A Quantitative Study on NoC Traffic Scenarios

Ming-hua TANG¹ and Jing LIN²,*

¹School of Internet Finance and Information Engineering, Guang Dong University of Finance, China
²School of Information Management, Shanghai Lixin University of Accounting and Finance, China
*Corresponding author

Keywords: NoC, Routing algorithm, Traffic scenario.

Abstract. In the research of Network-on-Chip (NoC), after a new routing algorithm is devised, its performance should be evaluated by a lot of simulations. In these simulations, a large number of traffic scenarios, such as uniform random, transpose, tornado, etc. have been taken into consideration. However, there is lack of comparative study of these traffic scenarios in the literature. In this paper, the routing pressures of ten routing algorithms are compared under 1,000,000 traffic scenarios. Then simulations are carried out under more than 100 traffic scenarios. The computation and simulation results show that if a routing has small routing pressure and good performance under both transpose1 and transpose2 traffics, then poor performance will not be achieved under any other traffic scenario, with high probability.

Introduction

With the rapid improvement of semiconductor technology, a huge number of processing cores could be integrated into a single chip. For those complex systems, the communication interconnects are the key to improve the system performance. Toward this end, Network-on-Chip has been proposed [1] in order to handle the SoC communication requirements.

For the mostly preferred 2D mesh topology, a large number of routing algorithms have been proposed. A part of turns have to be prohibited in order to avoid deadlock if no VCs are used. Consequently, the routing algorithms are referenced as the turn model based routings. Examples of turn model based routing algorithms include Dimension Order Routing (DOR) [2], turn model [3], Odd-Even (OE) turn model [4], RTM [5], etc. Apart from these well-known routing algorithms, other routing algorithms could be constructed by taking advantage of Divide-Conquer approach [7].

The performance of routing algorithms is usually evaluated through simulation. A number of traffic scenarios have to be considered in order to compare the performance of different routing algorithms. The always considered synthetic traffic scenarios include random, transpose1, transpose2, tornado, bitcomp, bitreversal, asymmetric, hot-spot, etc. Multimedia system [8] is the considered realistic traffic.

Although a huge number of traffic scenarios have been considered, the considered traffic scenarios are only a tiny part of the possible traffic scenarios. When a routing algorithm is asserted to have good performance under the well-known traffic scenarios after simulations, it still cannot be assured that it will not have poor performance under other undetected traffic scenarios.

Although simulation is the mostly accepted method to evaluate routing algorithms, a lot of time is needed. When thousands of routing algorithms are needed to be evaluated, the time needed by the simulations is not acceptable. Under this circumstance, a high efficiency routing metric could greatly improve the evaluation speed. Toward this end, routing pressure is proposed to measure the routing algorithms [6].

In this paper, ten groups of routing algorithms are firstly constructed with different routing pressure under transpose1 and transpose2 traffic scenarios. Then the routing pressure of the ten routings under 1,000,000 randomly traffics are computed. Furthermore, simulations are carried out under one
hundred randomly generated traffic scenarios. Computation and simulation results show that if a routing has better performance under both transpose1 and transpose2 traffics then it will not have poor performance under a large number of traffics, with high probability.

Related Work
Avoiding deadlock is one of most important tasks of designing routing algorithms. In 2D mesh topology, if no VCs are used, a proper group of 90 degree turns needs to be prohibited in order to avoid deadlock. The number and location of the prohibited turns which have significant impact on system performance are among the main research topics of routing algorithms.

In Dimension Order Routing (DOR) [2], half of the allowed turns are prohibited, which makes that there is only one deterministic path for each communication pair. It is pointed out that prohibiting a quarter of the allowed turns is sufficient to avoid deadlock [3]. Later, it is discovered that prohibiting different turns at odd and even columns could improve the routing performance [4]. The distributions of the prohibited turns are systematically studied in research [5]. After all possibilities are detected, the best one is proposed as RTM routing. With Divide-Conquer approach [7], all kinds of routing algorithms could be constructed. Furthermore, all routing algorithms for 2D mesh networks that are smaller than 5×5 could be constructed. Routing pressure is proposed as an efficient routing algorithm performance metric [6].

Traffic Analysis
In this paper we focus on 2D mesh topology which is abstracted as a 2D coordinate system. For an n×n 2D mesh network, the number of nodes is n². The traffic scenarios for n×n 2D mesh network is defined as following:

Traffic matrix(Λ): Any doubly-stochastic matrix has row and column sums of exactly one, where entry λ_i,j represents the fraction of traffic traveling from node i to node j.

Permutation matrix(Γ): A traffic matrix whose entries are either 0 or 1.

For an n×n 2D mesh network, its permutation matrix is n²×n² matrix. The total number of permutation matrix is n²!. For example, for 5×5 2D mesh network, its total number of permutation matrix is 25!.

According to [9], any doubly-stochastic traffic matrix Λ could be expressed as a weighted combination of a group of permutation matrices. Consequently, we only consider permutation matrix in this paper.

In 2D mesh NoC, there are a number of well-known and widely referenced traffic scenarios, such as random, transpose, tornado, bitcomp, bitreversal, asymmetric, hot-spot, etc. These considered traffic scenarios only occupy a tiny part of the permutation matrices. When a routing algorithm is asserted to have good performance under the well-known traffic scenarios after simulations, it still cannot be assured that it will not have poor performance under other undetected traffic scenarios.

In 2D mesh NoC, some traffic scenarios, such as butterfly, can be applied only on particular network sizes (power of 2). While for certain traffic, for example uniform, a routing algorithm that has good performance under this traffic may have very poor performance under other traffics. For example, XY routing has the best performance under uniform traffic. However, XY routing has very poor performance under other traffic scenarios.

In transpose1 traffic, source node (i, j) generates packets to destination node (N-1-j, N-1-i). In transpose2_positive traffic, source node (i, j) only generates packets to node (j, i). However, for transpose1 and transpose2 traffics, they contain the southeast, northwest, northeast and southwest communications. Furthermore, for most of the real traffics, packets travel in those four directions. Consequently, if a routing works well for transpose1 and transpose2 traffics, it may not work poorly under any other traffic. Then we can expect good performance routings under lots of traffics.
We take a communication pair (1, 34) as an example. Although it is not a communication pair in transpose1 traffic, it will not have poor performance if the routing has good performance under transpose1 traffic. Most of the real traffics are composed of a large number of such communication pairs. Although they are not communication pairs of transpose1 or transpose2 traffics, they will not have poor performance if the routing has good performance under both transpose1 and transpose2 traffics with high probability.

Using Divide-Conquer method, 50 (r1-r50) routing algorithms are constructed. The 50 routing algorithms are classified into ten groups, five routings in each group. The five routing algorithms of each group have the same routing pressure under transpose1 traffic scenario, and have the same routing pressure under transpose2 traffic, as shown in Table 1. For the first eight groups, each routing algorithm has the same routing pressure under both transpose1 and transpose2 traffics. For the last two groups, each routing has different routing pressure under transpose1 and transpose2 traffics. In the remaining of the paper, no matter computation of simulation, the average results of one group will be reported. Consequently, routing algorithms R1-R10 are used to represent the ten groups, respectively.

Table 1. Routing pressure of 50 routing algorithms.

<table>
<thead>
<tr>
<th></th>
<th>r1-r5</th>
<th>r6-r10</th>
<th>r11-r15</th>
<th>r16-r20</th>
<th>r21-r25</th>
<th>r26-r30</th>
<th>r31-r35</th>
<th>r36-r40</th>
<th>r41-r45</th>
<th>r46-r50</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transpose1</td>
<td>2.5</td>
<td>3</td>
<td>3.5</td>
<td>4</td>
<td>4.5</td>
<td>5</td>
<td>5.5</td>
<td>6</td>
<td>2.5</td>
<td>6</td>
</tr>
<tr>
<td>Transpose2</td>
<td>2.5</td>
<td>3</td>
<td>3.5</td>
<td>4</td>
<td>4.5</td>
<td>5</td>
<td>5.5</td>
<td>6</td>
<td>6</td>
<td>2.5</td>
</tr>
</tbody>
</table>

In order to evaluate the constructed routing algorithms, 1,000,000 permutation matrices are randomly constructed. The routing pressures of every routing algorithm are computed. The average of each group is shown in Figure 1. As the figure shows that group R1 has the smallest routing pressure under both transpose1 and transpose2 traffics, it still has the smallest routing pressure under the randomly constructed traffics. From R1 to R8, their routing pressures under both transpose1 and transpose2 traffics increase, and their routing pressures under the randomly constructed traffics also increase. R9 only has small routing pressure under transpose1 traffic and R10 only has small routing pressure under transpose2 traffic, they could not have small routing pressure under the randomly constructed traffics.

Simulations are conducted under 100 randomly created permutation matrices. For each permutation matrix, the average packet latencies of the ten groups are firstly normalized to R1. Then, the average is calculated over the 100 cases. Results show that R1 still has the smallest value. From R1 to R8, routing performance degrades gradually. R9 and R10 still could not have better performance than R1.

For every group Ri (i:2-10), the number of traffics where Ri has smaller average latency than that of R1 is counted. R2 has smaller latency than that of R1 in 39 out of the 100 traffics, which is largest. R8 has smaller latency than R1 in 3 cases which is the smallest.
The simulator of this paper is Noxim [10] which supports a wide range of synthetic traffic scenarios, such as Uniform, Transpose, bitcomp, bitrev, diagonal, asymmetric, tornado, etc.

The packet latency variations under transpose1 and transpose2 traffic scenarios are depicted in Figure 2 and Figure 3, respectively. The two figures show that the performances of R1, R2, R3, R4, R5, R6, R7 and R8 degrade gradually. Routing R1 has the best performance and Routing R8 has the worst performance. Under transpose1 traffic, routing R9 has the same good performance as that of R1 since R9 has the same small routing pressure. However, routing R10 has the same poor performance as that of R8 since they have the same routing pressure.

![Figure 2](image1.png)

**Figure 2.** Latency variations under transpose1 traffic scenario.

![Figure 3](image2.png)

**Figure 3.** Latency variations under transpose2 traffic scenario.

Under transpose1 traffic, routing R10 has the same good performance as that of R1 since R9 has the same small routing pressure. However, routing R9 has the same poor performance as that of R8 since they have the same routing pressure.

Figure 4 shows the results for asymmetric traffic scenario. R1 still has the better performance than six routing algorithms. Only two routing algorithms have better performance than R1.
Routing algorithm R1 has the best performance under both transpose1 and transpose2 traffic scenarios among the six routing algorithms. It still has the best performance under hs-c, hs-tr, and random traffics scenarios. Under the other traffics, although it does not have the best performance, it does not have the worst performance either. When 100 random traffic scenarios are considered, routing R1 still has the best performance. Consequently, from the simulations in this section, one important observation could be made that if a routing has good performance under both transpose1 and transpose2 traffics, it will not have poor performance under any other traffic scenario, with high probability.

Conclusions

In this paper, ten groups of routing algorithms are firstly created. Then the routing pressures of these routing algorithms under 1,000,000 randomly constructed permutation matrices are calculated. Finally, a wide range of simulations are conducted to compare the performances of the ten groups of routing algorithms under a large number of traffic scenarios. The considered traffic scenarios include a set of well-known traffic scenarios and 100 randomly generated traffic scenarios. Simulation results show that if a routing achieves good performance under both transpose1 and transpose2 traffics, it will not have poor performance under a large number of traffic scenarios, with high probability.

Acknowledgement

This work is supported in part by the Natural Science Foundation of Guangdong under Grants No. 2017A030313391.
References


