Hybrid Network on Chip (HNoC): Local Buses with a Global Mesh Architecture

Payman Zarkesh-Ha  
University of New Mexico  
Department of ECE  
Albuquerque, NM, USA  
+1 (505) 277-6724  
payman@ece.unm.edu

George Bezerra  
University of New Mexico  
Department of CS  
Albuquerque, NM, USA  
+1 (505) 277-3411  
gbezerra@unm.edu

Stephanie Forrest  
University of New Mexico  
Department of CS  
Albuquerque, NM, USA  
+1 (505) 277-7104  
forrest@cs.unm.edu

Melanie Moses  
University of New Mexico  
Department of CS  
Albuquerque, NM, USA  
+1 (505) 277-9140  
melaniem@unm.edu

ABSTRACT

Network on chip (NoC) is often implemented with only packet-based communication rather than bus connections between cores. Although NoC is a good solution for long-distance communication, local buses are more efficient for short-distance connections. In this paper, we propose a hybrid network on chip (HNoC) fabric that uses local buses for nearest-neighbor communication and the standard NoC topology for global interconnection. Local buses carry all the nearest-neighbor traffic, reducing traffic on the global network, which results in increased throughput and reduced energy consumption.

Based on a communication probability density (CPD) function derived from Rent's rule, it is shown that in a 25-processor chip multiprocessor, HNoC can remove up to 78% of the traffic from the global NoC topology, which results in 4.6x higher throughput and a 58% reduction in energy consumption compared to a conventional NoC topology.

Categories and Subject Descriptors

General Terms
Design, Performance, Theory.

Keywords
Stochastic model, hybrid network on chip, throughput, network traffic, local buses, global mesh.

1. INTRODUCTION

Microprocessor performance has increased exponentially over the last four decades as advancing semiconductor technology has vastly increased the quantity and improved the speed of on-chip transistors available to circuit designers. Traditionally, computer designers took advantage of these resources to improve uniprocessor structures because of its simpler programming model compared to systems with distributed structures [1]. Recently, however, power consumption and wire delay have limited the continued scaling of uniprocessor systems, while making chip multiprocessor architectures increasingly popular [2]. In addition, network on chip (NoC) has become the emerging paradigm for communications within large chip multiprocessor systems to overcome certain issues with global interconnects, such as scalability, power, and delay.

Many different NoC topologies for multiprocessor systems have been proposed and studied by researchers. For instance, in [3], Balfour and Dally published a comprehensive analysis of several possible NoC topologies, such as: mesh, torus, fat tree, concentrated mesh, concentrated torus, and tapered fat tree. Of these networks, the best topology in terms of energy and communication time was concentrated mesh, a type of mesh topology that uses larger-radix routers to cluster four processors at each mesh node and contains express channels around the perimeter of the network. Figure 1 illustrates three examples of NoC topologies.

To further optimize NoC performance and energy, a hybrid optical/electrical NoC architecture was recently proposed [4-6], where a photonic network and an electronic network coordinate to provide the system with high bandwidth communications. The optical circuit switching network handles long-lived bulk data transfers, whereas the secondary lower-bandwidth electronic packet switching network accommodates collectives and short-lived data exchanges. Although in this approach a high bandwidth optical network and a low cost electrical network are hybridized together to improve the energy and performance, the communication is still purely packet based.

![Figure 1. Three examples of NoC topologies [3].](image-url)
Finally, some discussions and conclusions are provided in Section 6 and 7, respectively.

HNoC versus conventional NoC. Some simulation results using a commercial NoC simulator are then presented in Section 5.

Optical/electrical NoC architecture that is purely packet-based, our HNoC uses local buses to transmit data directly to the nearest neighbors in a parallel fashion, which eliminates the need for serializer, router, and deserializer. Moreover, since the local bus interconnects length are short, they inherently provide higher bandwidth and consume lower power.

Section 2 gives details of the proposed HNoC architecture. In Section 3, we use a communication probability density (CPD) model derived from Rent’s rule to assess traffic patterns for conventional NoC. In Section 4, the Rent’s rule-based CPD is used to predict the performance and energy consumption of HNoC versus conventional NoC. Some simulation results using a commercial NoC simulator are then presented in Section 5. Finally, some discussions and conclusions are provided in Section 6 and 7, respectively.

2. HYBRID NoC (HNoC) TOPOLOGY

Today’s chip multiprocessor interconnects use packet-switch networks to connect the cores. Inter-processor messages are broken into packets that are routed through the NoC switches. However, as the number of cores increases, performance and power consumption of NoC degrade significantly due to higher communication costs [6]. To find a solution, let’s review the communication cost in a packet-switch network. To transfer data from one core to another through NoC fabric, let's review the Rent’s rule application on digital systems. A multiprocessor system can be seen as a digital circuit, where a logic gate is a representation of a processor core. By empirical observation and from the Rent’s rule based wire-length distribution, we find that the majority of interconnections are short. This is true because circuit designers tend to connect blocks that are closer together. Similarly, programmers try to map tasks with highest communication rate to the processors that are closer together in a multiprocessor system. It can be therefore expected that the majority of communications within a multiprocessor NoC system are for the close neighbors. Utilizing Rent’s rule, we use communication probability distribution model to quantify the advantageous of HNoC versus conventional NoC topologies.

3. COMMUNICATION PROBABILITY DISTRIBUTION (CPD) MODEL

For the first time, a statistical analysis for NoC traffic pattern was presented by Soteriou et. al. [7] in 2006. Then Greenfield et al. [8] applied the principle of Rent’s rule to the analysis of NoC architecture and showed that for a VLSI design consisting of many blocks wired together, if one replaced these wires with a NoC, then the Rent’s terminal-exponent of the former may match the bandwidth-exponent of the latter, or [8]:

$$B = b N^p,$$

(1)

where \( B \) is the communication bandwidth, \( N \) is the number of nodes, and \( b \) and \( p \) are the Rent’s coefficient and exponent, respectively.

It was also shown in [8] that Rent’s rule can also be used to characterize NoC architecture similar to interconnect modeling in VLSI designs. For instance, hop-length distribution can be directly derived from wire-length distribution used in VLSI designs [8]. Later, Heirman et al. [9] validated Rent’s rule by analyzing the SPLASH-2 benchmarks. They further analyzed temporal behavior of network traffic using Rent’s rule and confirmed that Rent’s rule and all of its applications to VLSI can also be applied to NoC topologies.

Recently, Bezerra et al. [10] presented a closed form model, communication probability Distribution (CPD), as another derivative of Rent’s rule for multiprocessor systems. CPD is the probability that a processor communicates with another processor at distance \( d \) in a chip multiprocessor system. The communication probability of distance \( d \) for an \( N \times N \) multiprocessor system is determined by [10]:

$$CPD(d) = \frac{\Gamma(f(d))}{d^{2}} \left[ \frac{1 + d(d - 1)^p}{1 + [d + 1]^p} \right],$$

(2)

where \( p \) is the Rent’s exponent, \( \Gamma \) is the normalization coefficient such that \( \sum_{d=1}^{2N^2} CPD(d) = 1 \), and \( f(d) \) is given by:

$$f(d) = \begin{cases} \frac{d^3}{3} - 2d^3N + \frac{d}{3} (6N^2 - 1), & 1 \leq d < N \\ \frac{d^3}{3} + 2d^3N - \frac{d}{3} (12N^2 - 1) + \frac{2}{3} N(4N^2 - 1), & N \leq d \leq 2N - 2 \end{cases}$$

To assess the benefits of HNoC versus conventional NoC fabric, let’s review the Rent’s rule application on digital systems. A multiprocessor system can be seen as a digital circuit, where a logic gate is a representation of a processor core. By empirical observation and from the Rent’s rule based wire-length distribution, we find that the majority of interconnections are short. This is true because circuit designers tend to connect blocks that are closer together. Similarly, programmers try to map tasks with highest communication rate to the processors that are closer together in a multiprocessor system. It can be therefore expected that the majority of communications within a multiprocessor NoC system are for the close neighbors. Utilizing Rent’s rule, we use communication probability distribution model to quantify the advantageous of HNoC versus conventional NoC topologies.
An example of CPD for 5×5 arrays of multiprocessors is shown in Fig. 3, for the Rent’s exponent of 0.6. In this example, 78% of the communications are within the nearest neighbors.

The communication probability given in (2) can be considered as simply the normalized hop-distance distribution proposed in [8] or the normalized wire-length distribution proposed by [11]. However, unlike the hop-distance distribution, CPD can provide substantial information about the NoC system with only knowing the Rent’s exponent, p, and independent of the bandwidth Rent’s coefficient, b, since it is cancelled out by normalization.

Figure 4 illustrates the CPD distributions using various Rents’ exponent ranging from 0.1 to 0.9. As shown, Rent’s rule predicts that the majority of communications in a 5×5 array of multiprocessors are within the nearest neighbors.

4. HNoC VERSUS CONVENTIONAL NoC

In Section 3, we showed that the nearest neighbor traffic can be directly transmitted through local buses rather than the main NoC fabric. In this Section, we use Rent’s rule based CPD to quantify the advantages of HNoC versus conventional NoC.

4.1 Throughput Analysis

Consider the 5×5 array of multiprocessors shown in Fig. 3. If 78% of the communication can be moved to local buses, the NoC will be responsible for only 22% of the traffic. Therefore, the throughput of the NoC can potentially improve by 4.6x for the maximum injection rate. In general the rate of throughput improvement is determined by:

$$\frac{\text{HNOC}_{\text{Throughput}}}{\text{NOC}_{\text{Throughput}}} \approx \frac{1}{1 - \text{CPD}(1)},$$

where CPD is given by (2) and (3).

The improvement rate for various array sizes is shown in Fig. 5 assuming p=0.6. As shown in this figure, even in a large array size of 10×10 (i.e. 100 processors), the predicted improvement rate is 3.4x.

Note that the improvement rates given in (4) and shown in Fig. 5 are for an ideal case, where it is assumed that the local buses impose no overhead. Equation (4) presents an ultimate HNoC benefits without making too many assumptions, which may be application or design dependent. In practice, however, depending on the design and application, the local bus overhead will impact the performance of HNoC as will be presented in Section 5.

4.2 Energy Consumption Analysis

Similarly, energy consumption can be reduced by introducing local buses. Again, consider the 5×5 array of multiprocessor shown in Fig. 3. The CPD shown in Fig. 3 can be used to compute the energy reduction rate in HNoC. Using the NoC energy model presented in [10] and assuming that the power consumption of routers is dominant, the energy consumption of the HNoC against conventional NoC for the same throughput can potentially be reduced by factor of 2.4x. In general, for the same throughput the energy reduction rate in an array of N×N microprocessors is approximated by [10]:

$$\frac{\text{NOC}_{\text{Energy}}}{\text{HNOC}_{\text{Energy}}} \approx \frac{\sum_{d=1}^{2N} d \cdot \text{CPD}(d)}{\sum_{d=1}^{2N} d \cdot \text{CPD}(d)},$$

where the CPD is given by (2) and (3).
Using (5) and assuming that $p=0.6$, the projection of the best case energy reduction in HNoC versus conventional NoC is shown in Fig. 6. As shown in this figure, the energy reduction rate is about 1.8x for large array size of 10×10.

Similar to the throughput improvement analysis, the power improvement model given in (5) is for an ideal case, where it is assumed that the local buses impose no overhead. In practice, however, depending on the design and application, the local bus power overhead will impact the power consumption of HNoC as will be presented in Section 5.

4.3 Scalability Analysis

It is expected that the number of cores increases with technology scaling. HNoC is scalable and provides better power and performance communication even within a chip with large number of cores. Although the advantage of HNoC over conventional NoC shrinks by growing the number of cores, the rate of improvements in throughput and energy are still considerable even for large multiprocessor systems. Based on Fig. 5, the rate of throughput improvement of HNoC over NoC stays at about 3.5x, and based on Fig. 6, the rate of energy saving in HNoC over NoC stays at about 1.5x.

5. SIMULATION RESULTS

To verify our analysis in Section 4, the proposed HNoC was implemented in a system simulator, Orion 2 [12] and compared to the conventional NoC. The system parameters for this simulation are shown in Table 1.

As shown in Fig. 7, the benefit of HNoC is negligible when the injection rate is low. Once the injection rate increases beyond 0.2 packets per cycle, the traffic starts to saturate the throughput of conventional mesh NoC. However, HNoC can handle most of the traffic using the local buses and continue to provide more throughputs. Figure 7 shows that at the injection rate of 0.6 packets per cycle, HNoC provides 2.6x more throughput than conventional mesh NoC.

Without the local bus overhead, equation (4) predicts a 3.7x improvement in throughput. However, due to the limitation of local bus bandwidth and the injection rate constraint in this test system, the HNoC throughput improvement is reduced to 2.7x.

Table 1. System Parameters used in simulations

<table>
<thead>
<tr>
<th>System Parameters</th>
<th>Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Cores</td>
<td>64</td>
</tr>
<tr>
<td>Die Size</td>
<td>1cm x 1cm</td>
</tr>
<tr>
<td>Technology Node</td>
<td>45 nm</td>
</tr>
<tr>
<td>Clock Frequency</td>
<td>1 GHz</td>
</tr>
<tr>
<td>Flit Size</td>
<td>64 bits</td>
</tr>
<tr>
<td>Packet Size</td>
<td>5 flits</td>
</tr>
<tr>
<td>Rent's Exponent, $p$</td>
<td>0.60</td>
</tr>
</tbody>
</table>

Figure 8 illustrates the energy consumption per packet in HNoC and conventional mesh NoC. As shown in this figure, HNoC dissipates 1.8x lower energy per packet than conventional mesh NoC for the injection rate below 0.2 packets per cycle, when the HNoC and conventional NoC have the same throughput. The mesh NoC energy consumption per packet continues to increase beyond 0.2 packets per cycle because throughput saturations in conventional NoC.

Similarly, without the local bus overhead, equation (5) predicts a 1.9x improvement in power, which is close to the experimental results. This means that in this test system the local bus power overhead is negligible compared with the mesh NoC power consumption.
6. DISCUSSION

Rent’s rule arises in digital systems because the EDA tools optimize placement and routing in order to reduce wiring requirements and minimize the number of long wires. Similar to EDA tools, compilers need to be designed such that locality becomes the primary objective. Therefore, it becomes evident that a “smart compiler” with optimized program mapping and task assignment must be developed to get the best benefit of NoC architecture.

The purpose of the analysis presented in this paper is to support the concept of traffic localization as previously suggested by some researchers [13-15]. Once the traffic localization is met, the proposed HNoC architecture can significantly improve the energy and performance of the system by directing the local communications through the low-latency, high-bandwidth, and low-power local buses and leaving the global communications to the standard NoC topology.

In practice, however, achieving this locality may be challenging. Even when the algorithm exhibits localized communication; the system needs to be able to map it such that the neighboring threads (from a communication point of view) are mapped onto neighboring network cores. Moreover, even though some applications communicate in a localized fashion at each point in time, sometimes a thread's neighbors can change over time, which may require runtime re-mapping and significant data movement [9]. Therefore, in such a short period of time, when there is only a large burst of long-distance communications, the local buses in HNoC may not be able to provide significant support. However, on average and over time, HNoC indirectly support long-distance communication by removing the local communication traffic from the mesh NoC, leaving the mesh NoC fully dedicated for long-distance traffic only.

7. CONCLUSIONS

A hybrid network on chip (HNoC) fabric that uses local buses for nearest-neighbor communication and the standard NoC topology for global interconnection is presented. It is shown that the local buses can carry all the nearest-neighbor traffic, reducing traffic on the global network, which results in increased throughput and reduced energy consumption.

Based on a communication probability density (CPD) function derived from Rent's rule, it is shown that HNoC can significantly improve the throughput and reduce the NoC energy consumption. To get the benefit of HNoC, compilers must take locality as the primary objective, similar to EDA tools used in VLSI designs.

8. ACKNOWLEDGMENTS

P. Zarkesh-Ha acknowledges the support of the US Department of Energy, Office of Science, under Grant DE-SC0002113. S. Forrest and G. Bezerra acknowledge National Science Foundation (NSF) under Grant CCF-0621900. The authors would like to thank fruitful discussions with Bill Loh at Verdura Systems, and Jim Koford, Venkatesh Akella, Mathew Wojko, and Douglas Boyle at Novarus Logic. Finally, the in depth review of the SLIP technical committee members and their constructive comments are greatly appreciated.

9. REFERENCES

