C-009

A Novel Routing Algorithm based on Path Diversity and Congestion Estimation

Yang Hong， Lian Zeng， Xin Jiang and Takahiro Watanabe
（Graduate School of Information, Production and Systems, Waseda University）

1 Abstract

This paper proposes a minimal adaptive routing algorithm for Network-on-Chip, which takes congestion information and routing diversity into consideration. Congestion is one of the most important factors on performance. Our algorithm can select a lower latency path for packet transmission according to the following conditions: the free buffer size of two neighbor routers is compared, the direction which has more different paths to the destination is chosen, it decides which direction the packet more tend to be transmitted by the position of the source and destination. As the result, the algorithm gets the proper direction for the next step. Comparing to other algorithms, our proposed routing algorithm has less latency and better throughput.

Keywords: NoC,routing algorithm, congestion, path diversity

2 Introduction

Recently, a digital electronic system cannot satisfy the needs of communication due to control logic and not arithmetic. Moreover, the bus architecture has restrictions limiting performance, cost, power and size. The design that takes too much time to dedicated architectures would be impossible.

Currently, VLSI technology allows the construction of multi-core on a chip. So in order to achieve a multiprocessor system, Network-on-Chip(NoC) technology which implements several hundreds or even thousands of resources rises up.

For the selection of NoC's topology, the mesh topology has becomes one of the most popular architecture, to achieve communication between nodes in the mesh by sending messages through the network. As the number of nodes and the complexity of processors are increasing, more and more routing algorithms are come up with to solve the problem about routing packets in the network. The routing algorithm mainly divides into deterministic and adaptive approaches. The routing path in a deterministic algorithm is fixed from each pair of source and destination. Obviously, this algorithm results in increased packet latency. In contrast, for the adaptive algorithm, a packet chooses a path based on the state of the network, which means it uses variable routing path. The traditional methods such as Dimension Order Routing algorithm routes packets across the dimensions with increasing or reducing order in mesh. In this paper, we propose a hierarchical routing algorithm that has better performance than XY algorithm and turn model algorithm [1]. A selection policy we construct for this algorithm will be introduced and analyzed.

3 Proposed Method

Our proposed method is based on full adaptive routing algorithm. A packet is sent to the X or Y direction by comparing the congestion in the network. The current node only uses the local congestion (i.e. the current node knows the free input buffer size of four nodes surrounded). The packet can avoid the state that they should spend time to wait for occupied channel by using the congestion information, which reduces the global average delay and increases the throughput in the network.

We define a variable $C_{[r][d]}$ named *congestion degree* to describe the buffer status as stated in (1). It shows how many free slots can be used. In (1), *r* and *d* depict current node number and direction including the four cardinal points. *Buffer_Size* is a fixed variable and it is preset.

$$C_{[r][d]} = \frac{Free\_Slots\_Size_{[r][d]}}{Buffer\_Size} \quad (1)$$

In our algorithm, path diversity is another parameter to control packet routing. Route a packet to one path and get more optional paths from current node to destination for next step. Path diversity can reduce the possibility that the packet is blocked. To describe path diversity, we define a variable $D_{[r][d]}$ named *diversity degree*. In (2) and (3), after calculate we find that the value of diversity degree is equal to the distance between current node and destination in one direction.

$$Distance\_DC_x = |Current_x - Destination_x| \quad (2)$$

$$Distance\_DC_y = |Current_y - Destination_y| \quad (3)$$

In (4), we use *Distance_DC* to depict diversity degree.

$$D_{[r][d]} = Distance\_DC_{x(y)} \quad (4)$$

The fairness of network is an aspect we take into consideration. For most of the common mesh architecture, the length in X and Y direction is same. So we consider when the pairs of source and destination is randomly distributed in the network, for sending all packets from the source to destination, the average time we spend in X and Y direction is nearly same. However, when the length in X is not equal to Y direction, the time we use to transmit the whole packets in one direction is greater than another resulting in the unfairness of the mesh. We add a parameter to let the time in X and Y direction is more balance.

We define a parameter $F_{[r][d]}$named *fair degree*. In (5), $Mesh\_Size_{one\_direction}$ indicates the length of X or Y of the mesh. The denominator means the total length of the mesh.

$$F_{[r][d]} = \frac{Mesh\_Size_{one\_direction}}{Mesh\_Size_X + Mesh\_Size_Y} \quad (5)$$

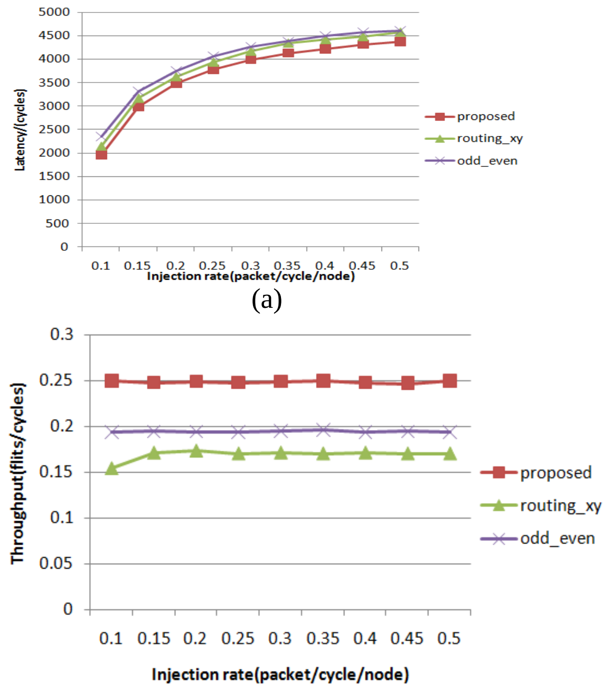
From the current node to destination, we can calculate three values in each direction. In (6), we use these values and add α, β, γ as weighted parameter to obtain *path selection parameter* which is used to decide packet routing path for next step.

$$V_{[r][d]} = \alpha C_{[r][d]} \times \beta D_{[r][d]} \times \gamma F_{[r][d]} \quad (6)$$

Finally, every time we only need to judge two directions for the packet from source to destination. After we calculate the path selection parameter in these two directions, compare the value and choose the greater one as the direction for next step.

4 Simulation Result

To demonstrate the network performance of our method, we simulate our algorithm by NOXIM simulator [2], where mesh size is 8×8 and $\alpha, \beta, \gamma = 1$. The buffer depth is 4. The min packet size is 2 flits and the max packet size is 10 flits. The traffic pattern is transpose. We compare our method with two traditional method XY routing algorithm and odd-even algorithm.



(a)

(b)

Figure 1. a) Latency comparison of proposed method, XY routing and odd-even routing. b) Throughput comparison of proposed method, XY routing and odd-even routing.

As shown in Figure 1, our proposed method has higher throughput and low latency than both odd-even and XY routing algorithm under an equal condition. The main reason is that our proposed method considers the situation of network and finds an optimal path for packet. For traditional algorithms, they can't balance the network and have great possibilities to cause block. However, we generate an evaluation mechanism used to reduce congestion occurs.

5 Conclusion

In this paper, we introduce a high performance routing method. It is based on the adaptive algorithm. The proposed method requires the information of the network. At first we consider the congestion of the node. In addition, we calculate path diversity from current node to destination. Moreover, we add constant to balance the network. Our method uses these three parameters to construct Evaluation Mechanism. This method can achieve better performance than traditional algorithms.

References

[1] C. J. Glass and L. M. Ni, "The turn model for adaptive routing," 19th Annual Int. Symp. On Computer Architecture (ISCA), pp. 278-287,1992.

[2] F. Fazzino, M. Palesi, and D. Patti, Noxim: Network-on-Chip Simulator, http://sourceforge.net/ projects /noxim, 2009.