Design and Evaluation of Technology-Agnostic Heterogeneous Networks-on-Chip

Haera Chung, Portland State University
Christof Teuscher, Portland State University
Partha Pande, Washington State University

Traditional metal-wire-based networks-on-chip (NoC) suffer from high latency and power dissipation as the system size scales up in the number of cores. This limitation stems from the inherent multi-hop communication nature of larger NoCs. It has previously been shown that the performance of NoCs can be significantly improved by introducing long-range, low power, and high-bandwidth single-hop links between distant cores. While previous work has focused on specific NoC architectures and configurations, it remains an open question whether heterogeneous link types are beneficial in a broad range of NoC architectures. In this paper we show that a generic NoC architecture with heterogeneous link types allows for NoCs with a higher bandwidth at a lower cost compared to homogeneous networks. We further show that such NoCs scale up significantly better in terms of performance and cost. We demonstrate these broadly-applicable results by using a technology-agnostic complex network approach that targets NoC architectures with various emerging link types.

Categories and Subject Descriptors: C.2.2 [Computer-Communication Networks]: Network Protocols
General Terms: Design, Algorithms, Performance
Additional Key Words and Phrases: networks-on-chip, heterogeneous, wireless, optical

ACM Reference Format:

1. INTRODUCTION
As the number of cores (or IP blocks) on a single chip increases, the communication between them becomes increasingly important. Traditional Systems-on-Chips (SoCs) interconnect architectures are based on a shared bus structure, which can carry only one communication transaction at a time. This limits the communication bandwidth and scalability [Lu et al. 2007]. Such an architecture is therefore not suitable for future large-scale SoCs. In recent years, Network-on-Chip (NoC) were proposed as a promising solution for designing large and complex SoCs [Benini and De Micheli 2002; Kumar et al. 2002]. The NoC paradigm provides better scalability and reusability for future SoCs. Despite these benefits, conventional metal wire interconnects limit the communication performance of NoCs because of their multi-hop nature for long-range on-chip communication. Multi-hop communication causes high latency and power dissipation [Ganguly et al. 2011], which can lead to the interconnect consuming 80% of the total chip power [ITRS 2007]. The ITRS roadmap also states that “it is now widely con-
H. Chung et al.

eced that technology alone cannot solve the on-chip global interconnect problem with current design methodologies."

To solve this problem, we need new interconnect fabrics that can support single-hop communication across an entire chip. Several new interconnect technologies, such as photonic interconnects [Shacham et al. 2008], multi-band RF NoC [Chang et al. 2008a], Carbon NanoTubes (CNTs) [Dai 2002; Kempa et al. 2007], and millimeter wave wireless (mmWave) [Deb et al. 2010b] were proposed and evaluated in the past. However, these evaluations used in all cases a specific performance metric, such as latency, throughput, or power. In reality, however, trade-offs are key and what may be optimal for one metric may not be optimal for all the others. For example, we can gain high performance by adding more wires to a NoC, but that will increase the wiring cost and the power consumption. It is an open problem how to use these new NoC interconnect technologies in an optimal way [Ogras et al. 2005].

Our research is driven by the following questions:

— Are heterogeneous link type networks beneficial compared to homogeneous networks?
— What are optimal heterogeneous link type distributions for a given traffic scenario?
— What is an optimal placement of the different link types for a given traffic scenario?
— Do heterogeneous link type networks scale better than mesh networks?
— Do sub-networks and small-world networks evolve with heterogeneous link type?

In this paper, we propose optimal NoC architectures with heterogeneous link types. Each link type is characterized by a set of specifications, which include performance and cost measures. We compare the performance of heterogeneous to homogeneous networks under different realistic traffic models and show that our proposed heterogeneous architectures outperform homogeneous architectures in performance, energy, and throughput. In addition, they scale better than regular 2D mesh networks. Our framework is based on technology-agnostic and abstract link types, which can however directly be mapped on actually technology, as we will show. The benefits of a technology-agnostic approach is the broad applicability of our fundamental results.

The main contributions of this work are as following:

(1) We introduce heterogeneous link types in NoCs to solve the current NoC multi-hop wired communication problems and significantly improve the network performance at a low cost. To the best of our knowledge, no one has thoroughly evaluated hybrid networks with three or more different kinds of interconnect technologies in a comprehensive framework that can deal with several design constraints.
(2) Our technology-agnostic approach allows us to obtain optimal networks for a broad range of current and future NoC interconnect technologies.
(3) We present evolutionary optimization techniques to obtain optimal NoC topologies. Our networks significantly outperform homogeneous and regular networks, and we can therefore say that networks with heterogeneous link types built from current technology are very promising because they allow to cover points in the design space that homogeneous networks cannot reach.
(4) We analyze the network topologies from a complex network perspective to determine the small-worldness and the community structure. Our results show that a hierarchical community structure evolves and that some networks have the small-world property. Both the community structure and the small-world property mean that such networks are more scalable than traditional topologies, such as a 2D mesh.

We developed a comprehensive software framework for the design, the optimization, and the evaluation of complex heterogeneous NoC networks. The framework can opti-
mize networks according to any number and combination of the common network performance metrics, such as the wiring cost, the average shortest path length (latency), the throughput, and the energy. Once an optimal heterogeneous NoC architecture has been obtained, we evaluate the network with commonly used uniform and non-uniform workloads to analyze the overall network performance and the cost.

The results presented in this paper are relevant for a broad set of NoC and SoC applications, which rely increasingly on different communication channels.

2. RELATED WORK

Traditional NoC architectures are based on packet-switching networks. In the last few years, several solutions were proposed to improve the network performance. Ogras and Marculescu [Ogras and Marculescu 2006] have proposed inserting a few long-range links to standard mesh NoC topologies to improve the performance of NoCs. The results show that adding long-range links reduces the average distance between source and destination nodes, which increases the network throughput and reduces the average packet latency. The authors did not consider cost and scalability in their paper.

Conventional NoCs use multi-hop packet switched communication. At each hop the data packet goes through a complex router/switch, which contributes considerable power, throughput and latency overhead. By using virtual express lanes to connect distant cores in the network, it is possible to avoid router overhead at intermediate nodes, thereby improving NoC performance [Kumar et al. 2008]. Performance is further improved by incorporating ultra low-latency and multi-drop on-chip global lines (G-lines) for flow control signals [Krishna et al. 2008; Mensink et al. 2007]. Despite the performance gains, schemes in [Kumar et al. 2008] and [Krishna et al. 2008] still require laying out long wires across the chip and avoid only the intermediate router overhead; hence, performance improvements beyond a certain limit will not be achievable.

Design principles of photonic NoCs are elaborated in various recent publications [Shacham et al. 2008; O’Connor et al. 2007; Brière et al. 2007; Joshi et al. 2009; Li et al. 2009]. It is estimated that a photonic NoC will dissipate an order of magnitude less power than an electronic planar NoC. However, some aspects of this new paradigm need more investigation. Photonic NoCs require multi-band laser sources, which may incur excessive power and floor planning overheads [Brière et al. 2007]. Though there has been tremendous recent progress in silicon photonics, the technology required to integrate photonic switches into a complete NoC is still far from mature [Shacham et al. 2008].

Another alternative is NoCs with multi-band RF interconnects (RF-Is) [Chang et al. 2008b]. Various implementation issues of this approach are discussed in [Chang et al. 2008a]. In these NoCs, electromagnetic (EM) waves are guided along on-chip transmission lines created by multiple layers of metal and dielectric stack. As the EM waves travel at the effective speed of light, low latency and high bandwidth communication can be achieved. RF-I NoCs are predicted to dissipate an order of magnitude less power than traditional planar NoCs with significantly reduced latency.

On-chip wireless interconnects were first demonstrated in [Floyd et al. 2002] for distributing clock signals. Recently, the design of a wireless NoC based on CMOS Ultra Wideband (UWB) technology was proposed [Zhao and Wang 2008]. The particular antennas used in [Zhao and Wang 2008] achieve a transmission range of 1 mm with a length of 2.98 mm. The overheads of a wireless link are not justifiable for 1 mm range of on-chip communication compared to a wired channel [Pande et al. 2005a]. A relatively long intra-chip communication range facilitates single-hop communication between widely separated blocks. This is essential to achieve the full benefit of on-chip
wireless networks for multi-core systems by reducing long distance multi-hop wireline communication. In [Deb et al. 2010b], the design of a wireless NoC with long-range millimeter-wave wireless links was proposed. Another wireless NoC architecture with CNT-antenna based THz wireless links was proposed in [Ganguly et al. 2011]. It is demonstrated that by incorporating the characteristics of a small-world network, the wireless NoC outperforms its traditional wired counterparts by orders of magnitude [Deb et al. 2010b; Ganguly et al. 2011].

While many authors have introduced new solutions to improve the network performance, they did not consider balancing all of the important performance metrics of the network, such as the wiring cost, the average shortest path length, the latency, the throughput, and the power consumption. Also, they compared the results with a regular 2D mesh network only and did not compare the networks with each other. Moreover, no one has tried to implement a NoC architecture with more than two heterogeneous link types, such as photonic interconnect, CNT wireless communication, and mm-wave wireless NoC on top of a regular mesh network.

3. ARCHITECTURAL OVERVIEW

In this section, we will present the basic architecture of our framework, define measures, and introduce the methodology.

3.1. The Network Model

We use a graph-theoretic approach to represent our networks [Steen 2010]. A graph $G$ is denoted by $G = (N, E)$, where $N$ is a set of nodes and $E$ is set of edges. Here an edge represents a bidirectional communication link between two nodes. On such a link, a packet can be sent from a source to a destination node. For example, a $4 \times 4$ node 2D mesh network is shown in Figure 1(c). In this example, node 5 is connected to four different adjacent nodes, $E_5 = \{\{5, 1\}, \{5, 4\}, \{5, 6\}, \{5, 9\}\}$, i.e., a packet can be transmitted between node 5 and the four adjacent nodes, and vice versa. The graph-theoretic approach can easily be extended and it allows us to directly employ all tools and methodologies from the graph and complex network community.

For our purpose, all network nodes are arranged in a 2D grid with a one-unit grid spacing (see Figure 1). As opposed to previous work, e.g., [Ogras and Marculescu 2006], we do not start with a mesh interconnect, instead, our optimization algorithm is allowed to place links in an unrestricted manner. That allows us to explore the entire search space of network topologies.

3.2. Link Type Definitions

To explore heterogeneous complex networks with different types of abstract links on the same network, we defined three types of links as shown in Table I. As one can see, each link is defined by a different value for the maximum wire length, the wiring cost, the energy consumption, and the throughput. We created these link characteristics by extrapolating the performance of each link type with respect to metal wires. The values also follow current and predicted data regarding the various interconnect technologies. As shown in [Deb et al. 2010a], the area overhead introduced by the photonic links in a NoC is higher than for wireless links. Thus, we have chosen a higher cost for link type 3 compared to link type 2. Similarly, the achievable bandwidth using photonic links is higher than wireless links. Hence, the throughput of link type 3 was chosen to be higher. Further, the energy dissipation of the photonic links is the lowest of all link types. Lastly, the maximum wire length is motivated by the maximum distance a packet can travel in one clock cycle.

While these links are abstract on purpose, they can nevertheless rather straightforwardly be mapped to actual physical interconnects realized with current technology.
Table I. Definition of the three different types of abstract links

<table>
<thead>
<tr>
<th>Link type</th>
<th>Link type 1</th>
<th>Link type 2</th>
<th>Link type 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Maximum length</td>
<td>1 unit</td>
<td>3 units</td>
<td>20 units</td>
</tr>
<tr>
<td>Variable cost</td>
<td>1</td>
<td>2 × actual length</td>
<td>4 × actual length</td>
</tr>
<tr>
<td>Energy</td>
<td>1 unit/packet</td>
<td>0.1 unit/packet</td>
<td>0.05 unit/packet</td>
</tr>
<tr>
<td>Throughput</td>
<td>1 packet/clock</td>
<td>2 packet/clock</td>
<td>15 packet/clock</td>
</tr>
<tr>
<td>Candidate technology</td>
<td>metal wire</td>
<td>wireless</td>
<td>photonic</td>
</tr>
</tbody>
</table>

Link type 1 corresponds to traditional metal wires, link type 2 to THz wireless links realized by means of carbon nano tubes (CNT), and link type 3 to photonic links. The energy and throughput values correspond to realistic estimates.

By using the abstract links as defined in Table I, the goal then becomes to find an optimal heterogeneous network that has a lower cost, a higher throughput, and a lower energy consumption compared to a homogeneous link type network.

For most of our experiments, we limit the number of each link type. The reason for that is twofold: (1) The network cost would explode if the optimization algorithm is able to place unlimited numbers of links when cost is not or only weakly considered; (2) Technological limitations. Link type limitations are specified in each experiment.

3.3. Network Performance Metrics

For our purpose, we considered network performance metrics such as the wiring cost, the average shortest path length, the throughput, and the energy dissipation to evaluate the network performance. In this section, we will provide brief definitions of these metrics, which are inspired by [Pande et al. 2005b]. For our purpose, we do not consider packet lengths or flits. We simply consider a packet an abstract unit that is communicated on the network. However, to validate our results, we also ran additional performance simulations in Section 7 by using the GEM5 framework, which uses realistic traffic scenarios and flits.

**Network Wiring Cost.** We define the network wiring cost as the sum of the cost of all wires in the network between a source node $i$ and a destination node $j$ that are directly connected.

$$WiringCost_{network} = \sum_{i,j \in G} Cost_{wire(i,j)}$$

The cost of each wire is defined as a function of the pre-defined cost of a link type multiplied by the actual length plus a constant,

$$Cost_{wire} = (a \times WireLength^b) + c,$$

where $a$, $b$, and $c$ are user defined parameters. For example, the network wiring cost for the $4 \times 4$-node mesh network shown in Figure 1(c) is 15. Increasing the network size or adding additional links will increase the total number of links in the network, and hence the wiring cost. Note that we have only used an additional fixed cost factor in Section 6.3.

**Network Average Shortest Path.** The Average Shortest Path (ASP) [Newman 2003] is defined as the average number of hops in the shortest path for all possible $i, j$ pairs of the network nodes with $i \neq j$. 

ACM Journal of Emerging Technologies, Vol. 0, No. 0, Article 0, Publication date: February 2012.
\[
ASP_{\text{network}} = \sum_{i,j \in G} \frac{\text{distance}(i, j)}{N(N - 1)}
\]  

(3)

where \(N\) is the number of nodes. The \(ASP\) measures how efficiently packets can be transported on the network. In an uncongested network with shortest path routing, the \(ASP\) is proportional to the packet latency. The \(ASP\) for the \(4 \times 4\)-node mesh network shown in Figure 1(c) is 3.87. When we add additional long-range links on the mesh network, the \(ASP\) will drop quickly while the clustering coefficient stays high. This is commonly called the \textit{small-world} property of a network [Watts and Strogatz 1998]. We will measure that property in Section 5.

\textit{Network Throughput.} When simulating traffic on the networks, packets will be injected into specific nodes with a given injection rate. The injection rate \(iR\) is defined as the average number of packets injected into a node per clock cycle. The network throughput is defined as the total number of packets arrived at their destination per node per cycle \(T\):

\[
TP_{\text{network}} = \frac{\text{Total packets arrived}}{(N \times \text{Total time})}
\]  

(4)

\(N\) is the number of nodes, and \(\text{Total time}\) is the time (in clock cycles) that elapses between the occurrence of the first message generation and the last message reception.

\textit{Network Energy Dissipation.} The network energy is defined as the sum of the energy required to move packets on the network across links divided by the number of cycles a simulation is run. On each link from source \(i\) to destination \(j\), the number of packets sent will require an energy of \(E_{\text{packet}}(i, j)\). We use the realistic energy estimates per packet as shown in Table I.

\[
E_{\text{network}} = \frac{\sum_{i,j \in G} E_{\text{packet}}(i, j)}{\text{duration}}
\]  

(5)

3.4. Network Traffic

In our framework, packet traffic is simulated in a cycle-accurate manner. However, we do not simulate flits and do not use sophisticated flow control because that is not essential to obtain our results. As a matter of fact, our findings are independent of these details. Traffic is routed by means of a shortest path algorithm. We have chosen to not use flits in order to keep the packet routing deadlock free. There are of course more optimal routing and flow control algorithms than simple shortest path routing, but our approach allows to keep things simple without losing generality. Each node contains FIFO buffers of size \(N\), where \(N\) is the number of network nodes, that store packets upon their arrival.

We use three types of synthetic traffic patterns as described below. For each packet, a source/destination pair \((i, j)\) is generated that depends on the traffic pattern used. In addition, we use the SPLASH-2 [Woo et al. 1995] FFT and RADIX benchmarks for realistic application-based traffic patterns.

\textit{Uniform Random.} In uniform random traffic, the source and destination nodes are randomly chosen among all nodes with equal probabilities.

\textit{Hot-Spot.} In hot-spot traffic, selected hot-spot nodes receive packets with a greater probability \((p)\) than non-hot-spot nodes \((1 - p)\). In our experiments, unless otherwise
stated, we use two hot-spot nodes, namely node 9 and 54 in 8 × 8-node networks (e.g., Figure 3). For networks of size 10 × 10-node and 12 × 12-node, node 11 and 88 and node 13 and 130 are used as hot-spot nodes respectively. The hot-spot probability is \( p = 0.25 \).

**Transpose.** In transpose traffic, the source \((i, j)\) and destination \((j, i)\) node pairs are located symmetrically to the diagonal in a matrix. A roulette wheel is used to select the source and destination pair. For our 8 × 8-node networks (e.g., Figure 3), we used the following node pairs: \((19, 26)\), \((13, 41)\), \((57, 15)\), and \((52, 38)\).

### 3.5. Finding Optimal Networks

Finding optimal NoCs is all about trade-offs. It is rather straightforward to design a high-performance network (e.g., a fully connected), but it will also have a significant cost. On the other hand, a low-cost network may not offer the best performance. The trade-off between performance and cost in general is a design decision that depends on the application. However, most NoC design problems involve several additional factors besides performance and cost. The problem can therefore straightforwardly be formulated as a multi-objective optimization problem.

**Evolutionary Algorithms (EAs)** [Elbeltagi et al. 2005; DeJong 2002] are a well-known metaheuristic technique to solve multi-objective optimization problems. EAs are stochastic search methods based on the principles of natural biological evolution. The basic operation is based on a population of candidate solutions. First, one randomly generates an initial population of individuals, which will then be evaluated by means of an objective or fitness function. If the termination criteria are not met, one creates a new generation of individuals by applying mutation and recombination operators to the parent individuals. This process is repeated until the best solution is found. A pseudo algorithm of the basic parallel evolutionary optimization procedure is described in Algorithm 1, which is inspired by [Fulgham and Snyder 1993].

The strength of EAs is that they perform well with problems that have multiple local optima [Zhou et al. 2011]. EAs are typically used to find best solutions for given problems that cannot easily be solved by using other optimization techniques. They were successfully used in a variety of optimization problems, such as scheduling, routing, transportation problems, and engineering design [Michalewicz 1996; Lienig and Thulasiraman 1993]. Our evolutionary algorithm platform is based on the ParadisEO framework [Cahon et al. 2004], which is a C++ white-box object-oriented framework dedicated to the reusable design of metaheuristics.

![Fig. 1. 4 × 4-node evolved networks. (a) \( w = 0 \): performance is favored. The result is an almost fully connected network. (b) \( w = 0.5 \): wiring cost and performance are equally favored. A hub in node 9 evolves. (c) \( w = 1 \): wiring cost is favored. The resulting network is a sparsely connected tree. Only one link type was used for this example.](image-url)
ALGORITHM 1: Pseudo-code for a parallel evolutionary algorithm.

\begin{algorithm}
\begin{algorithmic}
\State $G \leftarrow$ number of generation, $P \leftarrow$ population size, $O \leftarrow$ offspring, $g \leftarrow 0$
\State Randomly generate initial population $P$;
\While{$g < \text{total generation number } G$}
\State $g \leftarrow g + 1$
\State Evaluate initial population $P(t)$;
\State Crossover:
\If{$x < P_c$}
\State Randomly select two parents;
\State Recombine to create a new offspring $O$;
\State $P'(t) \leftarrow P'(t) \cup O$;
\EndIf
\State Mutation:
\If{$x < P_m$}
\State Mutate to create a new offspring $O$;
\State $P'(t) \leftarrow P'(t) \cup O$;
\EndIf
\State Evaluate $P'(t)$
\State Selection:
\State $P(t + 1) \leftarrow \text{select}(P(t) \cup P'(t))$;
\EndWhile
\end{algorithmic}
\end{algorithm}

Network representation. In our C++ complex network framework, the individual components that make up a network are nodes, links, and link types. The nodes are implemented as an array of pointers to node objects consisted of the physical $(x, y)$ coordinates of the nodes on the 2D grid. The link object stores the link information as a set of node pairs, which includes all relevant link properties, such as cost, maximum length, throughput, and energy consumption. We do not use a genotypical representation for our evolutionary algorithm, instead, all genetic operators work directly on the network level.

Crossover. We perform crossover in the following way: each individual $p_1$ and $p_2$ has a set of link data (i.e., node numbers) stored in an array for each link type. For each link type, we randomly pick two crossover points ($pt_1$, $pt_2$) and then perform standard t2o-point crossover. We do this for each link type.

Mutation. We use two mutation operators in our framework. The first mutation operator randomly selects a link type and changes the number of links by adding and removing a link from the link vector. The second mutation moves a current link to different location in the network. Unless otherwise specified, the mutation rate is 0.4.

Selection. We use deterministic tournament selection to select new individuals for the next generation.

Fitness function. For most of our problems, we consider multiple network performance factors. To consider two different factors only, we introduce an objective aggregate fitness function. For example, to optimize networks for both cost ($WiringCost$) and performance ($ASP$), we defined an aggregate objective fitness function as a following:

$$f(w) = w \cdot WiringCost_{\text{network}} + (1 - w) \cdot ASP_{\text{network}}$$

(6)
Here, $w$ is the weight factor that allows us to determine the importance of either of the two factors, and $\text{WiringCost}$ and the $\text{ASP}$ are normalized. For example, with $w = 0$, only the network performance is considered, and for $w = 1$, only the wiring cost is considered. For $w = 0.5$, both $\text{WiringCost}$ and $\text{ASP}$ are equally favored. The aggregate objective function can readily be extended to include additional factors a designer may want to consider, such as energy or area overhead.

**Population size and evolutionary runs.** Unless otherwise stated, we use a population of 60 individuals evolved over 10,000 generations. For all experiments, we did 10 evolutionary runs with 10 different initial populations and averaged the results. These parameters were determined experimentally and produced very robust results.

**Example.** As an illustrative example, we optimized networks by changing the weight $w$ in Equation 6 to see what kind of NoC topologies we would obtain. We used the EA as described above with only one link type. Figure 1 shows the results for a $4 \times 4$ node evolved network with different weights $w = 0$, $w = 0.5$, and $w = 1$.

We observe that when performance is favored ($w = 0$), we have more long-range links in the almost fully connected network to reach the destination with less number of hops. However, when we consider wiring cost only ($w = 1$), the network becomes a sparsely connected tree and uses local connections only to keep the network cost as low as possible. For $w = 0.5$, a hub in node 9 evolves. The hub allows to keep the average path length low while minimizing the network cost.

## 4. PERFORMANCE EVALUATION

In this section, we will present the performance evaluation experiments in detail. The goal is to show that networks with heterogeneous link types can be beneficial in terms of cost, performance, and energy. A series of experiments with increasing complexity will illustrate that.

### 4.1. Optimal Number of Links

The goal of the first experiment was to determine the optimal number of each of the different link types for three types of traffic scenarios, namely uniform random, hot-spot, and transpose traffic. For this experiment, we evolved optimal $8 \times 8$-node networks with an injection rate of $iR = 0.6$. We also limited the number of each link type to 112, the number of links that would be required for a complete local 2D mesh of size $8 \times 8$ nodes. As a baseline for comparison, we also used an optimal network that was evolved without traffic.

We will first present the results individually before comparing them in Section 4.1.5.

#### 4.1.1. Networks Without Traffic.

First, we evolve networks without inserting any traffic to establish a baseline. For this experiment, we only considered $\text{WiringCost}$ and $\text{ASP}$ to see the distribution of the three different link types in the network by changing the importance of $\text{WiringCost}$ versus $\text{ASP}$. The aggregate objective function $f(w) = w \times \text{WiringCost} + (1 - w) \times \text{ASP}$ is used in this experiment.

The resulting optimal networks are shown in Figure 3 while Figure 2 shows the distribution of the number of links as a function of the weight $w$. As one can see, when performance is favored ($w = 0$), all three link types are used in the network. In that case, a highly connected network with a hub (node 37) evolves. The hub serves to lower the average shortest path ($\text{ASP}$), so that between almost any pair of nodes, there is only one hop at most. The hub is only beneficial in a network without traffic because it would lead to significant congestion otherwise.
Heterogeneous link type distribution as a function of the weight $w$. Depending on the importance of performance versus cost, a different number of each link type is used. A tree evolves when cost is favored ($w = 1$), while a highly connected network with a significant hub (node 37) evolves if performance is favored ($w = 0$).

When we give more weight to $\text{WiringCost}$ (i.e., higher $w$), the proportion of link type 1 increases while the proportion of link type 2 and link type 3 decreases. When wiring cost is the only concern, we can see that only short-range links with lower cost links are used (see Figure 3 (c)). The network then basically becomes a sparse tree. For an equal importance of the two factors, one can see that two big hubs evolve in node 18 and 53. A smaller hub is located at node 21. The hubs help to lower the average shortest path while keeping the wiring cost low. The network in Figure 3 (a) is about 16 times more expensive than in (c), but the $\text{ASP}$ is only less than twice as low. Note that compared to Figure 1 (b), two hubs evolved in 3 (b) due to the bigger network size.

**4.1.2. Networks with Uniform Random Traffic.** Next, we added random uniform traffic to study how the heterogeneous link type distribution changes compared to the no-traffic scenario. For that purpose, we randomly generated packets with source and destination nodes selected with a uniform probability. The injection rate for this experiment was $iR = 0.6$.

Figure 4 shows the resulting link type distribution for networks optimized for $\text{WiringCost}$ and throughput $\text{TP}$ with the aggregate function $f(w) = w \times \text{wireCost} + (1 - w) \times \text{TP}$. Figure 5 shows the corresponding networks. Compared to Figure 2, link type
1 is not used in most of the networks, except for the two networks where cost matters the most. This can be explained by link type 2 and 3 having a much higher throughput than link type 1. Note the absence of hubs, which would lead to congested nodes. As opposed to a network with hubs, a tree-like network based solely on link type 1 can be seen in Figure 5 (c) because that is the cheapest possible way to build a network.

Fig. 4. Heterogeneous link type distribution as a function of the weight $w$. The networks are optimized for $WiringCost$ and $TP$ with uniform random traffic. Injection rate $iR = 0.6$.

Fig. 5. $8 \times 8$-node evolved networks with uniform random traffic. (a) $w = 0.05$: $TP$ only is important. $WiringCost = 342.8$, $TP = 0.59$. (b) $w = 0.5$: wiring cost and $TP$ are equally important. $WiringCost = 249.2$, $TP = 0.58$. (c) $w = 1.0$: wiring cost only is important. $WiringCost = 63$, $TP = 0.02$. Black links: type 1; green links: type 2; red links: type 3.

We also evaluated the energy $E$ combined with $WiringCost$. As one can see from Figure 6 and Figure 7, the results are similar to the throughput experiments. The network is constructed with cheaper links to lower the network cost and expensive links are used to reduce the network energy.

4.1.3. Networks with Hot-spot Traffic. In this experiment, we used hot-spot traffic [Ogras and Marculescu 2006] as a more realistic traffic pattern. The two hot spots are node 9 and 54. The hot spot probability is $p = 0.25$, i.e., 25% or the packets will be sent to the hot-spots.

The results of optimizing the networks for $WiringCost$ and $TP$ are shown in Figures 8 and 9. The distribution plot shows that almost all optimal networks use three types of links with that traffic scenario. As we will see later in Section 5, local links of type 1
Fig. 6. Heterogeneous link type distribution as a function of the weight $w$. The networks are optimized for $WiringCost$ and $E$ with uniform random traffic. Injection rate $iR = 0.6$.

Fig. 7. $8 \times 8$-node evolved networks with uniform random traffic. (a) $w = 0.05$: $E$ only is important. $WiringCost = 385.3$, $E = 0.4$. (b) $w = 0.5$: wiring cost and $E$ are equally important. $WiringCost = 154.2$, $E = 0.9$. (c) $w = 1.0$: wiring cost only is important. $WiringCost = 63$, $E = 18.1$. Black links: type 1; green links: type 2; red links: type 3.

are used around the hot-spots to absorb the traffic and long-range link of type 2 and 3 are used to connect the clustered subnets.

Fig. 8. Heterogeneous link type distribution as a function of the weight $w$. The networks are optimized for $WiringCost$ and $TP$ with hot-spot traffic. Injection rate $iR = 0.6$. 
Fig. 9. 8 × 8-node evolved networks with hot-spot traffic. (a) \( w = 0.05 \): TP only is important. \( \text{WiringCost} = 863.1, \text{TP} = 0.59 \). (b) \( w = 0.5 \): wiring cost and TP are equally important. \( \text{WiringCost} = 598.3, \text{TP} = 0.58 \). (c) \( w = 1.0 \): wiring cost only is important. \( \text{WiringCost} = 63, \text{TP} = 0.02 \). Black links: type 1; green links: type 2; red links: type 3. Hot spot nodes are marked with a ×.

Fig. 10. Heterogeneous link type distribution as a function of the weight \( w \). The networks are optimized for \( \text{WiringCost} \) and \( E \) with hot-spot traffic. Injection rate \( iR = 0.6 \).

Fig. 11. 8 × 8-node evolved networks with hot-spot traffic. (a) \( w = 0.05 \): \( E \) only is important. \( \text{WiringCost} = 902.7, E = 0.2 \). (b) \( w = 0.5 \): wiring cost and \( E \) are equally important. \( \text{WiringCost} = 189.3, E = 0.4 \). (c) \( w = 1.0 \): wiring cost only is important. \( \text{WiringCost} = 63, E = 3.7 \). Black links: type 1; green links: type 2; red links: type 3. Hot spot nodes are marked with a ×.
The results for optimizing $WiringCost$ and $E$ with hot-spot traffic are shown in Figures 10 and 11. As one can see, long-range links of type 2 and 3 are used more frequently for the majority of the weight values $w$ compared to the optimal network with uniform random traffic. The long-range links naturally help to reduce the average distance a packet travels on the network.

4.1.4. Networks with Transpose Traffic. In transpose traffic [Ogras and Marculescu 2006], the source $(i, j)$ and destination $(j, i)$ nodes pairs are located symmetrically to the diagonal in a matrix. A roulette wheel is used to select the source and destination pair. Here, we use the following node pairs: $(19, 26)$, $(13, 41)$, $(57, 15)$, and $(52, 38)$ for our $8 \times 8$-node networks.

Figures 12 and 13 show the results for networks optimized for $WiringCost$ and $TP$, i.e., the aggregate objective function is $f(w) = w \times WiringCost + (1 - w) \times TP$. As one can see, long-range link of link type 2 and 3 are used more frequently to absorb the traffic for the majority of the weight values. Especially, when we consider $TP$ to be more important, the optimal network uses more link type 3 compared to the optimal network with hot-spot traffic.

**Fig. 12.** Heterogeneous link type distribution as a function of the weight $w$. The networks are optimized for $WiringCost$ and $TP$ with transpose traffic. Injection rate $iR = 0.6$.

**Fig. 13.** $8 \times 8$-node evolved networks with transpose traffic. (a) $w = 0.05$: $TP$ only is important. $WiringCost = 955.4$, $TP = 0.59$. (b) $w = 0.5$: wiring cost and throughput are equally important. $WiringCost = 613.4$, $TP = 0.58$. (c) $w = 1.0$: wiring cost only is important. $WiringCost = 63$, $TP = 0.01$. Black links: type 1; green links: type 2; red links: type 3.
Figures 14 and 15 show the results for networks optimized for \textit{WiringCost} and \( E \). When energy is considered to be more important, the optimal networks use more long-range links because these links are more energy-efficient.

![Fig. 14](image)

**Fig. 14.** Heterogeneous link type distribution as a function of the weight \( w \). The networks are optimized for \textit{WiringCost} and \( E \) with transpose traffic. Injection rate \( iR = 0.6 \).

![Fig. 15](image)

**Fig. 15.** 8 × 8-node evolved networks with transpose traffic. (a) \( w = 0.05 \): \( E \) only is important. \( \text{WiringCost} = 576.3, E = 0.3 \). (b) \( w = 0.5 \): wiring cost and \( E \) are equally important. \( \text{WiringCost} = 185, E = 0.4 \). (c) \( w = 1.0 \): wiring cost only is important. \( \text{WiringCost} = 63, E = 3.8 \). Black links: type 1; green links: type 2; red links: type 3. Hot spot nodes are marked with a \( \times \).

**4.1.5. Comparison and Discussion.** We will briefly discuss and compare the results from Sections 4.1.1 to 4.1.4.

Figure 16 shows the total number of links (i.e., the sum of link type 1, 2, and 3) used in the networks from the previous experiments. Note that we had limited the number of each link type to 112, the number of links that would be required for a complete local 2D mesh of size 8 × 8 nodes. As one can see in Figure 16 (a), the evolved networks with hot-spot and transpose traffic use more than twice as many links compared to uniform random traffic to absorb the traffic. Hot-spot and transpose traffic show very similar results otherwise. When optimized for cost and energy, the results are somewhat different, as Figure 16 (b) shows. The total number of links converges to 63 links for weights \( w > 0.5 \), i.e., when energy becomes more important. Also, considering energy as an optimization factor results in using about half the number of links compared to networks where \( TP \) is considered besides the cost.
Fig. 16. Total number of links used as a function of the weight $w$ for three different traffic patterns with injection rate $iR = 0.6$. (a) The network is optimized for $WiringCost$ and $TP$; (b) the network is optimized for $WiringCost$ and $E$.

Figure 17 shows the usage of each link type for three different traffic patterns. Figure 17 (a) shows the networks optimized for cost and throughput $TP$ and Figure 17 (b) for cost and energy $E$. We can see that overall, link type 3 is used the least because it is expensive, yet offers great performance. The more $TP$ and $E$ are important, the more links of type 2 and 3 are used. When we only optimize for cost ($w = 1$), only link type 1 is used because it is very cheap.

In summary: with the traffic patterns and trade-offs under consideration so far, it is beneficial in all cases (except when cost only is important) to make use of heterogeneous links to increase the performance or lower the energy at a low cost. Note that we did not optimize networks for all three factors together, i.e., cost, energy, and throughput, because energy and throughput are essentially proportional.
4.2. Optimal Network Throughput

In this section we are investigating the optimal network throughput for different traffic patterns and injection rates. We limited the maximum number of links that can be used in the network for each link type to 112 for link type 1, 24 for link type 2, and 62 for link type 3 respectively. The specific limits for link type 2 and 3 are motivated by technological considerations. For example, it is currently possible to only create 24 non-overlapping wireless links (link type 2) using CNT antennas [Ganguly et al. 2011]. Similarly, using Wavelength Division Multiplexing (WDM), 62 maximum wavelength channels can be created in case of photonic interconnects (link type 3) [Preston et al. 2011].

We consider 8 × 8-node networks with three different traffic patterns (see Section 3.4), namely uniform random, hot-spot, and transpose with a target injection rate of iR = 0.6. First we find an optimal network by optimizing cost and throughput for each traffic pattern by equally weighting their importance (w = 0.5). Figure 19 shows the resulting networks for these three different traffic scenarios. A first observation is that no hubs evolve, not even for the hot-spot traffic scenario.

![Fig. 18](image)

**Fig. 18.** Network throughput of evolved topology and mesh network under different traffic scenarios with a target injection rate of iR = 0.6. All evolved topologies with heterogeneous links perform significantly better than a 2D mesh topology.

![Fig. 19](image)

**Fig. 19.** 8 × 8-node optimal evolved networks under different traffic scenarios. WiringCost and TP is equally weighted (w = 0.5) using an aggregate objective function: f = w × WiringCost + (1 − w) × TP. Black links: type 1; green links: type 2; red links: type 3. Hot spot nodes are marked with a ×.
Next, we took these networks that were optimized for $iR = 0.6$ and subjected them to different injection rates to find out the throughput saturation point. As one can see from Figure 18, the network throughput increases and peaks at the target injection rate of $iR = 0.6$. The throughput peaks at $iR = 0.6$ because the networks were optimized specifically for that target throughput. Because cost is a factor in the optimization, only as many links as needed will be added to the network. Thus, for any traffic that is beyond the target injection rate, the network will not be optimal. Second, the networks seem to lack structure. However, as we will see in Section 5, the underlying structure is not easily visible to the human eye.

In addition, we also compared the network throughput $TP$ of our evolved topologies with a $8 \times 8$-node mesh under the same traffic patterns. Figure 18 shows that all of our evolved topologies provide a higher throughput than the mesh networks. This shows once again that networks with heterogeneous link types can achieve better performance than regular 2D mesh networks. It is not surprising that adding long(er)-range connections to a mesh network helps to improve the performance. That was successfully shown by Ogras and Marculescu [Ogras and Marculescu 2006], but only for homogeneous networks. In Section 4.4 we will show that our heterogeneous networks perform better than any other homogeneous single link type network.

4.3. Optimal Energy Dissipation

As the number of cores on a single chip increases, energy reduction is another challenge in a traditional NoC. Multi-hop communication based on metal wires increases the energy consumption significantly, so any link type that can communicate to further away cores in a single hop with low energy and high bandwidth is interesting to explore.

In this section we use the abstract heterogeneous link types defined in Section 3.2 to find optimal networks under $\text{WiringCost}$ and energy $E$ constrains with an equal importance, i.e., $w = 0.5$, $f(w) = w \times \text{WiringCost} + (1 - w) \times \text{energy}$. We one again limited the maximum number of links that can be used in the network for each link type to 112 for link type 1, 24 for link type 2, and 62 for link type 3 respectively. The same traffic patterns as used before were used and the injection rate was $iR = 0.6$.

The resulting optimal networks with heterogeneous links are shown in Figure 21. We then took the optimal networks for each weight $w$ and subjected them to traffic

![Image of network energy comparison](image-url)
with different injection rates. The results were compared with mesh networks. As one can see in Figure 20, the evolved topologies consume much less energy comparing to the mesh for all three traffic patterns. This is because the evolved topologies are interconnected with long-range links that help to reduce the number of hops.

The results show that with respect to energy consumption, optimal networks with heterogeneous links are more efficient than a regular 2D mesh networks. Thus, as seen here and in Section 4.2, the evolutionary algorithm does a splendid job in obtaining networks that meet the target requirements while minimizing both energy and cost.

4.4. Network Performance Comparison with Mesh and Homogeneous Network

Next, we evaluated the performance metrics, such as throughput $TP$ and energy $E$ of networks with heterogeneous link type and compared them with regular 2D meshes and networks with homogeneous link types. We define a homogeneous link type network as a network that only uses link type 1, but without a length restriction in order to allow long-range links. This network is based on a regular 2D mesh but contains additional long-range connections to improve the performance. The network therefore corresponds to the type of network as proposed by Ogras and Marculescu [Ogras and Marculescu 2006]. The only difference is that we consider cost while they did not.

Fig. 22. Performance comparison of heterogeneous link type network with regular 2D mesh and homogeneous link type network under uniform random traffic. Injection rate $iR = 0.6$. (a) Network throughput $TP$ as a function of the weight $w$. (b) Network energy $E$ as a function of the weight $w$. 

ACM Journal of Emerging Technologies, Vol. 0, No. 0, Article 0, Publication date: February 2012.
Figures 23 and 24 show the same performance comparison for hot-spot and transpose traffic. Again, one can see that our evolved networks with heterogeneous links result in a higher throughput and a lower energy consumption than 2D mesh networks or homogeneous link type networks. The results in this section indicate that having multiple types of links improves the network performance and lowers the energy at a low cost.

Fig. 23. Performance comparison of heterogeneous link type network with regular 2D mesh and homogeneous link type network under hot-spot traffic. Injection rate $iR = 0.6$. (a) Network throughput $TP$ as a function of the weight $w$. (b) Network energy $E$ as a function of the weight $w$.

Fig. 24. Performance comparison of heterogeneous link type network with regular mesh and homogeneous link type network under transpose traffic. Injection rate $iR = 0.6$. (a) Network throughput $TP$ as a function of the weight $w$. (b) Network energy $E$ as a function of the weight $w$.

Figure 22 shows the comparative results of heterogeneous link type networks with regular 2D meshes and homogeneous link type networks under uniform random traffic. As one can see, both the network throughput $TP$ and energy $E$ of our evolved networks with heterogeneous links are significantly better compared to a regular 2D mesh network over the entire weight range, except when the designer gives a higher than 90% importance to cost.
4.5. Scalability

As the number of cores integrated on a single chip increases, scalability yet another challenge that needs to be addressed by the NoC community.

We were interested to find how the performance, cost, and energy of networks with heterogeneous links scale up as a function of the system size $N$. For this experiment, we evolved networks of sizes $8 \times 8$-node, $10 \times 10$-node, and $12 \times 12$-node and subjected them to hot-spot traffic with an injection rate of $iR = 0.6$. We then compared the network throughput $TP$ and the energy $E$ with regular 2D mesh networks. To account for the bigger network sizes, we increased the maximum number of link type 1 to 180 and 264 for a $10 \times 10$-node and $12 \times 12$-node network respectively. However, the maximum number of links of type 2 and 3 remained fixed to 24 and 62 respectively for all the networks.

The results for the network throughput $TP$ and the energy $E$ of different sizes are shown in Figure 25. As one can see, networks with heterogeneous link types provide higher throughput and lower energy consumption compared to regular 2D mesh networks for all system sizes under consideration. We also observe that the network throughput decreases as the system size increases. This is simply due to network congestion.

5. NETWORK ANALYSIS

Most of the networks evolved so far seemingly lacked structure and looked rather chaotic. To get a better insight into why the evolutionary algorithm comes up with specific networks, we analyzed the evolved networks from a complex network perspective. We used the brain connectivity analysis toolbox [Rubinov and Sporns 2009], which is an open source software package. First we were curious to determine if actual small-world networks evolved. We used the formula provided in Rubinov and Sporns’ paper [Rubinov and Sporns 2009]:

$$S = \frac{(C/C_{rand})}{(L/L_{rand})}$$  \hspace{1cm} (7)
Table II. Small-worldness for evolved topologies. If $S$ is greater than 1, the network is said to be a small-world network.

<table>
<thead>
<tr>
<th></th>
<th>Uniform random</th>
<th>Hot-spot</th>
<th>Transpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Optimized for throughput and cost</td>
<td>$S = 0.8894$</td>
<td>$S = 1.3894$</td>
<td>$S = 1.9190$</td>
</tr>
<tr>
<td>Optimized for energy and cost</td>
<td>$S = 0$</td>
<td>$S = 1.0540$</td>
<td>$S = 0$</td>
</tr>
</tbody>
</table>

Measuring the small-worldness ($S$) is based on the local clustering coefficient ($C$) and characteristic short path length ($L$). ($C$) is a measure of the neighbor's node connectivity and ($L$) is the average shortest path length (ASP) between all source and destination pairs in the network. For small-worldness to be high, a network needs to have a high connectivity and low characteristic path length. If $S$ is greater than 1, the network is said to be a small-world network. We use the Erdős-Rényi network for the random network in the formula [Erdos and Rényi 1960].

The results for our network are shown in Table II. As one can see, only the evolved networks optimized for throughput $TP$ and $WiringCost$ with both hot-spot and transpose traffic show small-worldness as defined by the formula above. As shown in Figure 16, the networks optimized for energy $E$ and $WiringCost$ use less wires, and are therefore much sparser, which results in a low clustering coefficient. That is main the reason why these networks not have the small-world property.

Next, we explored the evolved networks to find the network community structure by using the algorithm of Clauset et al. [Clauset et al. 2004]. This optimization algorithm detects communities that are strongly connected within each community and weakly connected between communities. Figure 26 (a) and (b) show the network community of the evolved networks from Figure 9 (b) and Figure 13 (b) respectively. These networks were optimized for throughput $TP$ and $WiringCost$ with equal importance ($w = 0.5$). The community structure is measured by a $Q$ value. If $Q$ gets closer to 1, then a network has a strong community structure.

The network community in Figure 26(a) is maximized at $Q = 0.4793$ with a partition into 4 communities while in Figure 26(b), it is maximized at $Q = 0.4615$ with a partition into 5 communities. As one can see, a short-range link of link type 1 is used to locally connect inside a community and long-range links of link type 2 and 3 are used to...
communicate between communities to distribute the traffic and improve the network performance.

So while the networks presented in the paper may all seem unstructured at first, there is an underlying structure that evolves as a function of the design constraints. Also, most of the networks are indeed small-world. In [Teuscher 2007], it was already shown that unstructured small-world interconnect networks can have major advantages over local 2D or 3D regular topologies, however, these networks were not evolved.

6. MODEL VARIATION

In this section we will change several model assumptions to illustrate that our outcomes are robust against such variations.

6.1. Performance Comparison of Heterogeneous Link Type Networks with Homogeneous Link Types

In order to show that networks with heterogeneous link types provide higher throughput and lower energy consumption compared to networks with homogeneous link types, we have performed additional simulations by using one type of link (only link type 2 or link type 3) for long-range link on top of a regular mesh network. Figure 27 shows that networks with three different link types improve the network performance compared to networks with one link type only.

![Fig. 27. Performance comparison between the networks with three different link types vs. a single link type. (a) The networks are optimized for $WireCost$ and $TP$; (b) the networks are optimized for $WireCost$ and $EG$ under uniform random traffic by equally weighting their cost-performance weight ($w = 0.5$). Injection rate $iR = 0.6$.](image)

While a single technology may indeed be cheaper and easier to manufacture, our results clearly show that performance can be significantly improved with additional technologies. How much one is willing to pay is ultimately a design decision influenced by the application and the market. We believe that there are plenty of applications (e.g., defense applications, aerospace, bioinformatics, datacenters, etc.) where performance is a constant bottleneck, but cost is not a primary issue of concern. The other advantage of our approach is that we can determine the optimal network for essentially any point in the cost-performance space.
6.2. Performance Evaluation with Different Traffic Scenarios

As explained in Section 3.5, our networks were so far evolved for specific traffic patterns. However, it is straightforward to design them for multiple traffic scenarios at the same time. For this experiment, we optimized networks for WireCost and TP under uniform random traffic and evaluated with hotspot and transpose traffic. The results are shown in Figure 28. As one can see, the results show that the optimal network with uniform traffic provides higher throughput compared to the other traffic patterns. This is because the network was optimized specifically for this target traffic pattern. However, one can also notice that although the networks were optimized for uniform random traffic, they still perform better on most other common traffic patterns. This illustrates that our evolved networks contain structure that is universally beneficial for various scenarios.

![Figure 28](image)

Fig. 28. Performance comparison of networks evolved under uniform random traffic and tested on hot-spot and transpose traffic. (a) network throughput; (b) energy; $w = 0.5$. The results show that the networks perform significantly better than a mesh network on the traffic patterns they were not evolved for.

6.3. Performance Comparison with a Different Cost Formula

To account for different cost assumptions, we added a fixed cost component for each new technology that is being used when a new link type is instantiated. The goal is to show that our main outcomes remain unchanged under that new and more realistic model. The wiring cost formula is shown in Equation 2, Section 3.3. The fixed cost values $c$ we used for each link type was set to 1 for link type 1, 2 for link type 2, and 4 for link type 3 respectively. For the sake of this experiment, we assumed that the technology becomes more expensive the more powerful the links are.

Figure 29 shows the results obtained by optimizing WireCost and TP under uniform random traffic. As one can see, the main results of this article remain unchanged. In particular, the throughput and the link type distribution are almost identical.

7. PERFORMANCE EVALUATION BY USING THE GEM5

In order to validate the results of our abstract framework, we used the GEM5 platform [Binkert et al. 2011] to undertake a full system simulation with the Ruby and Garnet fixed-pipeline network model on a 64-core Alpha architecture. For our results, we used the SPLASH-2 [Woo et al. 1995] FFT, RADIX, and LU kernels as real ap-
Fig. 29. Performance comparison of networks with and without technology cost. Networks are optimized for $WireCost$ and $TP$ under uniform random traffic by equally weighting their cost-performance weight ($w = 0.5$). Injection rate $iR = 0.6$. (a) Throughput comparison of the evolved topologies; (b) heterogeneous link type distribution comparison.

Application benchmarks to measure the network performance and to compare it with a regular 2D mesh.

Fig. 30. GEM5 performance comparison between evolved and mesh topologies. Our evolved topologies have a 33%, 41%, and 37% lower latency compared to a regular 2D mesh network under FFT, Radix, and LU traffic respectively. The execution time for FFT, RADIX, and LU are 0.003172, 0.064782, and 0.092254 respectively. The results validate our abstract framework.

To obtain our results, we add our obtained heterogeneous network topologies in GEM5 and run the SPLASH-2 benchmarks on them. Figure 30 shows the network performance comparison between our evolved topology and a regular 2D mesh network. As one can see, the latency with the SPLASH-2 FFT, RADIX, and LU traffic loads are much lower for our evolved topology than for a regular mesh. These results therefore confirm the results obtained so far by our abstract framework by means of realistic traffic loads.
8. CONCLUSION

In this paper, we presented a comprehensive study on the benefits of heterogeneous link types in a generic networks-on-chip architecture to solve the traditional multi-hop communication problem and to improve the overall network performance. We used an evolutionary framework to evolve optimal networks under various design constraints, traffic patterns, and injection rates. Compared to other work, we do not assume an initial network topology. Instead, the evolutionary algorithm can place links without restriction, which allows to explore the entire search space. While the link types are kept abstract on purpose to make our results and the framework applicable to a broad range of technologies, the presented links can rather straightforwardly be mapped to current interconnect technology.

We have shown that our optimal networks based on heterogeneous link types provide a higher throughput and a lower energy consumption compared to both homogeneous link type networks and regular 2D mesh networks under uniform random, hot-spot, and transpose traffic patterns. The obtained optimal network uses long-range links of link type 2 and 3 on the network to distribute the network traffic.

To test our abstract model against variations in the assumptions we made, we performed simulations that considered a fixed cost for each used technology. As the reader has seen, even with these new assumptions, all the outcomes of the article remained valid. We have also shown that, although the networks are optimized for a specific traffic pattern, they still outperform when used on other common traffic patterns. This confirms that our modeling abstraction level is appropriate for the type of study we performed and that our results are fundamental and broadly applicable. Simulations done with the GEM5 framework also support all our results.

When long-range links are added on the network, network performance and cost are both considered and balanced. With an increase in system size, the number of long-range links on the network increases to absorb the additional network traffic. This supports an efficient network scalability without significantly reducing system performance.

We conclude by giving brief answers of the questions posed in the introduction. First, our results clearly confirm that heterogeneous link type networks are beneficial compared to homogeneous networks. We have provided optimal link type distributions and optimal link placements throughout the paper. We can also confirm that the networks scale better. This is partially true to the small-world property. Last but not least, we were able to confirm that a sub-network structure evolves and that the different link types are efficiently used to connect the sub-networks.

We believe that our results are relevant for the design of NoCs with emerging link types, such as wireless and optical networks. While these technologies are in almost all cases considered separately, we have made a strong case that it would be beneficial to consider them in a combined way. Also, both the results and the evaluation methodology are applicable broadly because we kept them technology-agnostic.

Future work will focus on exploring more dynamic workload variations and additional benchmarks. We are also planning to compare our results with additional network topologies, in particular 3D topologies.

ACKNOWLEDGMENT
The work was partially sponsored by DARPA grant # HRL0011-09-C-001.

REFERENCES
Design and Evaluation of Technology-Agnostic Heterogeneous Networks-on-Chip

0:27


H. Chung et al.


