Hybrid Path-Diversity-Aware Adaptive Routing with Latency Prediction Model in Network-on-Chip Systems

Po-An Tsai[1], Yu-Hsin Kuo[1], En-Jui Chang[2], Hsien-Kai Hsin[2], and An-Yeu (Andy) Wu[1, 2]
[1]Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan
[2]Graduate Institute of Electronics Engineering, National Taiwan University, Taipei, Taiwan

Abstract—**Network-on-Chip (NoC) systems achieve higher performance than bus systems for chip multiprocessor (CMP) systems. However, as the complexity of network increases, routing problems become performance bottlenecks. Conventional routings only use local or regional buffer occupancy (BO) information to choose a better path to deliver a packet. Due to lack of path diversity (PD) information, which is global information, these routings are difficult to spread traffic to different paths for load balancing. Therefore, in this paper, we present a latency prediction model to simultaneously consider PD and BO information. Based on this model, this work proposes Hybrid Path-Diversity-Aware (Hybrid PDA) adaptive routing to overcome congestion problem in NoC. Experiments with different scenarios are conducted. The simulation results show that the proposed selection has a considerable latency reduction over other selection functions, with up to 94.6%, and has better scalability in large scale NoC.**

Keywords-Adaptive routing, network-on-chip, prediction model, path diversity

I. INTRODUCTION

With the development of semiconductor technology, the increased complexity and interconnection delay becomes the limiting faction of System-on-Chip (SoC) performance. Network-on-chip (NoC) has proven as a better interconnection method than bus because of modularity, scalability, reliability, and higher bandwidth [1-4]. The performance in NoC is largely influenced by the underlying routing technique, which chooses a path for a packet and in turn affects the traffic distribution [4]. Therefore, an effective adaptive routing algorithm is desired for NoC to achieve load balancing and high throughput.

Adaptive routing dynamically determines the path for each packet based on network status. It consists of a routing function and selection function, as shown in the bottom of Fig. 1. The routing function generates a set of candidate channels based on turn models [5], while the selection function chooses one candidate based on network information [10], [11]. Numerous factors are involved in designing adaptive routing algorithms, such as deadlock-free, livelock-free, fault tolerant, and adaptiveness [5-8]. Since the trade-off exists between these factors, routing algorithms with different concerns were proposed. No matter what strategies they chose, these routing algorithms all provided low latency in NoC. Strict time constraint is always required in a real-time system. The importance of low latency lies in the fact that the delay of packet sent from source to destination is greatly reduced, yielding a much more balanced traffic load. Therefore, achieving low latency is an ultimate goal for us in designing adaptive routing algorithms.

This work is supported in part by the National Science Council, Taiwan R.O.C., under Grant NSC 100-2221-E-002-091-MY3.

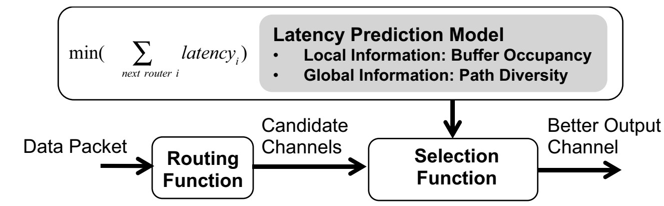


Fig. 1. Hybrid path-diversity-aware adaptive routing scheme. (The grey blocks represent the proposed latency prediction model.)

The level of congestion in NoC determines the latency of transmitted packets and affects the performance. However, adaptive routing algorithms using network information can improve the performance by routing packets through less congested areas and a balanced traffic distribution can be achieved [9]. To well predict the network condition, some routing algorithms using local or regional *buffer occupancy* (BO) information were proposed [11], [13]. They can dynamically detect the change of traffic to choose different paths. A globally adaptive routing algorithm, which uses *path diversity* (PD) as new information in selection, was proposed in [10]. It can efficiently relieve the congestion by forwarding packets to high PD regions.

Although using local or global information alone increases the performance of NoC, both of them ignore their weaknesses. For local information, it is restricted to local region which only provides a limited view of network condition [9]. They only take neighbor regions into consideration but ignore the condition when packets go beyond these regions. It is likely that packets are transmitted to a less congestion neighbor but faced severe congestion in the following transmission. For global information, such as PD, it tries to select the best path in the very beginning to achieve load balancing. However, it neglects the traffic variation in real scenario, which can affect the early prediction. We believe that local and global information are complement to each other and by combining them, the routing algorithm can achieve higher performance.

Therefore, in this paper, we remodel a latency model, which simultaneously considers the number of output path and buffer status, to predict the latency condition of the output channels. Based on this model, we propose *Hybrid Path-Diversity-Aware* (Hybrid PDA) adaptive routing scheme to overcome congestion problem in NoC, as shown in Fig.1. It can further efficiently capitalize on the flexibility of the routing function and improve selection quality by integrating both local BO information and global PD information. Our experimental results demonstrate that the proposed Hybrid PDA routing scheme has higher saturation throughput with an improvement of 3.82%-38.21% compared with existing BO-based adaptive routing schemes in different traffic patterns.

II. Related Works of Adaptive Routing

In this section, we briefly present the classification and the capability of selection functions. Besides, the problem of conventional adaptive routing algorithms is discussed.

A. Classification of Adaptive Routing Algorithms

Adaptive routing algorithms are divided into *fully adaptive* and *partially adaptive routing*. The former routes packets toward all possible directions, which provides the highest adaptiveness but at the risk of deadlock. On the other hand, the latter one adopts deadlock-free turn model at the cost of sacrificing some paths, such as *odd-even routing* [5]. Though the latter one has relative lower adaptiveness, its performance can compete with the fully adaptive routing and also guarantee deadlock-free.

B. Buffer-Occupacy-Based Selection

When the routing function generates a set of output candidates, selection functions are responsible for choosing the better one based on network condition, which can be local or global. Using local information is the most intuitive way and it can be easily derived from the network. For instance, the remaining free-slots in the next router, called *output buffer length* (OBL), can predict the level of congestion to a certain extent. Moreover, the information of free slots in *neighbor-on-path* (NoP) [11] is considered as a criterion in determining the path, and its performance is much better than OBL.

C. Path-Diversity-Based Selection

On the other hand, calculating the path diversity provides the global information of the network. Routing algorithms can use the result, as shown in Fig. 2, to choose the output channel with higher PD to give packet more paths in subsequent transmission [10]. Besides, for example, PD of odd-even-based routing can be calculated by

$$\mathrm{PD} = \frac{[h' + (hc\_y)]!}{h'!\,(hc\_y)!},\ h' = \begin{cases} \dfrac{hc\_x - 1}{2}, hc\_x \text{ is odd} \\ \dfrac{hc\_x}{2}, hc\_x \text{ is even} \end{cases}, \qquad (1)$$

where *hc_x* and *hc_y* are hop counts between source and destination in *x* and *y* direction, respectively.

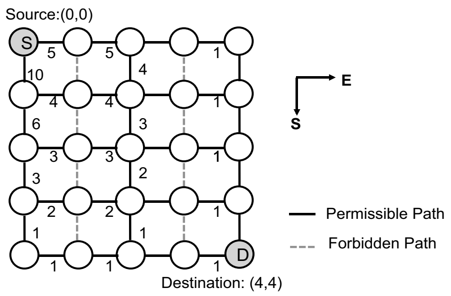


Fig. 2. The difference of path diversity in odd-even routing.

D. Problems of Conventional Adaptive Routing Algorithms

Local or global information alone increases the performance of NoC but has weaknesses. For local information, its effeteness only covered local region which only provides a limited and short-term correctness. For global information, like PD discussed previously, it neglects the traffic variation in real scenario, which may have significant influence in the early prediction. Moreover, none of those selection functions have provided a latency model to illustrate how they decrease the total latency encountered by a packet. Thus, by proposing a latency prediction model, we combine both local and global information to utilize their strength at the same time.

III. Proposed Hybrid Path-Diversity-aware Routing Algorithm with Router Delay Model

Although it is widely known that packets with high path diversity or path with low buffer occupancy will have lower latency, we still need a tangible metric to perform routing algorithm with those information. Also, since path diversity is static information in NoC, we need dynamic information to handle burst in traffic pattern. Therefore, a predictive method integrating path diversity and dynamic buffer occupancy information is necessary to achieve traffic-balancing routing algorithm. By following the analysis as in [12], we will show how path diversity and dynamic information can be combined to be a prediction model.

A. Router Delay Model

We assume that the router adopt wormhole flow control strategy. In Fig. 3, a router delay (***L***) is defined as follow: from the time of the header of a specific packet begins to be served by *R1* until the time of the header begins to be served by *R2*. This delay between the two adjacent routers can be model as two main parts. The first part is caused when the packet tries to cross switches to the output port of *R1*. It starts from the header flit begin to be served until an output port is allocated to that flit. Therefore, it consists of two parameters, *router service time* (***τ***) and *port acquisition delay* (***ρ***). The second part of router delay is caused when the packet goes from the output port of *R1* through the output buffer to the input port of *R2*. This delay also consists of two parameters, *buffer constant delay* (***τ'***) and *buffer transfer delay* (***β***). In summary, router delay (*L*) can be model as the sum of four parameters above:

$$\boldsymbol{L} = \boldsymbol{\tau} + \boldsymbol{\rho} + \boldsymbol{\tau}' + \boldsymbol{\beta} \qquad (2)$$

- ***τ*** and ***τ'***: ***τ*** is *router service time*, defined as the time that a router needs to route a packet. ***τ'*** is *buffer constant delay*, defined as the time required for a flit transfer in to a buffer. Those are two constant delays that depend on the architecture.
- ***ρ*** is *port acquisition delay*, representing the average delay of packet waiting for another disrupting packet release the output port. This delay is highly related to the arbiter in the router. If the arbiter is perfectly impartial to any input port, then every packet should have the same port acquisition delay.
- ***β*** is *buffer transfer delay*, mainly results from waiting for the output buffer to be empty. The more flits in the output buffer cause higher buffer transfer delay. Therefore, for a single packet, choosing different output ports may lead to different buffer transfer delay.

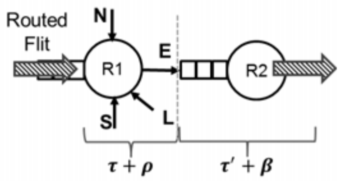


Fig. 3. The model of router delay.

These delays reflect how different architecture, arbiter, flow control technique, and routing algorithm can influence the total path latency of a packet. However, when we consider how selection function affects the latency of a packet, not all the parameters should be taken into account. In fact, as we know that ***τ*** and ***τ'*** are constant in the same architecture and that ***ρ*** is greatly influenced by arbiter, only *buffer transfer delay* (***β***) should be concerned when setting latency model for selection function.

B. Latency Prediction Model for Selection

The buffer transfer delay (***β***) is determined by the output port that is chosen by selection function. Therefore, the goal of selection function is to find output port that causes lower **β** for the packet. That is, by having an evaluation model of **β**, selection function can obtain an accurate metric to choose correct output port.

β is counted as the time that the header flit needs to go from output port to input port of next router. As a result, this delay is related to how many flits does the header expect to wait for in the buffer. In Fig. 4, there are two cases that may cause this situation:

- **Case 1**: When another packet is transmitted through the output port, the router R1 does not release the output port yet. In this case, the expectation of flits in buffer is calculated as the probability of contention (**π**), which represents how often a packet will encounter another packet in the output port it needs, multiplying the buffer occupancy (**δ**), how much buffer the encountered packet occupies.

- **Case 2**: Another packet has already passed through the output port, but the last few flits are still in the buffer. In this case, the expectation of flits in buffer is calculated as probability of buffer occupancy (**π'**), how often the residual packet will in the output buffer, multiplying the buffer occupancy (**δ'**), how much buffer the residual packet occupies.

In sum, **β** can be model as:

$$\boldsymbol{\beta} = \boldsymbol{\pi}\times\boldsymbol{\delta} + \boldsymbol{\pi}'\times\boldsymbol{\delta}' \quad (3)$$

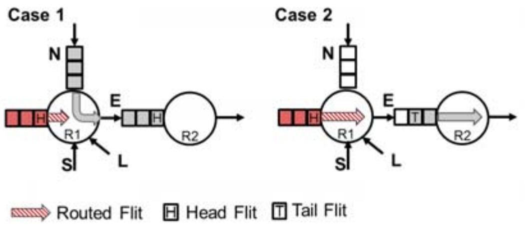


Fig. 4. Two situation causes buffer transfer delay.

Delay in the first case directly results from the occupancy of output port by other packets. Therefore, to reduce this delay, traditional selection functions, such as OBL, NoP, and PDA, will choose output port that is available at that time. In this way, most selection function can lower **β** to a certain level. However, not all selections can predict delay caused by second case well. That is, it's hard to model **π'** and **δ'**, which is the main research of selection functions.

The probability of buffer occupancy (**π'**) can be interpreted as the ability to output packet for the next router. If the ability is stronger, then a residual packet must stay less time in the output buffer. Therefore, we can simply derive that:

$$\pi' \propto {}^{1}/_{Ability\ to\ output\ packet} \quad (4)$$

Notably, path diversity is one of the most important factors that influence the ability to output packet. If there is higher path diversity in router, then this router must have better ability to let packet out because there are more choices to route packets in this router. Thus, there is positive correlated relationship between ability to output packets and path diversity:

$$Ability\ to\ output\ packet \propto Path\ Diversity \quad (5)$$

Combining (4) and (5), we have:

$$\pi' \propto {}^{1}/_{Path\ Diversity} \quad (6)$$

In this manner, we can model **π'** in a better way because path diversity of a router can be computed offline and stored in a table.

Furthermore, buffer occupancy (**δ'**) can be modeled by the number of free buffers of an output port. The more free buffers an output port has, the lower **δ'** it has. Therefore, we can easily derive that:

$$\delta' \propto {}^{1}/_{\#\ of\ Free\ buffer} \quad (7)$$

Further combining (3), (6), and (7), when all the output ports are available, our latency model for selection to predict the relative latency of two output port can be represented as following:

$$\boldsymbol{\beta} \propto \frac{1}{Path\ Diversity \times \#\ of\ Free\ buffer} \quad (8)$$

By multiplying path diversity and number of free buffer, we can predict the latency of choosing one of the possible output ports. Namely, by using this hybrid information as a metric, routing can compare which output port is better at that time.

C. Hybrid Path-Diversity-Aware Routing Algorithm

Based on this path-diversity-based prediction model of latency, we propose a *hybrid path-diversity-aware* (Hybrid PDA) *routing scheme*, which consists of deadlock-free odd-even routing function and *hybrid PDA selection functi*on, as shown in Fig. 1. The pseudo code of hybrid PDA selection is shown in Fig. 5. The number of path diversity is computed and stored as in [10]. We first compute the path diversity from the router to any other routers and store the value in a table. When a new packet comes, selection function can then look up the table with the destination of the packet.

The block diagram for Hybrid PDA selection is shown in Fig. 6. Most of the design follows those in [10], with two extra multipliers and one comparator, which are negligible when comparing with whole router architecture. Besides, additional wires and multiplexers are used for the transmission of free slot information. The number of free output buffer can be calculated by next router and transmitted to the current buffer. Once the hybrid PDA has two kinds of information, it first checks whether both output port is available. If one of them is not available, hybrid PDA directly chooses the other output port. If both of them are available, then hybrid PDA compares their predicted **β** and selects the output port with lower predicted **β**.

This selection function integrates global information and local information, path diversity and free buffers. Adopting (8) in selection function can bring both advantages in two kinds of information and further achieve lower latency than using one of them. Also, the prediction model comes from the offline analysis model and therefore has high accuracy.

The Procedure of Hybrid PDA Selection

```
Hybrid PDA (in : CCh out:O_port )  // CCh: candidate channels
{
if (one of CCh is not available) then O_port ← another output port;
L1 = path diversity of output port 1 * free slots of output port 1;
L2 = path diversity of output port 2 * free slots of output port 2;
if ( L2>L1) then O_port ← output port 2;
else  O_port ← output port 1;
}
```

Fig. 5. The pseudo code of Hybrid PDA selection.

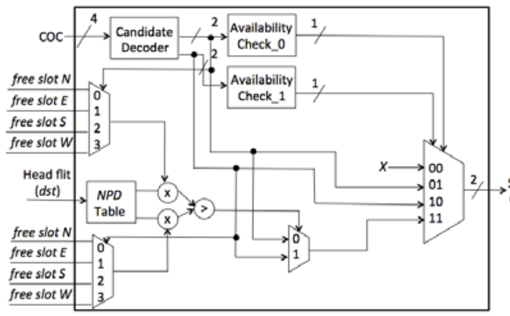


Fig. 6. The block diagram for Hybrid PDA selection.

IV. Performance Evaluation

The experiments are evaluated by the NoC Simulator, Noxim [13]. The topology is a 16×16 mesh network. The system runs with the wormhole switching mechanism and the matrix arbitration. Besides, each channel has an input buffer with the size of 4 flits, and each packet has 8 flits. The cycle of handshaking is one. We use the odd-even adaptive routing function to do the simulations under different traffic patterns, such as *transpose1* and *random traffic*. In transpose1 traffic, a source node (i, j) only sends packets to node (16-j, 16-i). In random traffic, each source is equally likely to send to each destination. The temporal distribution of traffic is Poisson distribution. For each run of simulation, the total time is 20,000 cycles, and the first 2,000 cycles is warm-up time of NoC system. The latency at the saturation throughput of Hybrid PDA is the performance metric in our experiments, and its definition is where average latency equals to twice of the zero-load latency [10].

Experiment 1:Hybrid Information versus Single Information

To see how hybrid information, buffer information plus path diversity information, affects the overall performance, we compare it with different selection functions using only local or global information in *transpose1* and *random* traffic. In Fig. 7 (a) and (b), Hybrid PDA selection is superior to all the other selections in latency. The latency improvements of Hybrid PDA are shown in Table I.

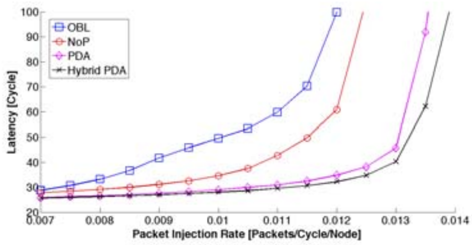
Under transpose1 traffic, the latency reductions are 27.3%, 66.2%, and 70.6% compared to PDA, NoP, and OBL, respectively. Also, under random traffic, the latency reductions are 40.5%, 76.7%, and 94.6% compared to PDA, NoP, and OBL, respectively. Besides, since these synthetic traffics have regular source-destination pairs, the improvement is not significant. This phenomenon can be seen when we compare Hybrid PDA with PDA. We infer that PDA can predict and react well enough under such predictable traffic patterns and therefore the improvement is limited.

Experiment 2: Performance Scalability of Hybrid PDA

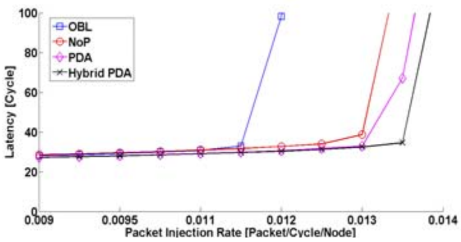
To see whether Hybrid PDA still has scalability, we examine the network throughput in different topology sizes. The *network throughput* (*NT*) is defined as the accepted traffic of the network at a given latency. Here, we use the latency of *Experiment 1*, which equals to twice of the zero-load latency in a 16 × 16 mesh NoC. Besides, we use the value in 8 × 8 mesh as our standard *NT*. In Fig. 8, for a given network size, Hybrid PDA performs better than the other selection functions, especially for large network sizes. Notably, OBL and NoP saturate much sooner than Hybrid PDA, and they only increases by 1.47 and 1.97 times in a 20 × 20 mesh. The *NT* of Hybrid PDA, in contrast, grows steadily as the network size increases. Like PDA, Hybrid PDA still improves the scalability of performance.

Experiment 3: Real Traffic Scenario

In a realistic traffic scenario, we consider a generic *mms* [14], which includes an H.263 video encoder, an H.263f video decoder, an MP3 audio encoder, and an MP3 audio decoder.



(a)



(b)

Fig. 7. The latency variation of PDA under (a) *transpose1* and (b) *random* traffic pattern.

TABLE I. PERFORMANCE EVALUATION OF PROPOSED ALGORITHM

	Latency (cycles)				Reduction Rate (%)		
	OBL	*NOP*	*PDA*	*Hybrid*	*OBL*	*NOP*	*PDA*
Trans.1	170.1	148.1	68.8	50	70.6	66.2	27.3
Random	924.6	215.0	84.1	50	94.6	76.7	40.5

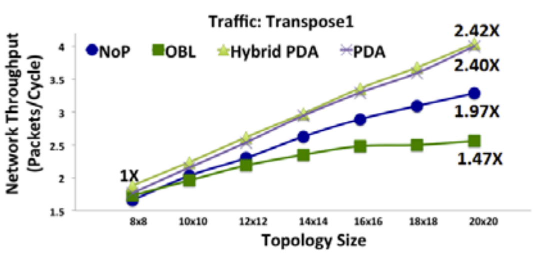


Fig. 8. The network throughput of different topoloty sizeds for different selection functions.

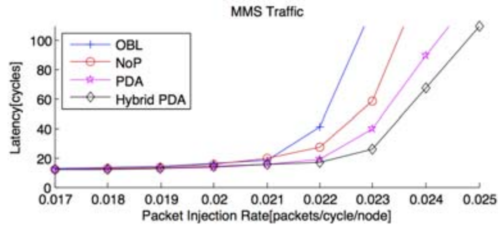


Fig. 9. The latency variation under the *mms* traffic scenario.

These tasks of application are assigned and scheduled onto 5 × 5 selected IPs as in [14]. As we can observe from Fig. 9, Hybrid PDA further improves PDA and outperforms the other selections. For a *pir* value of 0.023, Hybrid PDA exhibit an average delay 25 cycles and others exhibit from 40 to 123 cycles. With the progress of time, the application will be assigned onto a large-scale topology in the future and the improvement will become much clearer, which is supported by the results of *Experiment 3*.

VI. CONCLUSION

This paper presents a path-diversity-based latency model for selection. We propose hybrid PDA selection using both global path diversity information and local buffer information. From our experimental results, using combined information in selection can effectively improve the performance of NoC in every traffic scenario and reduce latency up to 94.6%. Besides, hybrid PDA outperforms other selections using only one kind of information. On the other hand, our proposed selections have great scalability. Hybrid PDA provided an enormous potential for large-scale NoC in the future design.

ACKNOWLEDGMENT

The authors would like to thank the helps provided by the members of NTU Access IC Lab.

REFERENCES

[1] R. Marculescu, Y. Ogras, L.-S. Peh, N. D. E. Jerger and Y. V. Hoskote, "Outstanding research problems in noc design: System, microarchitecture and circuit perspectives," *TCAD*, vol. 28, no. 1, pp. 3-21, 2009.

[2] L. Benini and G. DeMicheli, "Networks on Chip: A New SoC Paradigm," *IEEE Computer,* 35(1): pp. 70-78, Jan. 2002.

[3] W. J. Dally and B. Towles, "Route Packets, not Wires: On-Chip Interconnection Networks," *Proc. ACM/IEEE Design Automation conf.,* pp. 684-689, 2001

[4] D. Wu, B. M. Al Hashimi, M. T. Schmitz, "Improving routing efficiency for network-on-chip through contention-aware input seleciton," *in proceddings of ASPDAC*, pp. 36-41, 2006.

[5] G.-M. Chiu, "The Odd-Even Turn Model for Adaptive Routing," *IEEE Trans. On Parallel and Distributed Systems,* vol. 11, no. 7, pp.729-738, July. 2000.

[6] W. J. Dally and B. Toweles, *Principles and Practices of Interconnection Networks,* Morgan Kaufmann, 2004.

[7] N. E. Jerger and L.-S. Peh, *On-Chip Networks,* Morgan and Claypool, 2009.

[8] F. Gebali and H. Elmiligi, *Networks on Chip: Theory and Practive,* Taylor and Francis Group LLC-CRC Press, 2009.

[9] M. Ebrahimi, M. Daneshtalab, P. Liljeberg, J. Plosila and H. Tenhunen, "CATRA- Congestion Aware Trapezoid-based Routing Algorithm for On-Chip Networks," *Design, Automation & Test in Europe*, pp.320-325, 2012.

[10] Y.-H. Kuo, P-A. Tsai, H.-P. Ho, E.-J. Chang, H.-K. Hsin and A.-Y. Wu, "Path-Diversity-Aware Adaptvier Routing in Network-on-Chip Systems," *IEEE 6th International Symposium on Embedded Multicore SoCs,* 2012.

[11] G. Ascia, V. Catania and M. Palesi, "Implementation and Analysis of a New Selection Strategy for Adaptive Routing in Networks-on-Chip, " *IEEE Trans. Computers*, vol. 57, no.6, pp.809-920, June. 2008

[12] S. Foroutan, Y. Thonnart, F. Petrot, "An Iterative Computational Technique for Performance Evaluation of Networks-on-Chip," *IEEE Trans. Computers*, April. 2012.

[13] "Noxim: Network-on-Chip Simulator," http://sourceforge.net/projects/noxim , 2008.

[14] J. Hu and R. Marculescu, "Energy- and Performance-Aware Mapping for Regular NoC Architectures," *IEEE Trans. Computer-Aided Design of Integrated Circuits and Systems*, vol. 24, no. 4, pp. 551-562, Apr. 2005