node Pairs

According to [17], at least one node of each unreachable node pair should be declared as a lamb node, which could be utilized to forward but not to send and receive packets. Clearly, reducing the number of unreachable node pairs helps to reduce the lamb nodes. In this paper, we say a node pair, $(a, b)$, is unreachable, if at least one of the two routing paths, $a \rightarrow b$ and $b \rightarrow a$, is blocked by faults. Specially, for fully adaptive routing algorithms, if one of the possible routing paths between $a$ and $b$ is blocked by faults, we say $(a, b)$ is fully-adaptive-unreachable. Generally, if a node pair is unreachable, it is fully-adaptive-unreachable, but the opposite is not true. Therefore, compared with [17], the basic routing (I-version) of [20] that adopts adaptive routing in each phase always leads to more unreachable node pairs. [20] extended their routing algorithm to support deterministic routing (D-extension) and misrouting (M-extension), and concluded that (I+D) is the most efficient way due to the high cost of misrouting. In fact, the (I+D) version of [20] will lead to the same number of unreachable node pairs as [17]. Therefore, we only compare NMR-DOR with [17]. Note that, we do not take into account the faulty nodes and the nodes surrounded by faults, because it is impossible to provide faulty free path to these kinds of nodes.

In this experiment, we assume an $8 \times 8$ mesh with a node faulty rate that is smaller than 10%. For the network with one or two faults, we exhausted all possible fault distributions, i.e., 64 and 2016 distributions respectively. For networks with more than two faults, we randomly select 10,000 different fault distributions due to the unreasonable long time for exhausting. All reported results are the average of that for all simulated fault distributions.

At first, we assume that there is one VC per physical channel, and the routing function of Ho's algorithm [17] becomes as same as XY routing. We report the average percentage of unreachable node pairs in Table 2, where the first column indicates the routing algorithm utilized in each routing phase, the second column indicates which turn model is adopted by the NMR-DOR, and the last 6 columns show the results with respect to the networks containing 1 to 6 faults. From these results, we could find that the proposed NMR-DOR considerably improves the reliability of [17]. More than 50% of the unreachable node pairs in [17] are saved.

**Table 2**  Percentage of unreachable node pairs of an 8×8 mesh with one VC. EF (east-first), WF (west-first), NL (north-last), SL (south-last), EL (east-last), WL (west-last), NF (north-first), SF (south-first).

| Routing | Turn Model | fn=1 | fn=2 | fn=3 | fn=4 | fn=5 | fn=6 |
|---|---|---|---|---|---|---|---|
| [17] (XY) | | 12.84 | 22.64 | 30.111 | 35.65 | 39.84 | 42.80 |
| XY | EF | 4.64 | 8.83 | 12.58 | 16.00 | 19.09 | 21.69 |
| | WF | 4.64 | 8.83 | 12.63 | 15.93 | 19.12 | 21.74 |
| | NL | 4.64 | 8.83 | 12.58 | 15.97 | 19.08 | 21.71 |
| | SL | 4.64 | 8.83 | 12.60 | 15.96 | 19.09 | 21.74 |
| YX | EL | 4.64 | 8.83 | 12.63 | 15.93 | 19.12 | 21.74 |
| | WL | 4.64 | 8.83 | 12.58 | 16.00 | 19.09 | 21.69 |
| | NF | 4.64 | 8.83 | 12.58 | 15.97 | 19.08 | 21.71 |
| | SF | 4.64 | 8.83 | 12.60 | 15.96 | 19.09 | 21.74 |

**Table 3**  Percentage of unreachable node pairs of an 8×8 mesh with two VCs and no normal intermediate nodes. EF (east-first), WF (west-first), NL (north-last), SL (south-last), EL (east-last), WL (west-last), NF (north-first), SF (south-first), fn (number of fault).

| Solution | Routing | Turn Model | fn=1 | fn=2 | fn=3 | fn=4 | fn=5 | fn=6 |
|---|---|---|---|---|---|---|---|---|
| [17] | XY-XY | | 0 | 0.0138 | 0.0659 | 0.1752 | 0.4194 | 0.7665 |
| Proposed | XY-XY | EF-WF | 0 | 0.0138 | 0.0659 | 0.1752 | 0.4194 | 0.7665 |
| | | EF-NL | 1.3020 | 2.6557 | 4.0289 | 5.4633 | 7.0128 | 8.5096 |
| | | EF-SL | 1.3020 | 2.6557 | 4.0069 | 5.4849 | 7.0233 | 8.4713 |
| | | WF-NL | 1.3020 | 2.6557 | 4.0520 | 5.4292 | 7.0784 | 8.5187 |
| | | WF-SL | 1.3020 | 2.6557 | 4.0524 | 5.4421 | 7.0322 | 8.4986 |
| | | NL-SL | 0 | 0.0138 | 0.0659 | 0.1752 | 0.4194 | 0.7665 |
| | XY-YX | EF-EL | 0.0868 | 0.4782 | 1.1248 | 1.9460 | 3.0511 | 4.1934 |
| | | EF-WL | 0.0868 | 0.4782 | 1.1088 | 1.9692 | 3.0639 | 4.1786 |
| | | EF-NF | 0.3472 | 1.2850 | 2.6464 | 4.2830 | 6.2123 | 8.1166 |
| | | EF-SF | 0.0434 | 0.1069 | 0.2042 | 0.3459 | 0.5859 | 0.8280 |
| | | WF-EL | 0.0868 | 0.4782 | 1.1129 | 1.9601 | 3.0419 | 4.1842 |
| | | WF-WL | 0.0868 | 0.4782 | 1.1191 | 1.9412 | 3.0513 | 4.2012 |
| | | WF-NF | 0.0434 | 0.1069 | 0.2040 | 0.3401 | 0.5934 | 0.8861 |
| | | WF-SF | 0.3472 | 1.2850 | 2.6464 | 4.2830 | 6.2123 | 8.1166 |
| | | NL-EL | 0.3472 | 1.2850 | 2.6379 | 4.2820 | 6.1827 | 8.1264 |
| | | NL-WL | 0.0434 | 0.1069 | 0.2037 | 0.3429 | 0.6036 | 0.8965 |
| | | NL-NF | 0.0868 | 0.4782 | 1.1191 | 1.9412 | 3.0513 | 4.2012 |
| | | NL-SF | 0.0868 | 0.4782 | 1.1088 | 1.9692 | 3.0639 | 4.1786 |
| | | SL-EL | 0.0434 | 0.1069 | 0.2051 | 0.3386 | 0.5925 | 0.8798 |
| | | SL-WL | 0.3472 | 1.2850 | 2.6379 | 4.2820 | 6.1827 | 8.1264 |
| | | SL-NF | 0.0868 | 0.4782 | 1.1129 | 1.9601 | 3.0419 | 4.1842 |
| | | SL-SF | 0.0868 | 0.4782 | 1.1248 | 1.9460 | 3.0511 | 4.1934 |

Another conclusion is that different turn models produce almost the same results due to the rotation symmetry. For some applications, where each processing core is cheap, the one-VC NMR-DOR is attractive due to its ultra-low overhead. However, for applications with high cost processing cores, the complementary mechanisms are needed.

With two VCs, NMR-DOR could utilize different turn models and dimension-order routing in each VC. First, we assume that NMR-DOR adopts the XY routing algorithm in both VCs, but utilize different turn models. There are 6 different combinations, and the results are shown in $2_{nd}$ to $7_{th}$ rows of Table 3. We could find that the combinations, EF-WF (east-first + west-first) and NL-SL (north-last + south-last), get the best results. This is reasonable, because east-first and west-first (north-last and south-last) has complementary turn-legally intermediate nodes as shown in Table 1. Compared with [17] shown in the first row, NMR-DOR produces the same number of unreachable node pairs. However, the most important difference is that NMR-DOR could arbitrarily select VCs to balance traffic. Second,

we assume the NMR-DOR adopt different dimension-order routing in different VCs, and the results of all possible 16 combinations are shown in the $8_{th}$ to $23_{rd}$ rows of Table 3. The best results are got by 4 rotation symmetric combinations: EF-SF, WF-NF, NL-WL, and SL-EL. However, their results are not as good as that of [17]. The main reason is that they don't have complementary turn-legally intermediate nodes.

In general, as discussed in [17], this level of the amount of unreachable node pairs (smaller than 0.2%) is acceptable for most applications. In this paper, we further reduce it by applying the third mechanism that does not introduce any extra area and timing overhead to routers. The results are shown in Table 4, the best are got by two rotation symmetric combinations, EF-SF and WF-NF, which could tolerate all 2-faults distributions. For networks with more than two faults, more than 90% of the unreachable node pairs of [17] can be saved. We should emphasize again that the packets utilizing normal intermediate nodes cannot be utilized to balance traffic. The number of this kind of packets can be

**Table 4**  Percentage of unreachable node pairs of an 8×8 mesh with two VCs and with normal intermediate nodes. EF (east-first), WF (west-first), NL (north-last), SL (south-last), EL (east-last), WL (west-last), NF (north-first), SF (south-first), fn (number of fault).

| Solution | Routing | Turn Model | fn=1 | fn=2 | fn=3 | fn=4 | fn=5 | fn=6 |
|---|---|---|---|---|---|---|---|---|
| [17] | XY-XY | | 0 | 0.008 | 0.034 | 0.077 | 0.160 | 0.250 |
| Proposed | XY-XY | WF-WF | 0 | 0.0007 | 0.0080 | 0.0226 | 0.0773 | 0.1494 |
| | | EF-WF | 0 | 0.0052 | 0.02119 | 0.05157 | 0.1236 | 0.2031 |
| | | EF-NL | 0 | 0.0034 | 0.0182 | 0.0434 | 0.1188 | 0.2046 |
| | | EF-SL | 0 | 0.0034 | 0.0175 | 0.0470 | 0.1209 | 0.2089 |
| | | WF-NL | 0 | 0.0034 | 0.0167 | 0.0461 | 0.1167 | 0.2008 |
| | | WF-SL | 0 | 0.0034 | 0.0181 | 0.0455 | 0.1198 | 0.2113 |
| | | NL-SL | 0 | 0.0052 | 0.0221 | 0.0515 | 0.1252 | 0.2078 |
| | XY-YX | EF-EL | 0.0434 | 0.1486 | 0.3238 | 0.5545 | 0.9258 | 1.3013 |
| | | EF-WL | 0.0434 | 0.1486 | 0.3170 | 0.5572 | 0.9369 | 1.3248 |
| | | EF-NF | 0.3472 | 0.9860 | 1.8786 | 2.9410 | 4.2657 | 5.5917 |
| | | EF-SF | 0 | 0 | 0.0028 | 0.0091 | 0.0384 | 0.0695 |
| | | WF-EL | 0.0434 | 0.1486 | 0.3206 | 0.5522 | 0.9225 | 1.3106 |
| | | WF-WL | 0.0434 | 0.1486 | 0.3198 | 0.5583 | 0.9394 | 1.3140 |
| | | WF-NF | 0 | 0 | 0.0029 | 0.0091 | 0.0384 | 0.0683 |
| | | WF-SF | 0.3472 | 0.9860 | 1.8786 | 2.9410 | 4.2657 | 5.5917 |
| | | NL-EL | 0 | 0.02558 | 0.0930 | 0.2144 | 0.4410 | 0.7299 |
| | | NL-WL | 0.0434 | 0.1052 | 0.1894 | 0.2967 | 0.4810 | 0.6462 |
| | | NL-NF | 0.0434 | 0.1486 | 0.3198 | 0.5583 | 0.9394 | 1.3140 |
| | | NL-SF | 0.0434 | 0.1486 | 0.3170 | 0.5572 | 0.9369 | 1.3248 |
| | | SL-EL | 0.0434 | 0.1052 | 0.1908 | 0.2909 | 0.4691 | 0.6365 |
| | | SL-WL | 0 | 0.0255 | 0.0930 | 0.2144 | 0.4410 | 0.7299 |
| | | SL-NF | 0.0434 | 0.1486 | 0.3206 | 0.5522 | 0.9225 | 1.3106 |
| | | SL-SF | 0.0434 | 0.1486 | 0.3238 | 0.5545 | 0.9258 | 1.3013 |

got by reducing the results shown in Table 3 by the results shown in Table 4. We could find that their amount is very small. For example, 0.1069% packets should utilize normal intermediate nodes in WF-NF combination with 2 faults.
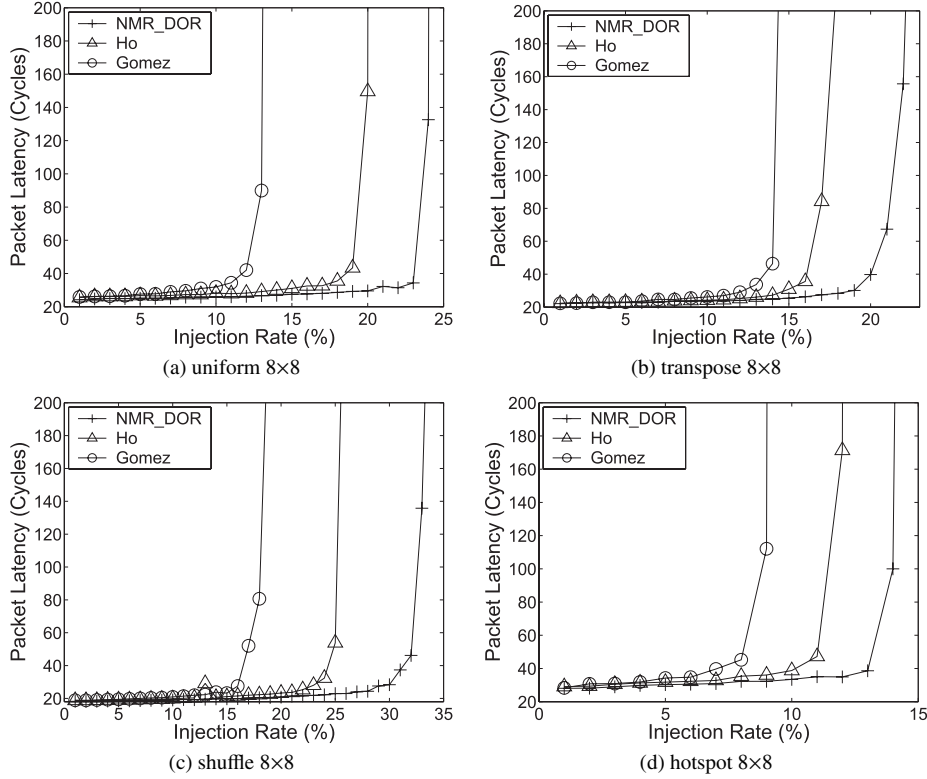
## 6.2 The Packet Latency

In this experiment, we utilize a cycle-accurate NoC simulator, the BookSim [31], to carry out the simulations. Book-Sim provides a flexible way to configure NoC parameters, such as network topology, and routing algorithm. By maintaining a global clock, BookSim could keep the simulation cycle-accurate. In the following simulations, router pipeline depth is assumed as four and link traversal latency is one. The round-robin policy is adopted to select requesting inputs in both VC and Switch allocation stages. Although we assume a canonical router architecture instead of the aggressive state-of-the-art ones, such as lookahead routing and speculation, it is fair for evaluating fault-tolerant routing algorithms. For the proposed NMR-DOR and [17], we assume that there are two VCs per physical channel, and each VC contains an FIFO with eight entries to hide the round-trip latency of flow-control credits. For [20], we assume three VCs per physical channel to support adaptive routing, but each VC has four entries for fair comparison. We should emphasize that [20] utilize the Duato's theory to avoid deadlock, thus it should not reallocate an VC unless it is empty and freed by the preceding packet. However, for the NMR-DOR and [17], we could reallocate an VC as soon as it receives the tail flit of the preceding packet. This kind of difference significantly affects the network performance as we will see in the following.

In the presence of faults, a source node $S$ may require $n$ intermediate nodes, $I_1$, $I_2$, ...,$I_n$, to reach the destination node $D$. Thus, packets between $S$ and $D$ should follow the routing path: $S \rightarrow I_1 \rightarrow I_2 \rightarrow, ..., \rightarrow I_n \rightarrow D$. At most times, there is more than one path to choose. In this case, the following policy is utilized to select path for node pairs:

1. Arrange node pairs in the increasing order of the number of alternate paths,
2. For the first node pair, select the path leading to the minimal variation in channel load,
3. Remove the first node pair, repeat 2 until all node pairs have been processed.

The current version of the selection policy is utilized to show the ability of routing algorithms for load-balancing. Thus, we exhaust all legal paths for each node pair. For large networks, this is a little time consuming. For example, it cost us hours to specify paths for all node pairs in an 16×16 mesh with 26 faults. In general, there is a trade-off between the algorithm complexity and the network performance. Further reducing the algorithm complexity without degrading the network performance is an interesting and challenging job for future work. Note that finding legal paths for each node pair is a typical depth-first searching, thus its detailed description is omitted.

We first assume an $8 \times 8$ mesh with six faulty nodes to bound the network performance. Packet size is assumed to be uniformly distributed between 1 and 8 to simulate the different packet header sizes. Furthermore, we select the fault distribution that maximizes the percentage of unreachable node pairs (based on the simulation results of above experiment), because this will show

**Fig. 7** The average packet latency in 8×8 meshes. The proposed NMR-DOR is shown as a solid line with crosses, [17] (marked as Ho) is shown as a solid line with up-triangles, and [20] (Marked as Gomez) is shown as a solid line with circles. Under hotspot traffic, the hotspot node, $node_{27}$, has extra 10% possibility for receiving packets.

us the worst-case performance. Specifically, the faults are $\{node_{12}, node_{21}, node_{25}, node_{30}, node_{35}, node_{50}\}$. For the proposed NMR-DOR, the WF-NF combination is selected due to its high performance got in the above experiment.

With the uniform traffic pattern, each node sends a packet to all its reachable destinations with the same possibility. As shown in Fig. 7 (a), the proposed NMR-DOR gets the best performance. The reasons are two folds. First, compared with [17], the proposed NMR-DOR could reduce packet latency by evenly distributing packets, which only utilize turn-legally intermediate nodes, among different VCs. Second, compared with [20], the packet latency is further reduced by improving the utilization of VC buffers due to the different VC reallocation time.

With the transpose traffic pattern, each source node, $S$, will only send packets to a destination node, $D$, where $D_i = S_{(i+3)\%6}$ ($i \in [0, 5]$) and $S$ and $D$ are neither the faulty nor lamb nodes. As shown in Fig. 7 (b), the relative performance among these three algorithms does not change, i.e., the NMR-DOR outperforms others and [17] gets better results than [20]. However, unlike the uniform traffic pattern that inherently balances the packets, transpose traffic patterns will cause more traffic contentions. Thus, the proposed NMR-DOR, which can evenly distribute packets among VCs, does much better than others. [17] and [20] get similar results, and [17] is better due to a higher utilization rate of VC buffers.

With the shuffle traffic pattern, each source node, $S$, will only send packets to a destination node, $D$, where $D_i = S_{(i-1)\%6}$ ($i \in [0, 5]$) and $S$ and $D$ are neither the faulty nor lamb nodes. As shown in Fig. 7 (c), the proposed NMR-DOR also gets the best performance, and [17] is better than [20]. Initially they get similar results, but [20] first gets saturated when the flit injection rate is higher than 18%. Later, [17] gets saturated when the flit injection rate is higher than 25%. Finally, when the flit injection rate is higher than 34%, the NMR-DOR is also saturated.

With the hotspot traffic pattern, each node sends a packet to all its reachable destinations, except the hotspot node, $node_{27}$, with the same possibility. For the hotspot node, an extra 10% possibility is assumed. [20] also gets the worst performance, and the difference between itself and other two algorithms is notable. The hotspot node, $node_{27}$, has a 1-hop faulty neighbor, $node_{35}$, a 2-hops faulty neighbor, $node_{25}$, and three 3-hops faulty neighbors, $node_{12}$, $node_{21}$, and $node_{30}$. Surrounding by faulty neighbors together with the fact that it is 10% hotter than others, make the channels towards $node_{27}$ saturated. These saturated channels make it difficult to balance traffic. In this case, NMR-DOR could significantly reduce the packet latency by efficiently utilizing the turn-legally intermediate nodes.

To show the scalability of the proposed routing algorithm, we redo the simulations on an 16×16 mesh. In this simulation, 26 nodes are assumed as faulty. The fault set
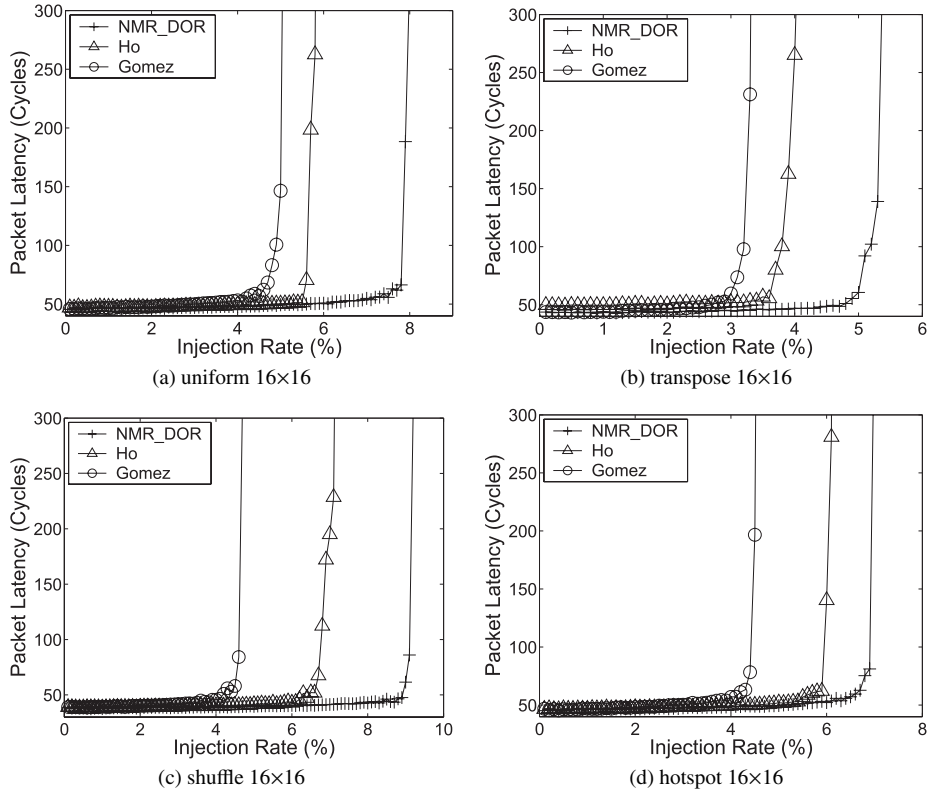
**Fig. 8** The average packet latency in 16×16 meshes. The proposed NMR-DOR is shown as a solid line with crosses, [17] (marked as Ho) is shown as a solid line with up-triangles, and [20] (Marked as Gomez) is shown as a solid line with circles. Under hotspot traffic, the hotspot node, $node_{27}$, has extra 10% possibility for receiving packets.

is {$node_{11}$, $node_{17}$, $node_{35}$, $node_{48}$, $node_{54}$, $node_{61}$, $node_{72}$, $node_{82}$, $node_{84}$, $node_{103}$, $node_{107}$, $node_{117}$, $node_{152}$, $node_{61}$, $node_{156}$, $node_{162}$, $node_{170}$, $node_{182}$, $node_{193}$, $node_{195}$, $node_{201}$, $node_{204}$, $node_{213}$, $node_{224}$, $node_{231}$, $node_{239}$, $node_{253}$}.

In large networks, congestion possibility increases due to the increased distance between source and destination nodes. In this case, NMR-DOR outperforms other two algorithms because it could use turn-legally intermediate nodes to balance traffic. For example, by comparing Fig. 7 (a) and Fig. 8 (a), we could find that the gap between NMR-DOR and other two algorithms is enlarged as the network size increases. We could find the same phenomenon under transpose and shuffle traffic patterns as shown in Fig. 8 (b) and Fig. 8 (c). Under hotspot traffic pattern, the absolute number of packets sent to hotspot dramatically increases due to the increased network size. Thus, the channels around the hotspot node are highly congested. The highly congested channels make it difficult to balance traffic even with turn-legally intermediate nodes.

## 7. Discussion

Currently, the NMR-DOR is implemented on 2D meshes, but it can be extended to other topologies if needed. The most challenge of this extension is to find turn-legally intermediate nodes in new topologies. In [5], the authors have

proposed the way to extend turn model to n-dimensional meshes, $k$-ary $n$-cubes, and hypercubes. Thus, extending NMR-DOR to these topologies is relative simple. As for other topologies, such as fat-tree, we should first extend the original turn model to them, thus we left it as the future work.

Multiple-round routing algorithms do need the global fault information, which was viewed as costly. In the NoC scenario, however, getting this kind of information is relatively convenient. For example, we could utilize the Build-In-Self-Test (BIST) technique to detect faults [32], [33], and scan out the fault information via the boundary scan chain, JTAG.

In fact, storing intermediate node information will introduce some area overhead to network interface. As stated in [17] and [20], this kind of overhead is acceptable for most applications. In our experiments, with the worst-case distribution of six faults, maximally 74 intermediate nodes are needed to be stored for each node. In this case, the router requires 74 bytes to store intermediate node information since each intermediate node needs one byte memory (2 bits for type and 6 bits for address). Furthermore, with the region-based technique [34], the number of maximal intermediate nodes could be reduced to be 12. Though the results are acceptable, we are trying to design a new compression technique to further reduce the number of stored intermediate nodes.

The proposed NMR-DOR classifies intermediate nodes to turn-legally and turn-illegally. Theoretically, we could utilize arbitrary number of turn-legally intermediate nodes inside each VC. However, according to our experiments (not shown in this paper), utilizing more turn-legally intermediate nodes inside a single VC gets dismissing results. Thus, we recommend the "turn-legally+normal+turn-legally" combination, such as the WF-NF configuration that gets good average results in above experiments. In application-specific scenarios, the designers should re-evaluate all configurations, and the best configuration maybe different.

In this paper, we focus on router failures. As for link failures, they could be treated by assuming one of two routers, connected by this faulty link, as faulty. As for core failures, they could be treated by assuming the router, connected with this core, as faulty, although they also could be handled by core-level strategies, such as [35], [36].

## 8. Conclusion

In this paper, we propose a new multiple-round dimension-order routing, namely NMR-DOR, that significantly improves the network performance by utilizing turn-legally intermediate nodes. The turn-legally intermediate nodes are those at which the turn made by the adjacent two dimension-order routing phases are allowed by an adopted turn model. Thus, utilizing them does not require adding extra VCs, making the routing algorithm more cost-efficient. Furthermore, given a number of VCs, the proposed NMR-DOR could tolerate more faults by utilizing more intermediate nodes. Thus, more computing capacity is saved, e.g., as for the unreachable node pairs in previous solutions, more than 50% of them are saved for networks without virtual channels and more than 90% are saved for networks with two virtual channels per physical channel. Finally, by evenly distributing packets that only utilize turn-legally intermediate nodes, the packet latency is dramatically reduced compared with existing multiple-round routing algorithms.
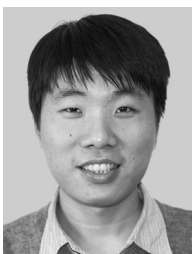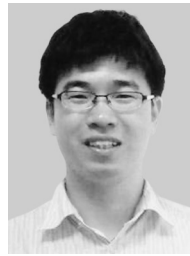
## Acknowledgement

### References

[1] W. Dally and B. Towles, Principles and Practices of Interconnection Networks, Morgan Kaufmann Publishers, 2004.

[2] S. Furber, "Living with failure: Lessons from nature?," IEEE European Test Symposium, pp.4–8, 2006.

[3] W. Dally and C. Seitz, "Deadlock-free message routing in multi-processor interconnection networks," IEEE Trans. Comput., vol.36, no.5, pp.547–553, 1987.

[4] J. Duato, "A new theory of deadlock-free adaptive routing in worm-hole networks," IEEE Trans. Parallel Distrib. Syst., vol.4, no.12, pp.1320–1331, 1993.

[5] C. Glass and L. Ni, "The turn model for adaptive routing," Proc. International Symposium on Computer Architecture, pp.278–287, 1992.

[6] G. Chiu, "The odd-even turn model for adaptive routing," IEEE Trans. Parallel Distrib. Syst., vol.11, no.7, pp.729–738, 2000.

[7] L. Peh and W. Dally, "A delay model and speculative architecture for pipelined routers," Proc. International Symposium on High-Performance Computer Architecture, pp.255–266, 2001.

[8] D. Linder and J. Harden, "An adaptive and fault tolerant wormhole routing strategy for k-ary n-cubes," IEEE Trans. Comput., vol.40, no.1, pp.2–12, 1991.

[9] A. Chien and J. Kim, "Planar adaptive routing: Low cost adaptive networks for multiprocessors," Proc. International Symposium on Computer Architecture, pp.268–277, 1992.

[10] C. Glass and L. Ni, "Fault-tolerant wormhole routing in meshes," Proc. International Symposium on Fault-Tolerant Computing, pp.240–249, 1993.

[11] R. Boppana and S. Chalasani, "Fault-tolerant routing with non-adaptive wormhole algorithms in mesh networks," Proc. Supercomputing, pp.693–702, 1994.

[12] S. Chalasani and R. Boppana, "Communication in multicomputers with nonconvex faults," IEEE Trans. Comput., vol.46, no.5, pp.616–622, May 1997.

[13] K. Chen and G. Chiu, "Fault-tolerant routing algorithm for meshes without using virtual channels," J. Information Science and Engineering, vol.14, pp.765–783, 1998.

[14] J. Wu, "A distributed formation of orthogonal convex polygons in mesh-connected multicomputer," Proc. International Parallel and Distributed Processing Symposium, pp.23–28, 2001.

[15] C. Chen and G. Chiu, "A fault-tolerant routing scheme for meshes with nonconvex faults," IEEE Trans. Parallel Distrib. Syst., vol.12, no.5, pp.467–475, 2001.

[16] J. Wu, "A fault-tolerant and deadlock-free routing protocol in 2D meshes based on odd-even turn model," IEEE Trans. Comput., vol.52, no.9, pp.1154–1169, 2003.

[17] C. Ho and L. Stockmeyer, "A new approach to fault-tolerant wormhole routing for mesh-connected parallel computers," IEEE Trans. Comput., vol.53, no.4, pp.427–438, 2004.

[18] V. Puente, J. Gregorio, F. Vallejo, and R. Beivide, "Immunet: A cheap and robust fault-tolerant packet routing mechanism," Proc. International Symposium on Computer Architecture, 2004.

[19] D. Xiang, J. Sun, J. Wu, and K. Thulasiraman, "Fault-tolerant routing in meshes/tori using planarly constructed fault blocks," Proc. International Conference on Parallel Processing, pp.577–584, 2005.

[20] M. Gómez, N. Nordbotten, J. Flich, P. López, A. Robles, J. Duato, T. Skeie, and O. Lysne, "A routing methodology for achieving fault tolerance in direct networks," IEEE Trans. Comput., vol.55, no.4, pp.400–415, April 2006.

[21] A. Mejia, J. Flich, J. Duato, S. Reinemo, and T. Skeie, "Segment-based routing: An efficient fault-tolerant routing algorithm for meshes and tori," Proc. International Symposium on Parallel and Distributed Processing, 2006.

[22] R. Holsmark and S. Kumar, "Corrections to Chen and Chiu's fault tolerant routing algorithm for mesh networks," J. Information Science and Engineering, vol.23, no.6, pp.1649–1662, 2007.

[23] Z. Zhang, A. Greiner, and S. Taktak, "A reconfigurable routing algorithm for a fault-tolerant 2D-mesh network-on-chip," Proc. Design Automation Conference, pp.441–446, 2008.

[24] D. Fick, A. DeOrio, G. Chen, V. Bertacco, D. Sylvester, and D. Blaauw, "A highly resilient routing algorithm for fault-tolerant NoCs," Proc. International Conference on Design, Automation and Test in Europe, pp.21–26, 2009.

[25] D. Fick, A. DeOrio, J. Hu, V. Bertacco, D. Blaauw, and D. Sylvester, "Vicis: A reliable network for unreliable silicon," Proc. Design Automation Conference, pp.812–817, 2009.

[26] B. Fu, Y. Han, H. Li, and X. Li, "A new multiple-round DOR routing

for 2D network-on-chip meshes," Proc. IEEE Pacific Rim International Symposium on Dependable Computing, pp.276–281, 2009.

[27] Y. Fukushima, M. Fukushi, and S. Horiguchi, "Fault-tolerant routing algorithm for network on chip without virtual channels," Proc. International Symposium on Defect and Fault Tolerance in VLSI Systems, pp.313–321, 2009.

[28] J. Flich, S. Rodrigo, and J. Duato, "An efficient implementation of distributed routing algorithms for NoCs," Proc. International Symposium on Networks-on-Chip, pp.87–96, 2008.

[29] S. Rodrigo, J. Flich, A. Roca, S. Medardoni, D. Bertozzi, J. Camacho, F. Silla, and J. Duato, "Addressing manufacturing challenges with cost-efficient fault tolerant routing. Networks-on-chip," Proc. International Symposium on Networks-on-Chips, pp.25–32, 2010.

[30] P. Gratz, B. Grot, and S. Keckler, "Regional congestion awareness for load balance in networks-on-chip," Proc. IEEE 14th International Symposium on High Performance Computer Architecture, pp.203–214, 2008.

[31] BookSim Simulator. http://nocs.stanford.edu/cgi-bin/trac.cgi/wiki/Resources/BookSim.

[32] X. Li and P. Cheung, "A loop-based apparatus for at-speed self-testing," J. Computer Science and Technology, vol.16, no.3, pp.278–285, May 2001.

[33] Y. Han, Y. Hu, X. Li, H. Li, and A. Chandra, "Embedded test decompressor to reduce the required channels and vector memory of tester for complex processor circuit," IEEE Trans. Very Large Scale Integr. Syst., vol.15, no.5, pp.531–540, May 2007.

[34] J. Flich, A. Mejia, P. Lopez, and J. Duato, "Region-based routing: An efficient routing mechanism to tackle unreliable hardware in network on chips," Proc. International Symposium on Networks-on-Chip, pp.183–194, 2007.

[35] L. Zhang, Y. Han, Q. Xu, and X. Li, "Defect tolerance in homogeneous manycore processors using core-level redundancy with unified topology," IEEE/ACM Design, Automation and Test in Europe, pp.891–896, 2008.

[36] L. Zhang, Y. Han, and H. Li, "A fault tolerance mechanism in chip many-core processors," J. Tsinghua Science and Technology, vol.12, no.s1, pp.169–174, 2007.

**Yinhe Han** received the B.Eng. degree from Nanjing University of Aeronautics and Astronautics, China, in 2001, and the M.Eng. and Ph.D. degree in computer science from the Institute of Computing Technology, Chinese Academy of Sciences, China, in 2003 and 2006, respectively. He is currently an Associate Professor at the Institute of Computing Technology, Chinese Academy of Sciences. His research interests include VLSI design and testing, on-chip interconnection and fault-tolerant. Dr. Han was a recipient of the Test Technology Technical Council Best Paper Award at the Asian Test Symposium 2003. He is a member of IEEE/ ACM /CCF. He was Program Co-Chair of Workshop of RTL and High Level Testing (WRTLT) in 2009, and serves on the Technical Program Committees of several IEEE and ACM conferences, including ATS, GVLSI etc.

**Huawei Li** received her B.S. degree in computer science from Xiangtan University, Hunan, China, in 1996, and the M.S. and Ph.D. degrees from the Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China, in 1999 and 2001, respectively. She is currently a Professor at the Institute of Computing Technology, Chinese Academy of Sciences. Her research interests include VLSI/SOC design verification, test generation, delay test and design for reliability. She was Program Co-Chair of IEEE Asian Test Symposium (ATS) in 2007, and Program Chair of Workshop of RTL and High Level Testing (WRTLT) in 2003. She is a senior member of IEEE.

**Xiaowei Li** received his B.Eng. and M.Eng. degrees in Computer Science from Hefei University of Technology (China) in 1985 and 1988 respectively, and his Ph.D. degree in Computer Science from the Institute of Computing Technology (ICT), Chinese Academy of Sciences (CAS) in 1991. Currently, he is a professor and deputy director of the Key Laboratory of Computer System and Architecture, ICT, CAS. His research interests include VLSI testing and design verification, dependable computing, wireless sensor networks. He is a senior member of IEEE. He is an associate Editor-in-Chief of the Journal of Computer Science and Technology (JCST), and a member of Editorial Board of Journal of Electronic Testing (JETTA) and Journal of Low Power Electronics (JOLPE). In addition, he serves on the Technical Program Committees of several IEEE and ACM conferences, including VTS, DATE, ASP-DAC, PRDC, etc. He was Program Co-Chair of IEEE Asian Test Symposium (ATS) in 2003, and General Co-Chair of ATS 2007.

**Binzhang Fu** received his B.Eng. degree in both Electronics and Information Engineering, and Computer Science and Technology from Huazhong University of Science and Technology in 2004. Mr. Fu now is a Ph.D. candidate in computer science in Institute of Computing Technology, Chinese Academy of Science. His research interests include Fault-tolerant Routing, DfT in NoC, and NoC virtualization.