TELKOMNIKA, Vol. 11, No. 12, December 2013, pp. 7392~7398
e-ISSN: 2087-278X

Congestion Prediction Algorithm for Network on Chip

Hua Cai[1], Yong Yang*[2], FuHeng Qu[2], JianFei Wu[2], Bo Wang[2]
[1]College of Electronic and Information Engineer, Chang Chun University of Science and Technology, ChangChun 130022, Ji Lin, China
[2]College of Computer Engineer, Chang Chun University of Science and Technology, ChangChun 130022, Jilin, China
*Corresponding author, e-mail: cc.caihua@hotmail.com

Abstract

Network on chip (NoC) traffic congestion is one of the important reasons for the data transmission performance degradation. In this paper, we present a congestion judgment algorithm, which is based on neural network. The congestion control algorithm firstly uses the hamming network to compute the NoC's link buffer congestion state, secondly uses the competitive network to find the worst congestion node, and then adopts avoiding congested node routing policy to improve the NoC's transmission performance. In this paper, the congestion control algorithm can make the data stream as far as possible evenly distributed in the NoC's nodes and links and reduce the transmission resource competition. The simulation results show that the congestion control algorithm can achieve better network throughput and average transmission delay.

Keywords*: network on chip, congestion control, neural network*

1. Introduction

With the development of semiconductor technology, more and more functions were integrated to one single chip and formed namely System-on-Chip (SoC). Now System-on-Chip development was confronted by severe challenges, such as managing deep submicron effects, scaling communication architectures and bridging the productivity gap. Network-on-Chip (NoC) has been a rapidly developed concept in recent years to tackle the crisis with focus on network-based communication [1]. NoC proposes networks as scalable, reusable and global communication architecture to overcome the pains of future System-on-Chip. In a NoC, several cores have been integrated into one single chip, which have different function such as processors, memories and other logical component. These cores were interconnected by switch and communicated information by routing packets. So the topology, routing algorithms and flow control were the key technology in NoC [2].

In this paper, we focused on communication efficiency of the NoC and aimed to improve the performance of NoC's communication. Network congestion was one of the important reasons for the data transmission performance degradation. We proposed a congestion aware algorithm based on neural network, which can provide global congestion state. Relative to local congestion information, it could adjust the allocation of network resources better and enable the data stream as far as possible evenly distributed in the network nodes and links. So it could get better network throughput and average transfer delay by reducing the congestion.Throuth the simulation, when in the medium saturated network environment, the congestion-aware method could improve the perfoamance 15%.

The rest of the paper was organized as follows: the related work is firstly summarized in the next Section. In Section 3, the proposed congestion aware algorithm method is presented. In Section 4, the performance evaluation are detailed. Finally, we concluded our paper in the last Section.

2. Related Work

2.1. Network Topology

The definition of the NoC's topology was the arrangement of the link and node. The common topologies was shown in Figure 1.

Received June 23, 2013; Revised August 17, 2013; Accepted August 30, 2013

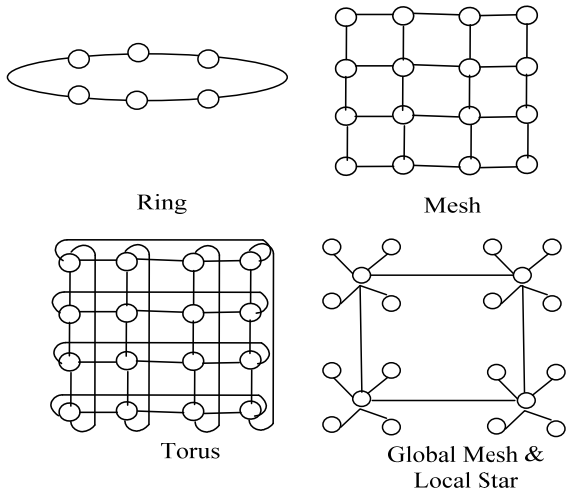

Figure 1. The Topology of the Network on Chip

In the topology, the number of connections on each node with the links directly impacted on the valid path number of network-on-chip. When the topology has abundant valid paths, under the same injection rate, it can obtain a shorter time delay. On the contrary, the networks are prone to congestion and resulted in network saturation.

However, in the design of the network topology, except considering the network performance, also needed to consider the actual manufacturing cost, especially chip interconnect density, the length of the line, power consumption and area etc. So, in a two-dimensional structure, mesh and torus structure was the commonly used topology. Moreover, three-dimensional topology was also studied [3].

2.2. Routing Algorithm

The routing algorithm was to find a data flow path from the source node to the destination node through the specific topology [10]. The routing algorithm was one of the important reasons for the NoC performance. So in the design of the routing algorithm, it was necessary to try to make the data streams uniformly distribution in the links or the nodes of the NoC to achieve the maximum throughput of the network design and to avoid the data streams to concentrate in one node or one link resulting in congestion. The common used routing methods were X-Y routing, obvious routing and adaptive routing [4]. In the mesh topology, the three routing algorithm was shows in Figure 2. The figure shown from node A to node B, in which the black solid node represents a congested node.

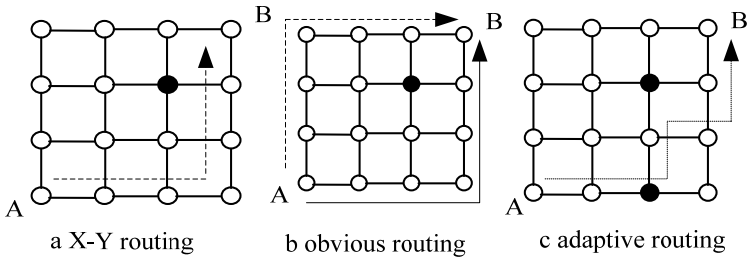

Figure 2. Network-on-chip Routing Algorithm

X-Y routing algorithm was a simple routing strategy, in which the data stream was first along one-dimensional direction (the X dimension or the Y dimension) from the source node to reach the destination node where this dimension endpoint, and then proceeded along the

direction of the other dimension. In obvious routing algorithm, the routing path between the source node and the destination node was determined and randomly selected one routing path when every routing time.

X-Y routing algorithm and obvious routing algorithm were the simple routing algorithm, which didn't consider the current state of the network, even if the nodes in the routing path already congested, it also followed the routing path accordance with the established routing strategy. Adaptive routing was a complex routing algorithm, which based on the current network congestion state and according to the routing rules to select the optimal routing path. But the adaptive routing algorithm was only based on the current node information to determine routing policy, which may cause multiple data streams to constitute a ring cycle, namely deadlock or livelock. Therefore, to obtain the global congestion information and congestion-aware routing algorithm became the research focus. So in this paper, we presented a congestion control algorithm, which based on the global congestion information.

3.1. Flow Control

The flow control was defined as how to allocate bandwidth and the buffer etc network resources during data packets transmission in the NoC. The flow control process is a node was shown in Figure 3.

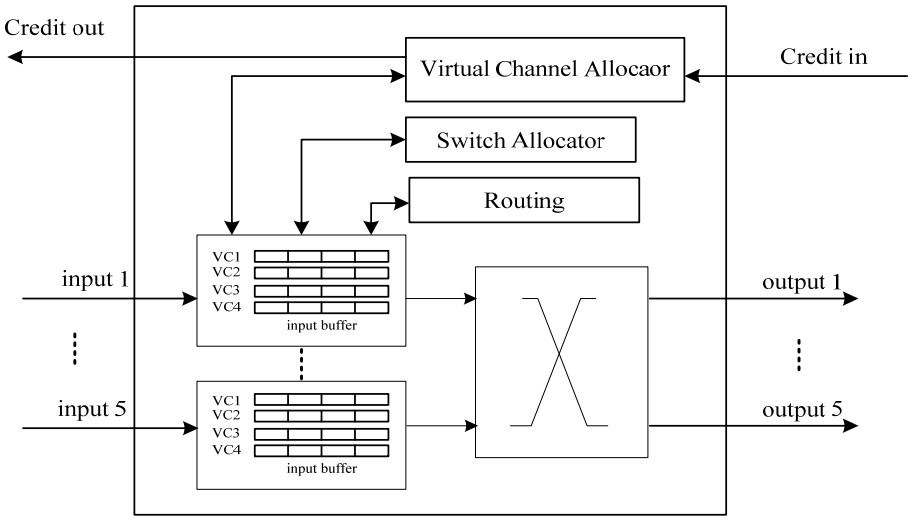

Figure 3. The Diagram of Flow Control in One Node

In network-on-chip, on-chip resources were the buffer, link and status register. Flow control strategies aimed to efficient allocate these resources and achieve the maximum throughput and the smallest packet transmission delay.

According to the usage of the buffer, the simplest flow control strategy was non-cache mechanism. Because it could not temporarily store the data packets, so it only handled one packet one time. The excess packets lost or used of non-direct route processing. In the congested node, the data packet header would wait until the resources released. The process will continue until the complete communication path is establishment.

Circuit switching was a simple storage flow control, in which just storing data packet header information and through the data packet header information to establish a communication path.

The most efficient flow was storage-forwarding control mechanism, which based on the buffer reused. Using the buffer to avoid data packets coupling, especially use the wormhole switching and virtual channel technology, can improve network performance.

3. Congestion Control Algorithms based on Neural Networks

Many key technologies in Network-on-chip aimed to pursuit of maximum network throughput and minimum packet transmission delay. However, when network congestion occurred, it would directly affect the network performance. Therefore, in congestion control, we would comprehensive consider the network topology, routing algorithms and flow control strategies [6].

3.1. The Existing Congestion Control Algorithm Analysis

In paper [5], proposed an approximate congestion-aware technology to control congestion. This method was aware the neighboring nodes' pressure information and then made a judgment of the node routing and flow control. However, this method was lack of global information and could not fully utilize the entire network on chip resources. In paper [7], proposed a buffer control and bandwidth allocation to control congestion, although this method divided different levels in the fault-tolerant control and obtained better results, but it also leaded to area and power trade-offs, which needed more complex logic control units.

For an efficient congestion control mechanism, it was able to according to the global network state to make the data flow evenly distributed between the nodes and links and to avoid the emergence of the "hot spots" and their regional. In this paper, based on the network-on-chip key technologies and neural network parallel processing information characteristics, we proposed to utility the neural network to perceive the distribution status of data packets on the NoC, then adjusted routing and flow control strategy based on these information to reduce congestion occurrence probability and achieved the maximum throughput of the NoC.

3.2. Congestion Prediction Algorithms based on Neural Networks Analysis

Neural network provided the parallel processing information mechanism, which was constituted by the neurons and the interconnection between the neurons. The neurons were the computing nodes and the interconnection completed the connection between the computing nodes [8].

In this paper, we used two-step to find the congested nodes. The first part was judgment the dimension of congested node, which used multi-sensing layer neural network structure.

The first layer was hamming network to compute the usage of the every dimension's resource and the second layer was competitive network, which to find the biggest one-dimensional from the hamming network. The second step was to find the congested node in the determined dimension. Congestion control is shown in Figure 4.

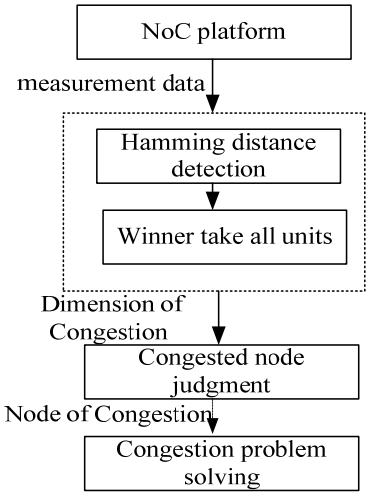

Figure 4. The Diagram of Congestion Aware Control

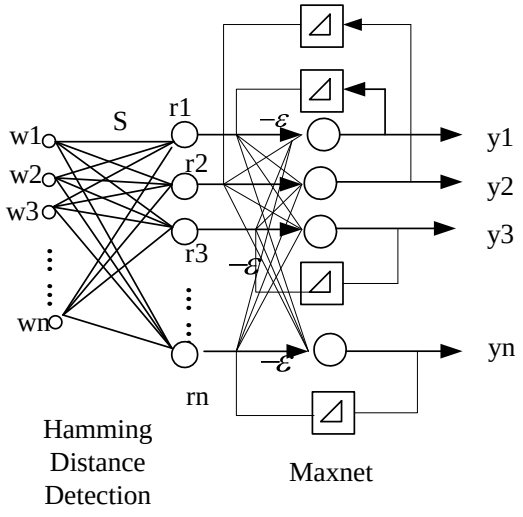

Figure 5. Congestion-aware Neural Network

Congestion-aware network structure is shown in Figure 5. The input of the neural network was the congestion status of a dimension in network on chip. Weight matrix S was the status value of the previous time for the input congestion state, the initial state was 0. The hamming network calculated the hamming distance between the input and the each dimension of the weight matrix S, which achieves the intermediate nodes.

In hamming distance detection network, the weight matrix was

$$W = \frac{1}{2}\begin{bmatrix} s_1^1 & s_2^1 & \cdots & s_n^1 \\ s_1^2 & s_2^2 & \cdots & s_n^2 \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ s_1^p & s_2^p & \cdots & s_n^p \end{bmatrix} \quad (1)$$

The distance of input and weight matrix was:

$$d_h(x, s^m) = \frac{n}{2} - w^T s^m \quad (2)$$

When set the Threshold value $n/2$, according to (2), defined:

$$net_m = \frac{1}{2} w^T s^m + \frac{n}{2} = n - d_h(x, s^m) \quad (3)$$

We adopted the unit activation function as $f(net_m) = \frac{1}{n} net_m$, when $d_h(x, s^m) = 0$, $f(net_m)$ achieve 1.

Behind hamming network was the competition network, which was the Maxnet except feedback weights is 1 others feedback weights is $-\varepsilon$. Maxnet weight matrix is:

$$W_M = \begin{bmatrix} 1 & -\varepsilon & \cdots & -\varepsilon \\ -\varepsilon & 1 & & -\varepsilon \\ \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots \\ -\varepsilon & -\varepsilon & & 1 \end{bmatrix} \quad (4)$$

Maxnet was a dynamic network, which output was:

$$y^{k+1} = \Gamma(W_M y^k) \quad (5)$$

In which Γ was a nonlinear diagonal operator and the element is:

$$f(net) = \begin{cases} 0, & net < 0 \\ net, & net \geq 0 \end{cases} \quad (6)$$

After the Maxnet operator, we could get a maximum output of intermediate nodes [9], which meant we could attach the most congestion dimension. Then, in this dimension, in turn to find the most depleted resource node, you could get most likely congested node.

4. Simulation and Results

In the simulation analysis, we adopted system to model and build a 8×8 *Mesh NoC* model. In the NoC model, we used uniform distribution data flow model, which was each node to send data packets with uniform probability. Routing algorithm used the improved X-Y routing, when congestion node in the same direction with the routing direction in the first time, packets temporary switch to another dimension once, avoid congested nodes; when the congested node routing direction to the second forward-dimensional with the routing algorithm so that the data stream was temporarily stored in the host node, waited until the congestion was released. NoC detailed structure was depicted in Figure 6.

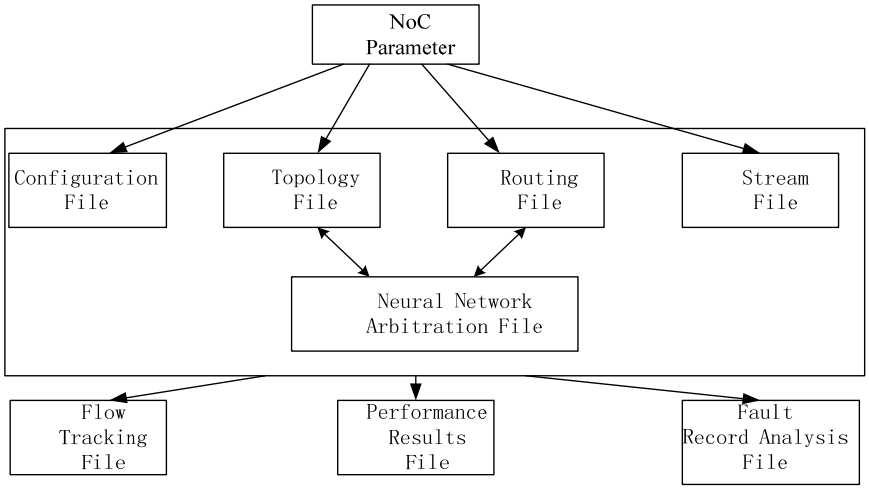

Figure 6. NoC Detailed Structure

Simulation results compared the NoC performance curve. Figure 7 shown the relationship between the data packet transmission delay and the injection rate and Figure 8 shown the relationship between the network throughput and the injection rate, respectively, compared to standard X-Y router.

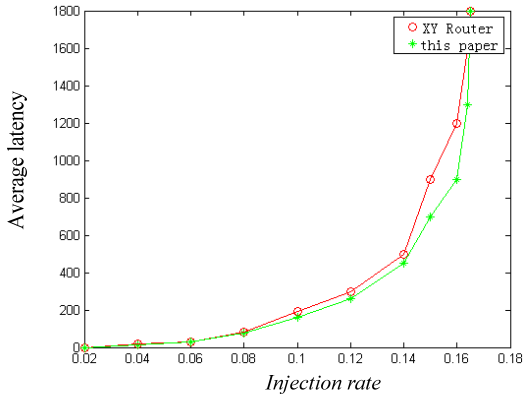

Figure 7. Average Delay of Packet Transmission

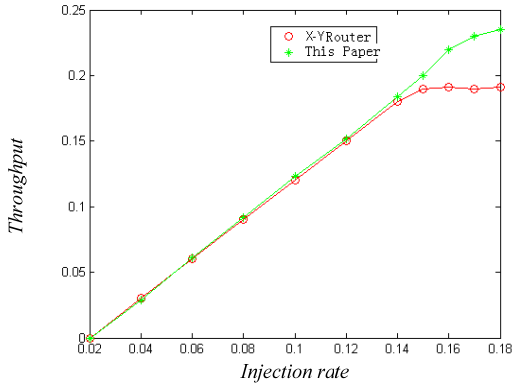

Figure 8. Throughput of Network on Chip

In Figure 7, when the injection rate was low, since the network resources were abundant and no congestion occurred, so both performance curves closed. When at higher injection rate, the network closed to saturation, the average delay of the network was

increased dramatically. When the intermediate state, it was obvious that congestion-aware algorithm outperformed standard X-Y routing.

In Figure 8, the congestion aware control algorithm improved of the performance of the network throughput about 15%.

5. Conclusion

The proposed method of perceived global congestion state in this paper could more effectively improve the performance of a network on chip, especially in the medium saturated network environment, it could reduce the packet transmission delay and at the same time it also could improve the throughput of the network.

Acknowledgement

This work was supported by JiLin Provincial Science&Technology Department.

References

[1] Winter, GP Fettweis. Guaranteed service virtual channel allocation in NoCs for run-time task scheduling. *Design, Automation Test in Europe Conference Exhibition (DATE).* 2011; 1-6.
[2] PH Pham, P Mau, J Kim, C Kim. An On-Chip Network Fabric Supporting Coarse-Grained Processor Array. Very Large Scale Integration (VLSI) Systems. *IEEE Transactions.* 2012; 16(3): 1-5.
[3] Jongman Kim et. *A novel dimensionally-decomposed router for on-chip communication in 3D architectures*. Proceedings of the 34th annual international symposium on Computer architecture. 2007; 138-149.
[4] N Agarwal, T Krishna, LS Peh, NK Jha. GARNET: A detailed on-chip network model inside a full-system simulator. *IEEE International Symposium on Performance Analysis of Systems and Software.* 2009; 33-42.
[5] Nilsson, E Millberg M, Oberg J, Jantsch A. *Load distribution with the proximity congestion awareness in a network on chip*. Design, Automation and Test in Europe Conference and Exhibition 2003; 1126-1127.
[6] Mohammad S Talebi, Fahimeh Jafari, Ahmad Khonsari, Mohammad H Yaghmae. A Novel Congestion Control Scheme for Elastic Flows in Network-on-Chip Based on Sum-Rate Optimization. *Computational Science and Its Applications-ICCSA*. 2007; 398-409.
[7] Pullini A, Angiolini F, Bertozzi D, Benini L. *Fault Tolerance Overhead in Network-on-Chip Flow Control Schemes*. 18th Symposium on Integrated Circuits and Systems Design. 2005; 224-229.
[8] Smiths R, et al. Adaptive Hybrid Learning for Neural Networks. *Neural Computation*. 2004: 16: 139-157.
[9] Schmid A, Leblebici Y, Mlynek D. Hardware realization of a Hamming neural network with on-chip learning. *IEEE International Symposium on Circuits and Systems.* ISCAS. 1998; 191-194.
[10] AK Lusala, JD Legat. Combining SDM-Based Circuit Switching with Packet Switching in a Router for On-Chip Networks. *International Journal of Reconfigurable Computing.* 2012; 1-16.