Investigating Performance Advantages of Random Topologies on Network-on-Chip

Sarat Yoowattana
Asian Institute of Technology, Thailand
sarat.yoowattana@gmail.com

Ikki Fujiwara, Michihiro Koibuchi
National Institute of Informatics, Japan
{ikki,koibuchi}@nii.ac.jp

Abstract— As technology continues to scale down, the number of cores significantly increases, e.g. 64 cores. The communication latencies increasingly give the negative impact on the performance of parallel applications on Chip MultiProcessors (CMPs). A random topology, which provides lowest diameter and average shortest path length, has been recently considered for low-latency Network-on-Chip (NoC). In this work we investigate its advantage in throughput-and-latency properties for various traffic patterns and we compare the random topology with traditional non-random topologies, such as two-dimensional mesh in various network sizes. Thorough our cycle-accurate network simulation, we found that the random topology significantly outperforms 2-D mesh and 2-D torus in terms of network latency.

I. Introduction

Recently, Network-on-Chips (NoCs) have been used in chip multi-processors (CMPs) [1, 2, 6, 11, 14] in order to connect a number of processors and cache memories on a single chip, instead of traditional bus-based on-chip interconnects that suffer from poor scalability. NoCs can be evaluated in various aspects, such as throughput, communication latency, hardware amount and power consumption. Reducing the communication latency among them is the most important aspect in CMPs. As technology scales down, the number of cores continues to increase on a chip. It is easily expected that the communication latency increasingly have a negative impact on the performance of parallel applications on CMPs. We thus focus on the design of low-latency network topology to reduce the communication latency. Although there are a variety of topologies that have a good layout on a homogeneous chip, their diameter and average shortest path length (ASPL) are generally large. For example, $n \times n$ two-dimensional mesh has a diameter of $2n - 1$. By contrast, random topologies, which are generated by augmenting a tree topology with random links, surprisingly provide the minimum diameter and ASPL.

In this work we investigate the advantages of random topologies in latency-and-throughput performance using a cycle-accurate network simulation. Our main findings are as follows:

- For synthetic imbalanced and well-distributed traffic patterns in which a node independently communicates with each other, the random topologies outperform the same-degree non-random traditional topologies in terms of both network latency and throughput.
- For a benchmark traffic pattern that models some parallel CMP applications, non-random traditional topologies provide good performance properties in small-sized networks, i.e. up to 64 cores; while random topologies provide better performance for larger-sized networks.

The paper is organized as follows. Section II describes related work. Section III shows the simulation results for each topology with various traffic patterns. Section IV makes our conclusions.

II. Related Work

A. Existing On-chip Topologies

There exists a variety of on-chip topologies and their layouts for NoCs. The k-ary 2-meshes and folded k-ary 2-tori have intuitive layouts that make each link length uniform and short [5]. The butterfly networks (k-ary n-fly) can be efficiently mapped onto a 2-D VLSI by utilizing high-radix routers [9]. In the flattened butterfly [8], routers in each row of a conventional butterfly are combined into a single router. It has a large diversity of router degrees for each network size. Spidergon topology, which is a ring topology with links that connect diagonal counterparts on the ring, has been discussed for cost-effective on-chip networks [4]. It can be efficiently mapped onto a chip, as well as k-ary 2-meshes, in which almost all links have a minimum length. However, its average hop counts considerably increase as the number of nodes increases, even though it provides diagonal links to mitigate the increase of diameter compared to a conventional ring.

When considering low diameter and ASPL especially in a large network size, such as 256 cores, random topologies, in which each link is connected to nodes that are randomly selected, are better. Another option to reduce path hops by the random effect is the small-world topologies [12]. It consists of non-random topology with a small number of random shortcut links. In this work, our purpose is to reduce the communication latency of NoCs. In terms of ASPL and diameter, random topology is better than small-world network. We thus attempt the evaluation of random topologies.

B. Long-link Implementation on a Chip

To map random topologies onto a chip, we have to consider how to implement their long links on a chip. It is an open issue

not only for random topologies but also for high-radix non-random topologies. The simplest way is to insert the optimal number of repeaters on a long link. The link delay of metal wiring is ideally less than 50 ps/mm under 65nm process [15]. Thus the long-link would maintain the chip frequency as far as the chip frequency is not much high.

Another way is to use wireless technology to connect distant nodes on a chip. Candidate technologies include 60GHz directional radio wave and free-space optics. We consider the implementation of long links is possible though its implementation technology is out of the scope in this work.

III. Evaluations

The NoCs that have random topologies were evaluated through cycle accurate network simulations in terms of the network latency and throughput. They are compared to typical non-random topologies, namely two-dimensional mesh and torus. All the topologies take the same degree, four, in order to make a fair comparison.

A. Cycle-accurate Network Simulation

A flit-level simulator written in C++ is used to measure the throughput [10]. Every on-chip router thus has three, four or five ports. A single processing element (PE) is connected to each router. "Core" thus consists of a PE and its local on-chip router. Wormhole switching is used as the switching technique of the router. Three clock cycles are required for a flit to pass through a router, that is, one clock for routing, one for transferring the flit from input channel to output channel through a crossbar, and one for transferring the flit to the next node. The PEs inject packets independently of each other. We set the packet length at 8 flits, including one header flit. To take minimal paths, we use up*/down* routing with escape path for all the cases except that no deactivated cores are used. The number of cores on a chip, N, is set to $16 \leq N \leq 100$.

We simply compare 2-D mesh, 2-D torus, and random topology in this study. Although application sepcific NoC design would generate a good topology for a give traffic pattern [3], it is out of the scope in this study when considering the ease of understanding.

Random topologies in which each node has the same degree are generated under the condition that a single link connects two different switches as follows: after each node is connected by a tree, the remain links are randomly added. Such random topology generation is discussed in [7]. The approach for building random topologies does not take into account the quality/usefulness of the random shortcuts. However random generation does not impact on the throughput and latency significantly. In this evaluation, we thus pick up a single random generation to plot the graph.

B. Simulation Results under Synthetic Traffic Patterns

Three traffic patterns are used in the simulation assuming that binary coordinates are assigned to each core.

- *uniform*
  All destination nodes are selected randomly, and so the traffic is distributed uniformly.
- *non-uniform*
  - *bit-reversal*
    A node with the identifier $(a_0, a_1, \cdots, a_{n-1})$ sends a packet to the node whose identifier is the bit reversal $(a_{n-1}, \cdots, a_1, a_0)$ of the source node.
  - *matrix transpose*
    A node (x, y) sends a packet to the node $(k-y-1, k-x-1)$(k is the number of nodes in each dimension) or $(k-x-1, k-y-1)$ when $x+y = k-1$.

Figure 1 shows the latency vs. accepted traffic load for random, 2-D mesh and 2-D torus under the synthetic traffic patterns. Each legend represents the topology, its average shortest path length (ASPL) in the target traffic pattern and degree. The X-axis is the accepted traffic, whereas the Y-axis is the average network latency whose unit is the simulation cycle. The lower latency is thus better and the maximum value of the accepted traffic can be regarded as the network throughput.

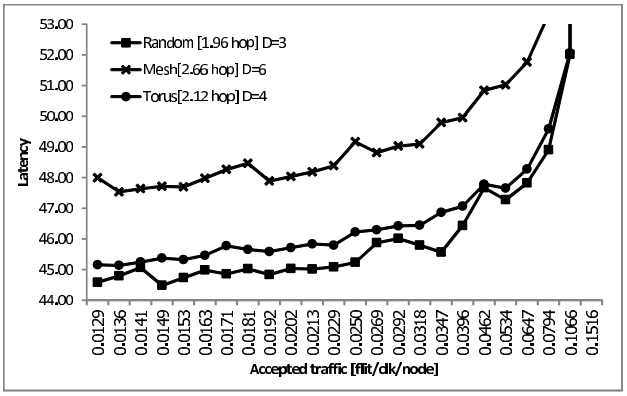
The main concern is the network latency. Although the latency curve of the random topology is not well stabilized, its value is drastically lower than that of 2-D mesh and 2-D torus at each accepted traffic load. This comes from that the diameter and ASPL of random topologies are empirically better than those of 2-D mesh and 2-D torus. Its improvement becomes large in larger network size.

When considering the performance affect given by traffic patterns, as expected, non-uniform traffic patterns, matrix transpose and bit reversal, gives larger impact on the latency gaps between random and non-random topologies.

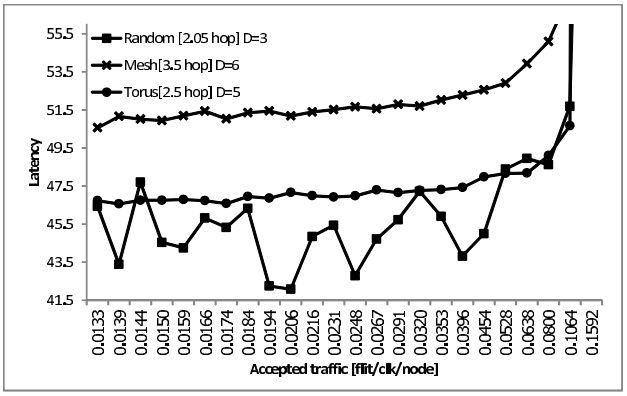
C. Simulation Results under Traffic Pattern of Parallel Benchmark

The MCSL NoC Benchmark Suite [13] is used to evaluate random, mesh and torus topologies. The benchmark provides seven sets of traffic patterns based on real applications. Each set supports network size varied from 4x4 to 16x16. There are two versions of the traffic patterns, a recorded traffic pattern (RTP) and a statistical traffic pattern (STP). The RTP is beneficial for accurate NoC simulation, while the STP is designed for simulating long application execution steps. In this evaluation, we used RTP traffic pattern.A file format of the RTP traffic pattern provides execution tasks and communication tasks. We ignored the execution task because our research focuses on network communication. A PB block was assumed in order to reduce the simulation execution time. The communication task contains the information of source node, destination node, sequence number and message size. The message size is given in word (32 bits).

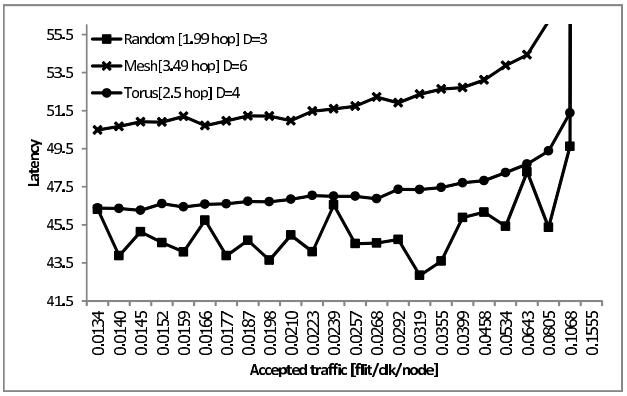
Figure 2 shows the simulation results of random, 2-D mesh and 2-D torus under MCSL NoC Benchmark Suite. We found that the random topology has lowest latency, hop count and diameter in the uniform, bit reversal and matrix transpose traffic



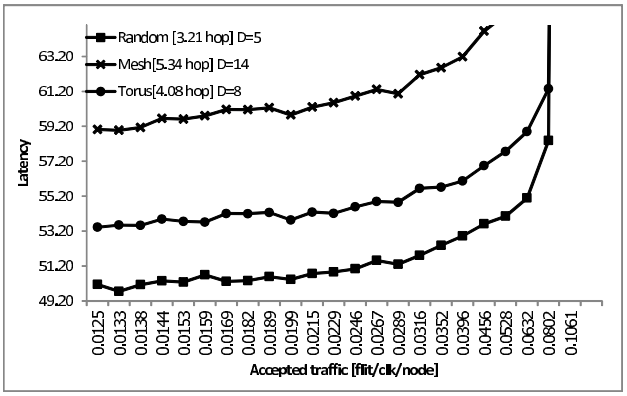
(a) Uniform traffic, 16 Cores



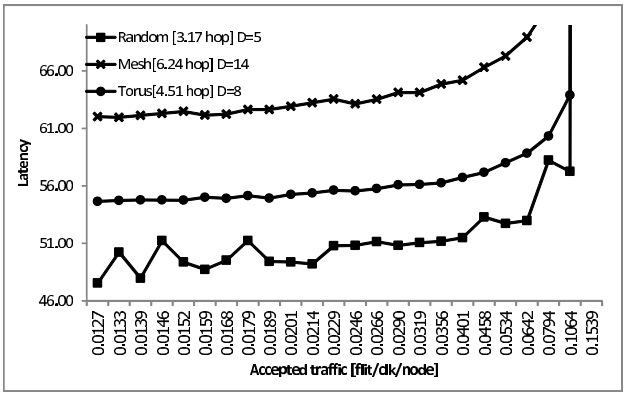
(b) Matrix transpose traffic, 16 Cores



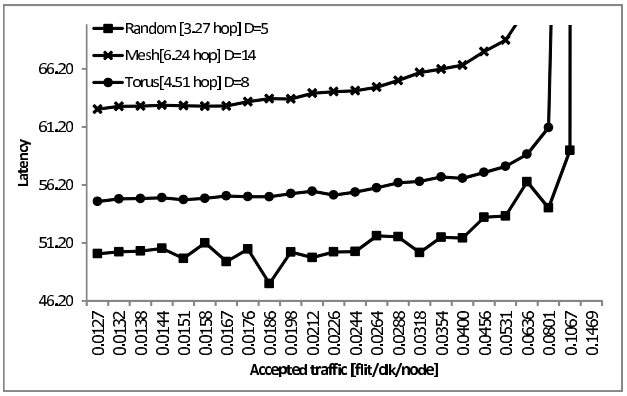
(c) Bit reversal traffic, 16 Cores



(d) Uniform traffic, 64 Cores

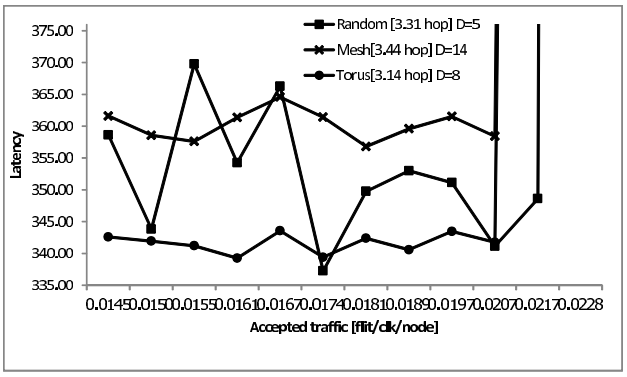


(e) Matrix transpose traffic, 64 Cores

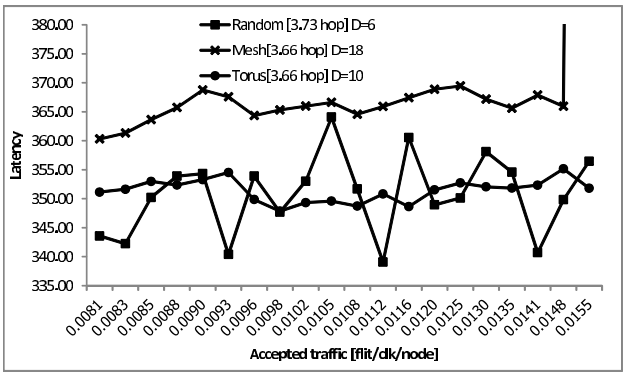


(f) Bit reversal traffic, 64 Cores

Fig. 1. Accepted traffic vs. latency for random, 2-D mesh and 2-D torus under synthetic traffics



(a) Fpppp, 64 Cores



(b) Fpppp, 100 Cores

Fig. 2. Accepted traffic vs. latency for random, 2-D mesh and 2-D torus under MCSL NoC Benchmark Suite

patterns. As well as the results of random topologies under synthetic traffic patterns, their performance is not well stabilized under Fpppp applications.

Surprisingly, in the case for small-sized networks, 2-D mesh and 2-D torus sometimes outperform the random topologies. The traffic patterns generated by parallel applications often have a strong locality. The average packet hops shown in the legend on the graph are similar to those of 2-D mesh and 2-D torus. The main reason why the random topologies provide better performance is to reduce the path hops. Thus, this advantage of the random topologies becomes small in the traffic patterns generated by parallel applications.

Another finding is that the random topologies outperform 2-D mesh and 2-D torus again in the large network sizes. This is because the influence of packet path hops relatively becomes large on the performance of NoCs.

IV. Conclusions

As technology continues to scale down, the number of cores on a chip increases significantly. The communication latencies increasingly give a negative impact on the performance of parallel applications on Chip MultiProcessors (CMPs). A random topology, that provides lowest diameter and average shortest path length, has been recently considered for low-latency network-on-chip (NoC). In this work we investigate its throughput-and-latency properties for various traffic patterns and compare the random topology with traditional non-random topologies, such as two-dimensional mesh, in various network sizes. Thorough our cycle-accurate network simulation, the random topologies outperform the same-degree non-random traditional topologies in terms of both network latency and throughput for synthetic imbalanced and well-distributed traffic patterns in which a node independently communicates with each other. By contrast, for a benchmark traffic patterns, non-random traditional topologies provide good performance properties in small-sized networks, i.e. up to 64 cores; while random topologies provide better performance for larger-sized networks. Since the number of cores will become larger in a chip, we optimistically consider that the random topologies are valuable in future NoCs.

Acknowledgements

This work is partially supported by NII Joint Research Fund and KAKENHI 25280043.

References

[1] T. W. Ainsworth and T. M. Pinkston. Characterizing the Cell EIB On-Chip Network. *IEEE Micro*, 27(5):6–14, Sept. 2007.

[2] B. M. Beckmann and D. A. Wood. Managing Wire Delay in Large Chip-Multiprocessor Caches. In *Proceedings of the International Symposium on Microarchitecture (MICRO'04)*, pages 319–330, Dec. 2004.

[3] L. Benini. Application specific noc design. In *DATE*, pages 491–495, 2006.

[4] M. Coppola, R. Locatelli, G. Maruccia, L. Pieralisi, and A. Scandurra. Spidergon: a novel on-chip communication network. In *Proceedings of the International Symposium on System-on-Chip (ISSOC'04)*, page 15, Nov. 2004.

[5] W. J. Dally and B. Towles. Route Packets, Not Wires: On-Chip Interconnection Networks. In *Proceedings of the Design Automation Conference (DAC'01)*, pages 684–689, June 2001.

[6] P. Gratz, C. Kim, K. Sankaralingam, H. Hanson, P. Shivakumar, S. W. Keckler, and D. Burger. On-Chip Interconnection Networks of the TRIPS Chip. *IEEE Micro*, 27(5):41–50, Sept. 2007.

[7] A. Jouraku, M. Koibuchi, and H. Amano. An Effective Design of Deadlock-Free Routing Algorithms Based on 2-D Turn Model for Irre gular Networks. *IEEE Transactions on Parallel and Distributed Systems*, 18(3):320–333, Mar. 2007.

[8] J. Kim, J. Balfour, and W. J. Dally. Flattened Butterfly Topology for On-Chip Networks. In *Proceedings of the International Symposium on Microarchitecture (MICRO'07)*, pages 172–182, Dec. 2007.

[9] J. Kim, W. J. Dally, B. Towles, and A. K. Gupta. Microarchitecture of a High-radix Router. In *Proceedings of the International Symposium on Computer Architecture (ISCA'05)*, pages 420–431, June 2005.

[10] M. Koibuchi, H. Matsutani, H. Amano, D. F. Hsu, and H. Casanova. A Case for Random Shortcut Topologies for HPC Interconnects. In *Proc. of the International Symposium on Computer Architecture (ISCA)*, pages 177–188, 2012.

[11] A. S. Leon, K. W. Tam, J. L. Shin, D. Weisner, and F. Schumacher. A Power-Efficient High-Throughput 32-Thread SPARC Processor. *IEEE Journal of Solid-State Circuits*, 42(1):7–16, Jan. 2007.

[12] Ü. Y. Ogras and R. Marculescu. "It's a small world after all": NoC performance optimization via long-range link insertion. *IEEE Trans. VLSI Syst.*, 14(7):693–706, 2006.

[13] The MCSL NoC Benchmark Suite. http://www.ece.ust.hk/ẽexu/index_files/benchmark.htm.

[14] D. Wentzlaff, P. Griffin, H. Hoffmann, L. Bao, B. Edwards, C. Ramey, M. Mattina, C.-C. Miao, John F. Brown III, and A. Agarwal. On-Chip Interconnection Architecture of the Tile Processor. *IEEE Micro*, 27(5):15–31, Sept. 2007.

[15] N. Weste and D. Harris. *CMOS VLSI Design: A Circuits and Systems Perspective (4th Edition)*. Addison-Wesley, 2010.