To appear in *Int'l Journal of Low Power Electronics*, American Scientific Publishers, 2009

NED: A Novel Synthetic Traffic Pattern for Power/Performance Analysis of Network-on-chips Using Negative Exponential Distribution

Amir-Mohammad Rahmani[1], Ali Afzali-Kusha[1], and Massoud Pedram[2]

[1]Dept. of Electrical and Computer Eng., Univ. of Tehran, Tehran, Iran,
am.rahmani@ece.ut.ac.ir, afzali@ut.ac.ir

[2]Dept. of EE-systems, Univ. of Southern California, Los Angeles, CA 90089
pedram@usc.edu

Abstract *— In this paper, we present an empirically-derived synthetic traffic model based on the Negative Exponential Distribution (NED) for homogenous and heterogeneous Network-on-chips (NoCs) with any dimensionality. Compared to conventional synthetic traffic profiles, this synthetic traffic profile accurately captures key statistical behavior of realistic traces obtained by running different applications on Network-on-chips. To assess the usefulness of this new NoC traffic model, the average packet hops for the proposed traffic profile is compared with those of some synthetic and realistic traffic patterns. The results show that the NED traffic profile has more similarity with the realistic traffic profiles than those of conventional synthetic ones. Adding this traffic profile to the existing profiles, improves the design and characterization of NoCs.*

Keywords — Network-on-chips, Power/Performance Analysis, Negative Exponential Distribution

1 INTRODUCTION

Following the Moore's law, on-chip transistor densities have increased steadily enabling the integration of dozens of components on a single die. These components include regular arrays of processors and cache banks in tiled chip multiprocessors (CMPs) and heterogeneous resources in system-on-chip (SoC) designs [1][2][3]. One outcome of greater integration is that interconnection networks begin to replace shared buses and other forms of communication featuring long global wires. Networks-on-chip (NOCs) scale better than traditional forms of on-chip interconnect and have better performance and fault tolerance characteristics [1][4]. Many research groups have devoted their efforts to different aspects of NoC's, including topology, routing algorithms and architectures, power management, and core mapping (see, e.g., [2][5][6]).

To assess the performance of various NoC options over the full range of input characteristics, simulations with different traffic profiles should be used. These simulations are used to determine the power and latency characteristics of a given NoC architecture. Traffic profiles may be classified as either synthetic or realistic. Synthetic traffic patterns are abstract models of message passing in NoCs whereas realistic traffic patterns are traces of real applications running on NoCs. In contrast to realistic traffic which is representative of a more specific class of applications, synthetic traffic should cover a broad class of applications running on the NoCs [7][8].

Simple synthetic traffic patterns include uniform random and hotspot traffic profiles. These models are helpful in that they allow a network to be stressed with a regular, predictable pattern which aids NoC designers in acquiring insights, however since they do not represent real-life traffic, they cannot be used to drive a realistic network design-space exploration [8][9]. Some of the other synthetic traffic profiles include Transpose, Bit-Complement, Bit-Reversal, and Self-Similar patterns. Recently, several synthetic traffic profiles have been proposed. In [10], a traffic model for on-chip networks is proposed which is a good model for the multimedia applications running on NoC, but it may not be suitable for other applications.

In this paper, we present a synthetic traffic profile based on negative exponential distribution (NED). Compared to the conventional synthetic traffic profiles, this synthetic traffic profile is more similar to the statistical behavior of realistic traces obtained by running different applications on NoCs.

The rest of the paper is organized as follows. Section 2 briefly introduces related works in this area and presents the motivation of presenting a new traffic profile for the NoC analysis. The NED traffic model and its properties are given in Section 3 while the comparison of the results of this traffic profile with those of others is discussed in Section 4. Finally, the conclusion is given in Section 5.

2 RELATED WORKS

Network performance depends highly on the network traffic, and hence, access to different traffic models is critically important for a thorough analysis of different characteristics of the network architectures, protocols, and implementations. There has been extensive prior research work on traffic models for different networks ranging from the Internet [11][12], Ethernet [13][14], wireless LANs [15] to shared-memory-multiprocessor (SMP) networks [16][17][18]. These models provide critical insights into the traffic behavior of the corresponding networks. Network-on-chips, however, exhibit substantially different traffic behaviors compared to those of the traditional network fabrics. The differences are self-evident versus the Internet, Ethernet and wireless LANs, where the applications, protocols, and implementations differ simply in purpose, scale and granularity [8][19]. Even compared to traditional multi-chassis interconnection networks of SMPs, clusters and supercomputers, multi-core on-chip systems have significantly different network traffic characteristics for a number of reasons [18]. First, although intra-chip (on-chip) systems have a memory hierarchy similar to that of inter-chip systems, the shorter communication delays in the on-chip systems and the limited silicon area suggest a more compact cache organization with small L1 caches and shared L2 caches [20][21]. There are more communication transactions among the on-

chip cores leading to NoC traffic that behaves differently from that of inter-chip systems. Second, new chip multiprocessor (CMP) architectures are being explored aggressively in industry and academia [22][23][24], where each CMP architecture and its organization uniquely affect the volume and profile of the NoC traffic. Third, in application-specific MPSoCs, each MPSoC and its associated on-chip network are designed for a distinct set of target applications [25][26], and hence, network designs and associated network traffic are very different from those of large-scale, traditional interconnection networks.

As discussed in the previous section, traffic profiles for the design and the analysis of NoCs can be categorized into realistic and synthetic groups. Next, we briefly discuss widely used traffic profiles.

2.1 Realistic Traffic

Realistic traffic loads have been used to analyze the power and delay of NoCs. Examples include GSM voice CODEC [27], SPLASH-2 [28], MediaBench [29], and SPEC [30] traffic profiles. It, however, should be noted that the traffic patterns generated by different modules in a NoC strongly depends on the application for which the NoC is designed. Since performance of the NoC is a function of the traffic profile, the most accurate way to assess the characteristics of the NoC would be to invoke the traffic profiles corresponding to the application. In many cases, the system is designed for multiple applications. In these cases, the traffic profiles corresponding to all applications should be used during the NoC design and analysis. This can be time consuming even if all the applications are known beforehand. As another option, synthetic traffic profiles which can represent a class of applications may be used. This suggests that the use of both realistic and synthetic traffic profiles forms a complete set for the evaluation of the techniques proposed for NoC systems. Next, synthetic traffic models are introduced and their features including the application that they may represent are mentioned.

2.2 Synthetic Traffics

Different synthetic traffic patterns have been used for evaluating interconnection networks. Uniform, Transpose, Bit-Complement, Bit-Reversal, Hotspot [7], and Self-similar [10] are the most widely used traffic models for the analysis of power and delay in interconnection networks.

To describe the synthetic patterns, let each node (x, y) in the NoC design be labeled with an address resulting from the concatenation of x and y indicates of the node. The m-bit binary number representation of xy is $n_1n_2\ldots n_{m-1}n_m$. Also, let $\bar{0} = 1$ and $\bar{1} = 0$.

- Uniform Traffic: Each node sends messages to other nodes with an equal probability (i.e., destination nodes are chosen randomly using a uniform probability distribution function).
- Hot-spot Traffic: Each node sends messages to other nodes with an equal probability except for a specific node (called Hotspot) which receives messages with a greater probability. The percentage of additional messages that a Hotspot node receives compared to the other nodes is indicated after the Hotspot name (e.g., Hotspot 10%).
- Transpose Traffic: Each node sends messages only to a destination with the upper and lower halves of its own address transposed. i.e., the destination whose address is given by ($n_{m/2}n_{(m/2)+1}$ … $n_mn_1n_2$ … $n_{(m/2)-1}$).
- Complement Traffic – Each node sends messages only to a with One's complement of its own address, i.e., the destination whose address is given by ($\bar{n}_1\ \bar{n}_2\ \bar{n}_3$ … $\bar{n}_{m-2}\ \bar{n}_{m-1}\ \bar{n}_m$).
- Bit reversal Traffic: Each node sends only to a whose address is bit reversal of the sender's address, i.e., the destination with address ($n_m\ n_{m-1}\ n_{m-2}$ … $n_3\ n_2\ n_1$).

The uniform traffic model is a standard benchmark used in network routing studies. This model can

be considered as the traffic model for of well-balanced shared memory computations. In the Hotspot traffic pattern, one or more nodes are designated as the hot spot nodes, which receive Hotspot traffic in addition to the regular traffic. Therefore, the Hotspot node represents a very busy node. For example, in multiprocessors, Hotspot nodes could be the traffic representative of computations in which there are critical sections or shared/replicated data with more packet exchanges. For the transpose traffic, two types of patterns are proposed. With the first transpose traffic pattern, a node (i, j) only sends messages to node ($n - j$, $n - i$) where n is the network dimension (e.g., $n \times n$ in the mesh topology). This traffic pattern is very similar to the matrix-transpose. In the second transpose traffic pattern, a node (i, j) only sends messages to node (j, i). The bit-complement, reversal, and transpose traffics can model traces of applications related to numerical computations [7].

New synthetic traffic patterns may be inspired by analyzing the traffic patterns in a class of applications. As an example, in [10], a self-similarity concept is utilized to propose a new traffic pattern. The objective of the work was to introduce self-similarity as a fundamental property of the bursty traffic patterns flowing between the modules in typical MPEG-2 video applications. This property was inferred by examining the extracted traces when common video statistical tests were performed on the chip.

The above discussion shows that each synthetic traffic model is useful for certain classes of applications. Next, we discuss the motivation for another synthetic traffic profile which may be a representative of a broader class of NoC applications.

2.3 Motivation for NED

An important state in design of NoC systems is to map an application onto the cores existing on the chip. The mapping is an optimization problem typically with the objective of minimizing the total power consumption and propagation delay of the communication on the chip. Several research works have been focused on the application mapping onto NoCs (see, e.g., [31][32][33]). The power

consumption as well as the delay for each data communication operation is minimized by lowering the number of hops and shortening the total physical distance between the source and destination cores. As a result of using these mapping algorithms, an application should be mapped among different cores such that cores with a higher communication volume are mapped as close to one another as possible. For these networks, the closer the nodes are, the more packets they send to each other. An example of this situation is depicted in Fig. 1 which shows the number of packets sent from Node $X_{3,3}$ to other nodes. For this case, a total of 1,000,000 packets have been considered. For a more accurate evaluation of these networks, a synthetic traffic pattern with this preferred neighbor property should be used. Most of the above synthetic traffic profiles do not have this property, and hence, a new traffic profile is required.

3 NED TRAFFIC MODEL

In a network, a source node S that is located at position (a, b) is referred to as $S = X^*_{a,b}$. In addition, other nodes that are placed at position (i, j) are referred to as $X_{i,j}$. Assuming a mesh topology for now, the *Distance* matrix, R, for the source node is defined as

$$R_{n\times n} = [r_{i,j}]$$

where $r_{i,j}$ is the distance (number of hops) between S and $X_{i,j}$ given by

$$r_{i,j} = |i + j - (a + b)| \quad (1)$$

Using $R_{n\times n}$ and setting $k = |\max r_{i,j}|$, entries of the Distance Frequency (DF) vector, $D_{1,k}(n)$ for source node S may be defined as follows:

$$d_{1\times k}(n) = \sum_{j=1}^{n} \sum_{i=1}^{n} \frac{r_{i,j} = k}{k} \quad (2)$$

The j^{th} column of this matrix indicates the number of nodes in the network that have a distance of j from S. Fig. 2 shows R, and $D_{1,6}(4)$ for a 4×4 network with the source node of $X^*_{1,1}$.

We are looking for a Probability Distribution Function (PDF) that computes the probability of sending from the source node to other nodes with the property that the probability decreases as the distance between the source and destination nodes increases. In addition, PDF should be dependent on the source position. The reason is that the longest distance between a source and other nodes depends on the source position. In addition, the number of nodes that have a specific distance from a source is different for different source positions.

Fig. 3 shows the distance matrices for $S = X^*_{1,1}$ and $S = X^*_{3,3}$ in a 5×5 network. As shown in Fig. 3(a), four nodes have a distance of one from the source node while the longest distance to the source is 4. On the other hand, in Fig. 3(b), only two nodes have a distance of one from the source node while the longest distance to the source is 8. The PDF should have the general properties of probability distribution functions as well. If we denote P_r as the probability of sending a packet to a destination with the distance r from the source, and D is the set of all the distances from the source, then $0 < P_r < 1$ and $\sum_D P_r = 1$. Note that the proposed distribution function may be used for other topologies such as Torus, Hypercube, and 3D Torus. The reason for this is that this distribution function is only dependent on the distance between the source and destination.

In this work, we are looking for a probability distribution function in which the value of P_r decreases exponentially with increasing r. Denoting P_1 by P, we propose the following probability function:

$$P_r = P^{m^{(r-1)}} \times P \quad (3a)$$

or

$$P_r = P^{(m(r-1))+1} \quad (3b)$$

Here m is a parameter between 0 and 1.

Based on the central limit theorem [34], the sum of a large number (practically 30 or more) of independent and identically distributed random variables will have approximately a normal distribution. For a normal distribution function, the area under the function in the range $\mu \pm 3\delta$ is

97.65% of the total area. Based on this, we suggest that a suitable value for the parameter *m* be the one that results in a traffic pattern in which the length of the longest path is equal to μ+3δ. Thus, the value of the parameter *m* is chosen to be $m = \frac{1}{n}$.

As an example, in Fig. 4, the probabilities of sending a packet from $S = X^*_{1,1}$ to nodes with different distances in a 4×4 network for *m* = 3, 2 , 0.1 and 1/4 are shown. Using $\sum_D P_r - 1$, we can write

$$\sum_{i=1}^{k} (r_{1,1}(n) \times P_i) = 1 \quad (4)$$

For the example shown in Fig. 2, we have

$$2P_1 + 3P_2 + 4P_3 + 3P_4 + 2P_5 + P_6 = 1 \quad (5)$$

Putting $P_r = P \times P^{(r-1)m}$ in (7) leads to

$$2P + 3P^{m+1} + 4P^{2m+1} + 3P^{3m+1} + 2P^{4m+1} + P^{5m+1} = 1 \quad (6)$$

Assuming *m* = 1/4 and $Q = P^{1/4}$ ($P = Q^4$), we have

$$2Q^4 + 3Q^5 + 4\,Q^6 + 3Q^7 + 2Q^8 + Q^9 = 1 \quad (7)$$

The value of Q may be found by solving the above equation numerically. Notice that there is only one solution between 0 and 1 for the class of the above equations. Therefore, the solution is unique, a general result which is formulized using the following theorem.

Theorem: For the equation given by

$$Y = aQ^m + bQ^{m+1} + cQ^{m+2} + \ldots + zQ^{m+n} - 1 \quad (8)$$

where the coefficients *a* to *z* are integers greater than or equal to zero, there is a unique solution between 0 and 1.

Proof: The derivative of *Y* with respect to *Q* is given by

$$Y' = maQ^{m-1} + (m+1)bQ^m + (m+2)cQ^{m+1} + \ldots + (m+n)zQ^{m+n-1} \quad (9)$$

where Y' is a continuous function on the set of real numbers. On the other hand, if $Q > 0$, Y' is also greater than zero. Therefore, Y is strictly increasing in the range of 0 and 1. Since the value of Y for $Q = 0$ and 1 are -1 and $a + b + c + d + \ldots + z - 1 > 0$, respectively, and Y is continuous and strictly increasing, therefore, there is exactly one point in the range of 0 and 1 at which $Y = 0$.

Solving Equation (7) leads to $Q = 0.6247$. The probability distribution diagram for $S = X^*_{1,4}$ is shown in Fig. 5.

The computation for a network with a specific dimension is performed only once. In addition, due to the symmetry, there are nodes with the same distances, the computation should be only done for about one fourth of the nodes. This is shown in Fig. 6 for 4×4 and 5×5 meshes. Fig. 7 shows the probabilities of receiving packet for three different positions of source node in 5×5 mesh size ($n = 1/5$). To illustrate the probability distribution in a 3D mesh under the NED traffic model,Fig. 8 illustrates a case when the source node is in the center of a 3×3×3 mesh ($S = X^*_{2,2,2}$).

It should be noted that for ring, 2D, and 3D torus topologies, the NED traffic results in less computational complexity than that of 2D mesh. Fig. 9 shows ring, 2D torus, and 3D torus topologies. In these networks, the computations are done just for one node in a network with any dimension. To elaborate on this point, let us define $n(d)$ for the source node S as the number of nodes with the a *Manhattan distance* of d hops from S. Table 1 shows mathematical expressions of $n(d)$ for various network topologies. These expressions reveal that, regardless of the source position, the model of adjacent nodes is similar and is a function of d. As an example, matrices of Fig. 3 for a mesh topology are repeated for the same nodes in a 5×5 torus topology and shown in Fig. 10. Distance matrices of these two figures are equal regardless of the fact that the position of the source is changed.

Finally, it should be noted that for heterogeneous networks [5] with any dimension, the NED traffic model may be used. Fig. 11 shows an example of this type of networks for a 2D semi-mesh NoC, and illustrates its Distance matrix and Distance Frequency vector.

4 RESULTS AND DISCUSSION

To evaluate the efficacy of using the NED traffic profile in evaluating NoCs, we compared the average hop counts of all packets transported based on different traffic profiles. In addition to NED, other traffic models included Transpose, Uniform, Hotspot 5%, Hotspot 10%, Hotspot 20%, and Bit-complement, and profiles generated based on some realistic applications mapping on an $n \times n$ mesh. For the Hotspot synthetic traffic profile, the hotspot point was chosen to be the node $(\lceil n/2 \rceil, \lceil n/2 \rceil)$. For mapping realistic applications on $n \times n$ meshes, we used *EvoMP* [35]. *EvoMP* is a mapping tool that maps DSP or similar applications onto distributed-control multiprocessor system on chip (MPSoC.) This tool has run-time task decomposition and scheduling capabilities. Each computational cell in this platform is a special processor which can be configured to 8, 16, or 32 bit mode. These cells are placed in a 2D-mesh topology and uses NoC scheme for communication. The size of each dimension of mesh can be configured independently.

Table 2 shows the average hop counts of synthetic and realistic traffic profiles on networks with different mesh sizes. For all the switches, the data width was set to 32-bits. Each input virtual channel had a buffer (FIFO) with the size of six flits. In all the simulations, the latency was measured by averaging the latency of the packets when each local core generated 30,000 packets. The router used the minimally fully adaptive reserved virtual channel (VC) deadlock avoidance technique discussed in [36]. As this table shows, compared to other synthetic traffic models, the average hop count of NED is more similar to those of realistic traffic patterns. Other than NED, Uniform and Hotspot (which is a specific kind of uniform traffic) are among the synthetic traffic model with good hop count match to those of the realistic benchmarks. Even for these traffic profiles, the difference between their average hop counts and the realistic ones is larger than that of NED.

In the case of the semi realistic benchmark of "GSM + Uniform" [11], the difference is less. In this benchmark, just a few cores generate packets based on the GSM voice codec and the remaining cores are sent packets based on the Uniform traffic profile. For other realistic benchmarks, the NED traffic

has closer average packet hops to those of these benchmarks. In addition, the difference of NED and other synthetic traffic profiles increases as the dimension of the network increases. Note that an empty slot means that the mesh size is too large for the application. Fig. 12 shows the average hop counts of different synthetic traffic models for different mesh sizes. The rate of increase in the average hop count in NED with the network size is lower than those of other synthetic traffic model. This behavior makes NED resemble more realistic traffics.

In the second set experimental results, we have compared the XY routing algorithm [1][2] under Uniform, NED, and Real 128-point FFT traffic profiles. The comparison is performed in terms of the power and latency for these traffic profiles where the simulations were performed for a 4×4 mesh NoC. The performance of the network is evaluated using latency curves as a function of the packet injection rate (*i.e.*, the number of packets injected into the network per cycle). The packet latency is defined as the time duration from when the first flit is created at the source core to when the last flit is delivered to the destination core. It was assumed that the packets had a fixed length of five flits, the buffer size of each virtual channel was five flits, and the data width was set to 32 bits.

The NoC performances for the XY routing algorithm under uniform, NED, and 128-Point FFT traffic profiles are given in Fig. 13 (a). As seen in the figure, the pair of XY-NED and XY-FFT-128 have almost the same performance at any traffic loads. As the load increases, the packet latency rises dramatically due to the network congestion. The results show that the XY-Uniform curve does not have the same performance in comparison with the XY-NED and XY-FFT-128 particularly at high traffic loads. This is caused mainly by the high average hop count of the uniform traffic. The power consumptions of the XY-switch under these three traffic profiles which are computed by Synopsis Power Compiler using a the 0.13μm standard CMOS technology are presented in Fig. 13 (b) . As the results reveal, the average power consumptions graph confirms the similarity of NED traffic profile to the real application traces.

5 CONCLUSION

In this work, a synthetic traffic profile based on Negative Exponential Distribution (NED) for network on chips was proposed. In this traffic profile, the probability of sending a packet from a source to a destination decreases exponentially as the distance between them increases. This property made NED more similar to traffic profiles of real applications where the cores with higher packet communication loads were mapped closer to each other to minimize the communication delay and power consumption. To show this property for NED, the average packet hops for some synthetic and realistic traffic profiles were compared. The results showed similarity of NED traces with those of realistic applications running on NoCs with different sizes.

ACKNOWLEDGEMENT

A.-M. Rahmani and A. Afzali-Kusha acknowledge the financial support by the Research Council of the University of Tehran.

REFERENCES

[1] L. Benini et al., "Networks on chips: A new SoC paradigm," IEEE Computer (**2002**), vol. 35, pp. 70–78.

[2] A. Jantsch et al. (Eds.), Networks on Chip, Kluwer (**2003**).

[3] W. J. Dally et al., “Route Packets, Not Wires: On-Chip Interconnection Networks,” Proceedings of International Conference on Design Automation (**2001**), pp. 684–689.

[4] Paul Gratz et al., “Regional Congestion Awareness for Load Balance in Networks-on-Chip,” Proceedings of International Symposium on High-Performance Computer Architecture (**2008**), pp. 203-214.

[5] T. Bjerregaard et al., “A Survey of Research and Practices of Network-on-Chip,” ACM Computing Surveys (**2006**), Vol. 38, No. 1, pp. 1-51.

[6] A. M. Rahmani et al., “Forecasting-based Dynamic Virtual Channels Allocation for Power Optimization of Network-on- Chips,” Proceedings of International Conference on VLSI Design (**2009**), pp. 151-156.

[7] M. L. Fulgham et al., “Performance of Chaos and Oblivious Routers under Non-Uniform Traffic,” Technical Report UW-CSE-93-06-01 (**1993**), Univ. of Washington.

[8] K. Lahiri et al., “Evaluation of the traffic-performance characteristics of system-on-chip communication architectures,” Proceedings of the International Conference on VLSI Design (**2000**), pp. 29–35.

[9] W. J. Dally et al., Principles and practices of interconnection networks. Morgan Kaufmann Publishers (**2004**).

[10] G. V. Varatkar et al., “On-chip traffic modeling and synthesis for MPEG-2 video applications,” IEEE Transactions of Very Large Scale Integration (VLSI) Systems (**2004**), Vol. 12, No. 1, pp. 108-119.

[11] P. Doukhan et al., Theory and applications of long-range dependence. Birkhäuser, ISBN: 0817641688. Switzerland (**2002**).

[12] K. Park et al., Self-similar network traffic and performance evaluation. John Wiley and Sons, ISBN: 0471319740 (**2000**).

[13] W. E. Leland et al., “On the self-similar nature of Ethernet traffic (extended edition),” IEEE/ACM Transactions on Networking, Vol. 2, No. 1 (**1994**), pp. 1-15.

[14] V. Paxson et al., “Wide-area traffic: the failure of Poisson modeling,” IEEE/ACM Transactions on Networking, Vol. 3, No. 3 (**1995**), pp. 226-244.

[15] A. Balachandran et al., “Characterizing user behavior and network performance in a public wireless LAN,” Proceedings of the ACM SIGMETRICS international conference on Measurement and modeling of computer systems (**2002**), pp. 195-205.

[16] I. Y. Bucher et al., “Models of access delays in multiprocessor memories,” IEEE Transactions on Parallel and Distributed Systems (**1992**), Vol. 3, No. 3, pp. 270-280.

[17] F. Darema-Rogers et al., “Memory access patterns of parallel scientific programs,” ACM SIGMETRICS Performance Evaluation Review (**1997**), Vol. 15, No. 1, pp. 46-58.

[18] S. Turner, “Performance analysis of multiprocessor interconnection networks using a burst-traffic model,” Ph.D. Dissertation Thesis, University of Illinois at Urbana-Champaign (**1995**).

[19] G. De Micheli et al., “Networks on Chips: Technology and Tools,” Morgan Kaufmann (**2006**).

[20] P. Kongetira et al., “Niagara: a 32-way multithreaded Sparc processor,” IEEE Micro, Vol. 25, No. 2 (**2005**), pp. 21-29.

[21] The Power6 Architecture. Available [online]: http://www.ibm.com.

[22] R. Rajwar et al., “Virtualizing transactional memory,” Proceedings of the 32nd International Symposium on Computer Architecture (**2004**), pp. 494-505.

[23] K. Sankaralingam et al., “Exploiting ILP, TLP, and DLP with the polymorphous TRIPS architecture,” Proceedings of the 30th International Symposium on Computer Architecture (**2003**), pp. 422-433.

[24] M. B. Taylor et al., “Evaluation of the Raw microprocessor: an exposedwire-delay architecture for ILP and streams” Proceedings of the 31st International Symposium on Computer Architecture (**2004**), pp. 2-13.

[25] A. Jalabert et al., "×pipesCompiler: a tool for instantiating application specific networks on chip," Proceedings of the Design, Automation and Test in Europe Conference and Exhibition (**2004**), Vol. II, pp. 884-889.

[26] S. G. Pestana et al., "Cost-performance trade-offs in networks on chip: a simulation-based approach," Proceedings of the Design, Automation and Test Conference in Europe Conference and Exhibition (**2004**), Vol. II, pp. 764-769.

[27] D. Wu et al., "Improving Routing Efficiency for Network-on-Chip through Contention-Aware Input Selection," Proceedings of Asia and South Pacific Conference on Design Automation (**2006**), pp. 36-41.

[28] S.C. Woo et al., "The Splash-2 Programs: Characterization and Methodological Considerations," Proceedings of International Symposium on Computer Architecture (**1995**), pp. 24-36.

[29] C. Lee et al., "Mediabench: a tool for evaluating and synthesizing multimedia and communications systems," Proceedings of the International Symposium on Microarchitecture (**1997**), pp. 330-335.

[30] The Standard Performance Evaluation Corporation. Available [online]:http://www.spec.org/.

[31] M. Nickray et al., "Power and Delay Optimization for Network on Chip," Proceedings of European Conference on Circuit Theory and Design (**2005**), pp. 277-281.

[32] A. Mehran et al., "A Heuristic Energy Aware Application Mapping Algorithm for Network on Chip," Proceedings of IP Based SoC Design Conference & Exhibition (**2006**), pp. 289-294.

[33] S. Murali et al., "Bandwidth-Constrained Mapping of Cores onto NoC Architectures," Proceedings of Design, Automation and Test in Europe (**2004**), pp. 896-901.

[34] Robert V. Hogg et al., "Introduction to Mathematical Statistics," Prentice Hall (**2005**).

[35] Sh.Vakili et al., “EvoMP: a novel MPSoC architecture with evolvable task decomposition and scheduling,” will be appear in IET Computers & Digital Techniques (**2009**).

[36] L. M. Ni et al., “A survey of wormhole routing techniques in direct networks,” IEEE Computer (**1993**), Vol. 26, No. 2, pp. 62–76.

FIGURES AND TABLES

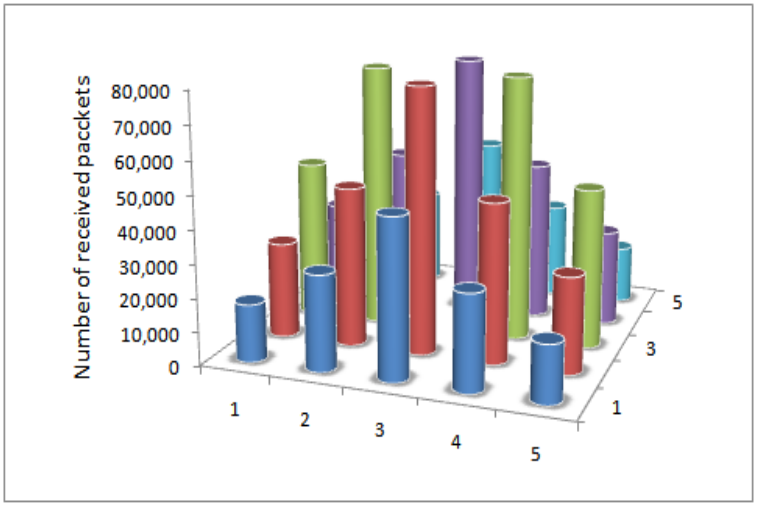
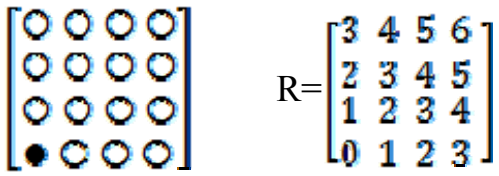


Fig. 1 Number of messages sent by Node (3, 3) to other nodes in a 5×5 mesh



$D_{1*6}(4) = [2\ 3\ 4\ 3\ 2\ 1]$, $k = 6$

Fig. 2 Distance matrix(*R*), Distance Frequency vector (*D*).

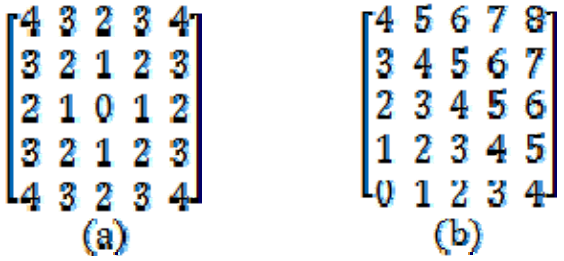


Fig. 3 (a) Distance matrix for $S = X^*_{1,1}$,(b) Distance matrix for $S = X^*_{3,3}$

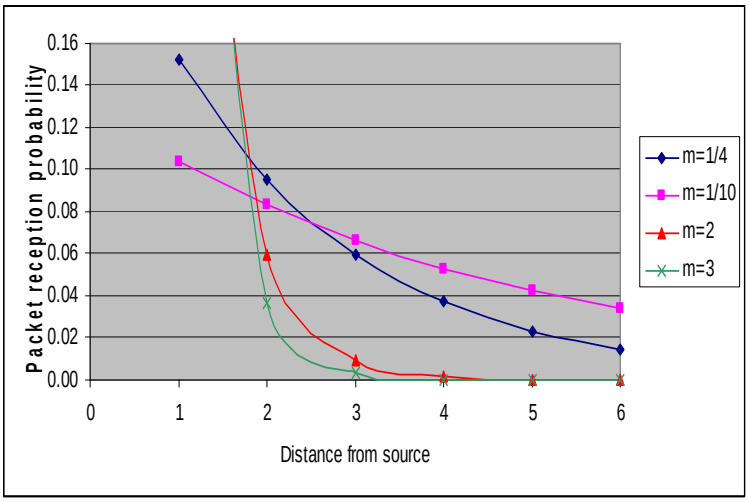


Fig. 4 Sending probability for m = 3, m = 2, m = 0.1, and m = 1/4

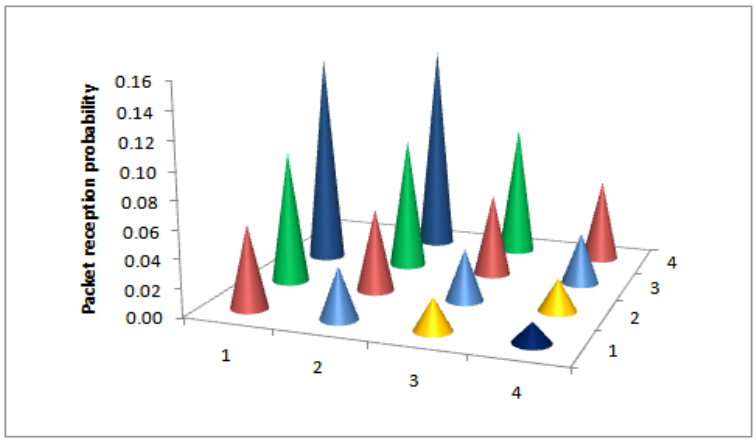
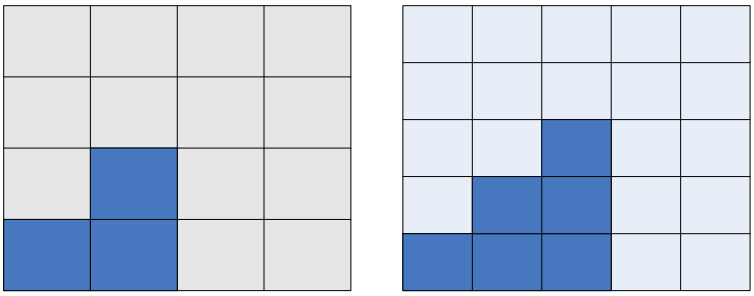


Fig. 5 Probability distribution diagram for $S = X^*_{1,4}$ for a 4×4 mesh network.



(a) (b)

Fig. 6 The nodes that require specific computation for (a) 4×4 and (b) 5×5 meshes.

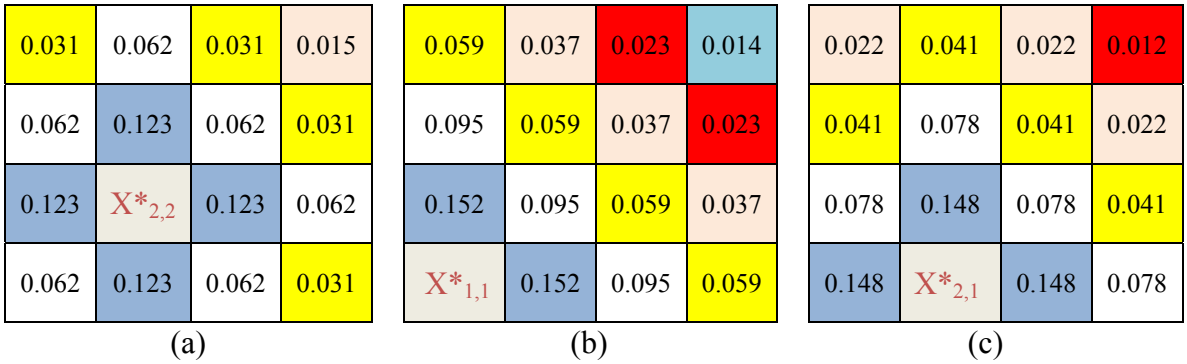


Fig. 7 The probabilities of receiving packets for three different (main) positions of source node in a 5×5 mesh NoC (a) $S = X^*_{2,2}$ (b) $S = X^*_{1,1}$ (c) $S = X^*_{2,1}$

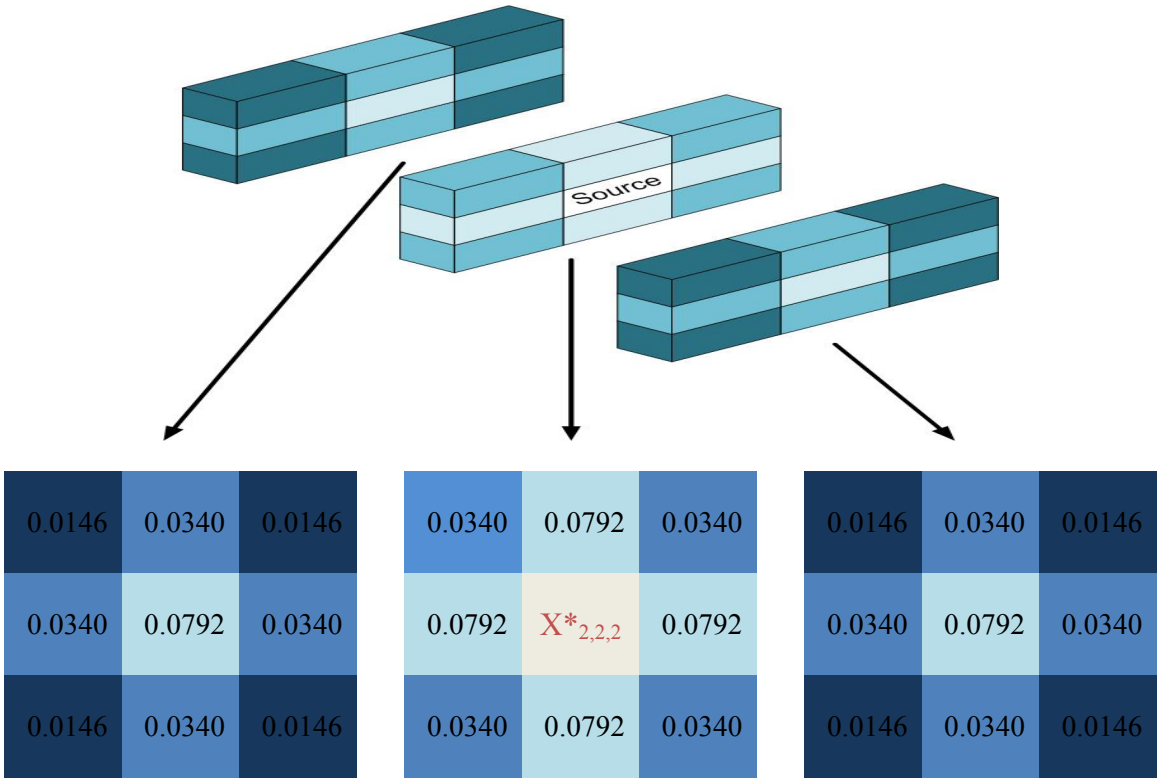


Fig. 8 The probabilities of receiving packet in a 3D 3×3×3 mesh NoC when source node is in the center of network

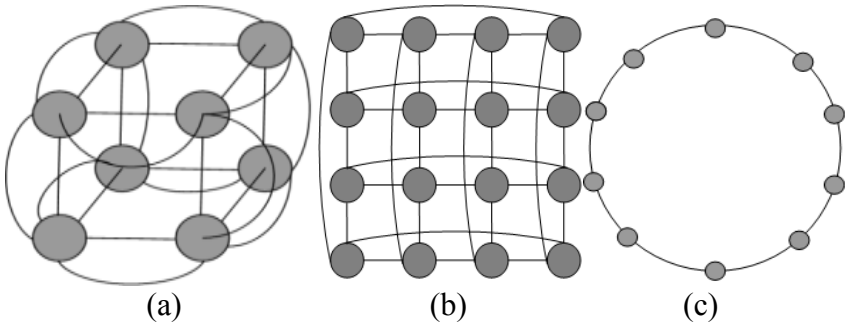
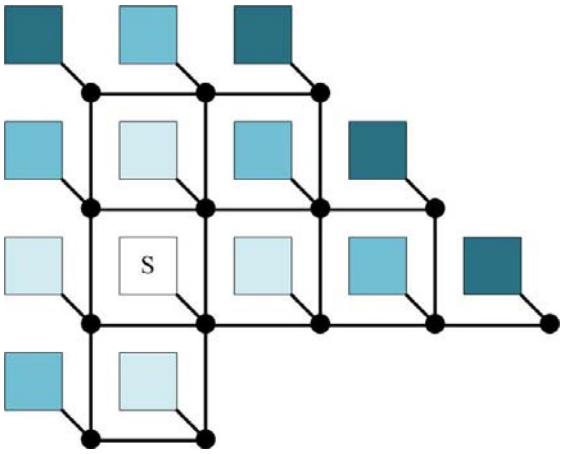


Fig. 9 (a) 3-dimensional torus (b) 2-dimensional torus (c) Rings (1-dimensional) network-on-chip topologies.

$$\begin{bmatrix} 4 & 3 & 2 & 3 & 4 \\ 3 & 2 & 1 & 2 & 3 \\ 2 & 1 & 0 & 1 & 2 \\ 3 & 2 & 1 & 2 & 3 \\ 4 & 3 & 2 & 3 & 4 \end{bmatrix} \quad (a) \qquad \begin{bmatrix} 1 & 2 & 3 & 3 & 2 \\ 2 & 3 & 4 & 4 & 3 \\ 2 & 3 & 4 & 4 & 3 \\ 1 & 2 & 3 & 3 & 2 \\ 0 & 1 & 2 & 2 & 1 \end{bmatrix} \quad (b)$$

Fig. 10 (a) Distance matrix for S = $X^*_{1,1}$, (b) Distance matrix for S = $X^*_{3,3}$ for torus topology. For this case, $D_{1*4}(5)$ = [4 8 8 4], k = 4.



(a)

$$R=\begin{bmatrix} - & - & - & - & - \\ 3 & 2 & 3 & - & - \\ 2 & 1 & 2 & 3 & - \\ 1 & 0 & 1 & 2 & 3 \\ 2 & 1 & - & - & - \end{bmatrix}$$

(b)

$D_{1*6}(4) = [4\ 5\ 4]$, $k = 3$

(c)

Fig. 11 (a) A heterogeneous semi-mesh network when $S = X_{2,2}$, (b) its Distance matrix(R), (c) its Number of Distance matrix

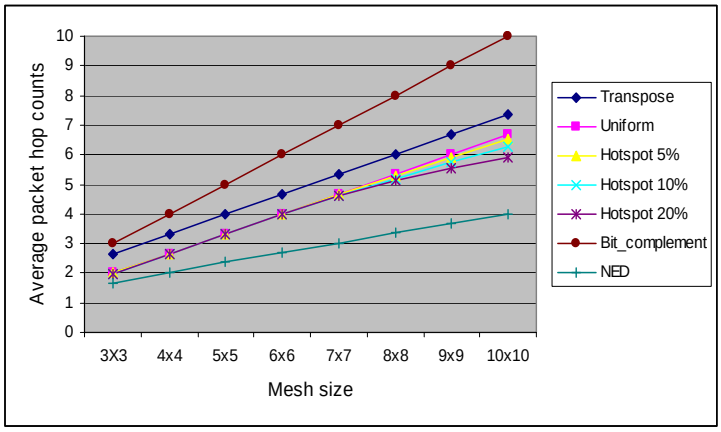
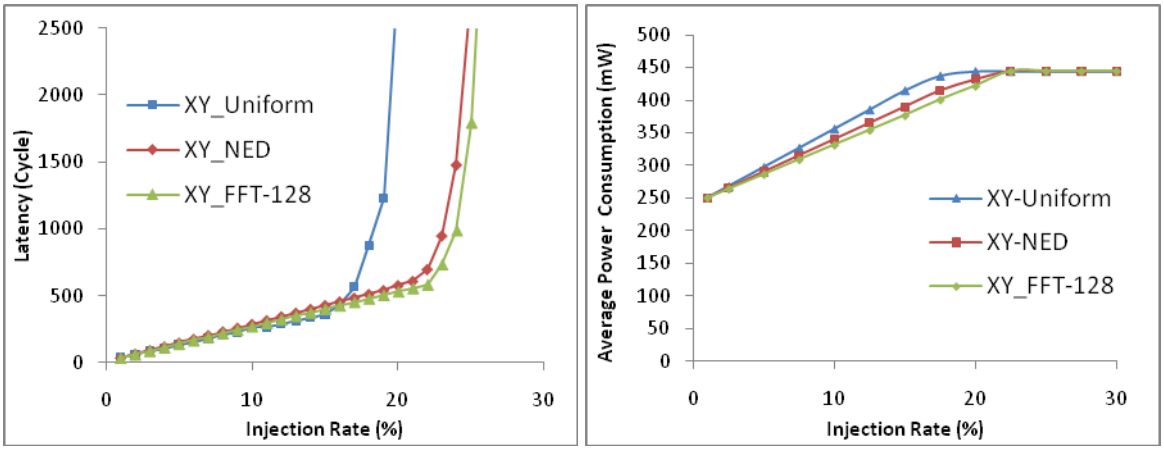


Fig. 12 Average hop counts of different synthetic traffics for different mesh sizes.



(a) (b)

Fig. 13 (a) Load-latency (b) Load-Power graph of XY routing algorithm for a 4×4 2D mesh

Table 1 n(d) for major network topologies.

Topology	$n(d)$
Rings (1-dimensional)	2
2-dimensional torus	$4d$
3-dimensional torus	$4d + 2 + 8\sum_{a=1}^{d}(d-a)$

Table 2 Average packet hops for some synthetic and realistic traffic patterns

Traffic Type	Traffic Pattern	3×3	4×4	5×5	6×6	7×7	8×8	9×9	10×10
Synthetic	*Transpose*	2.667	3.333	4	4.667	5.333	6	6.667	7.333
	Uniform	2	2.667	3.333	4	4.667	5.333	6	6.667
	Hotspot 5%	1.997	2.663	3.332	3.991	4.64	5.282	5.917	6.518
	Hotspot 10%	1.994	2.663	3.331	3.983	4.619	5.201	5.761	6.251
	Hotspot 20%	1.988	2.66	3.325	3.972	4.598	5.107	5.531	5.886
	Bit complement	3	4	5	6	7	8	9	10
	NED (m = 1/n)	**1.652**	**2.034**	**2.399**	**2.692**	**3.026**	**3.359**	**3.693**	**4.015**
Realistic	*GSM+Uniform*	-	2.128	2.765	3.177	3.647	-	-	-
	Order 16 FIR	1.455	1.93	2.454	-	-	-	-	-
	Order 24 FIR	-	2.121	2.478	3.586	-	-	-	-
	Order 64 FIR	-	2.22	2.493	3.592	-	-	-	-
	Order 128 FIR	-	2.271	2.503	3.595	-	-	-	-
	5×5 Matrix Multiplication	1.235	1.915	2.775	3.032	-	-	-	-
	7×7 Matrix Multiplication	1.575	2.156	2.558	2.895	3.289	-	-	-
	802.11a	-	2.199	2.567	2.867	-	-	-	-
	ADPCM	-	1.995	2.232	2.991	-	-	-	-
	128-point FFT	1.388	1.867	2.327	-	-	-	-	-
	256-point FFT	1.599	2.09	2.445	2.893	-	-	-	-
	512-point FFT	-	2.341	2.65	2.93	3.43	-	-	-
	8-points DCT	1.596	1.895	2.043	-	-	-	-	-
	16-points DCT	-	2.252	2.45	2.822	-	-	-	-
	32-point DCT	-	2.257	2.542	2.88	-	-	-	-