IEICE TRANS. INF. & SYST., VOL.E95–D, NO.12 DECEMBER 2012

PAPER Special Section on Parallel and Distributed Computing and Networking

A Hybrid Photonic Burst-Switched Interconnection Network for Large-Scale Manycore System

Quanyou FENG[†a)], *Nonmember*, **Huanzhong LI**[†], *Student Member*, *and* **Wenhua DOU**[†], *Nonmember*

SUMMARY With the trend towards increasing number of cores, for example, 1000 cores, interconnection network in manycore chips has become the critical bottleneck for providing communication infrastructures among on-chip cores as well as to off-chip memory. However, conventional on-chip mesh topologies do not scale up well because remote cores are generally separated by too many hops due to the small-radix routers within these networks. Moreover, projected scaling of electrical processor-memory network appears unlikely to meet the enormous demand for memory bandwidth while satisfying stringent power budget. Fortunately, recent advances in 3D integration technology and silicon photonics have provided potential solutions to these challenges. In this paper, we propose a hybrid photonic burst-switched interconnection network for large-scale manycore processors. We embed an electric low-diameter flattened butterfly into 3D stacking layers using integer linear programming, which results in a scalable low-latency network for inter-core packets exchange. Furthermore, we use photonic burst switching (PBS) for processor-memory network. PBS is an adaptation of optical burst switching for chip-scale communication, which can significantly improve the power efficiency by leveraging sub-wavelength, bandwidth-efficient optical switching. Using our physically-accurate network-level simulation environment, we examined the system feasibility and performances. Simulation results show that our hybrid network achieves up to 25% of network latency reduction and up to 6 times energy savings, compared to conventional on-chip mesh network and optical circuit-switched memory access scheme.

key words: *manycore system, network on chip, photonic burst switching, flattened butterfly, 3D stacking*

1. Introduction

High performance manycore processors are essential for Exascale computing era [1]. To improve their capacity, we generally integrate more computing cores on a single die. For example, the Tilera TILE-Gx processor [2] provides unprecedented performance by leveraging 100 cores. With the trend of manycore chips towards increasing number of cores, for example, 1000 cores [4], interconnection network has become the central subsystem. How to provide efficient inter-core communications and sustain the enormous demand for off-chip memory access in an energy-efficient manner has become a critical challenge.

Inter-core communications and processor-memory networks have different constraints and challenges when manycore system scales up. It has been discovered that, latency, not bandwidth, dominates in inter-core communication for current manycore chips [3], since conventional on-chip topologies (such as, 2D mesh [9]) generally use low-radix routers and hence result in long network diameter. Moreover, projected scaling of electrical processor-memory network appears unlikely to meet the enormous demand for off-chip bandwidth while satisfying stringent power budget. Though existing wavelength routed [8] or circuit-switched optical interconnects [11], [12] are possible to meet the bandwidth demand, they suffer from poor bandwidth efficiency because a dedicated wavelength channel along the whole route has to be reserved exclusively for packet transmission. These flaws motivate us to perform further investigations.

In this work, by leveraging the recent advent of three dimensional integration (3D-I) [5] and silicon photonics [6], we propose a hybrid photonic burst-switched interconnection network for large-scale manycore system. Our hybrid architecture uses separate networks for inter-core and processor-memory communication. We embed a low-diameter flattened butterfly into 3D stacking layers for inter-core traffic flows. The 3D flattened butterfly outperforms conventional topologies by using high-radix routers and express one-hop vertical interconnects. In addition, we use photonic burst switching (PBS) for processor-memory network. PBS is an adaptation of optical burst switching [14] for chip-scale network using silicon photonic devices. The PBS network meets the enormous bandwidth demand and stringent energy constraints by using high-speed low-power CMOS-compatible photonic devices. Furthermore, it has higher bandwidth utilization than wavelength routing and optical circuit-switching because of sub-wavelength optical switching.

We examined the system feasibility and performances using physically-accurate network-level simulation environment. We evaluated the architecture using synthetic traffic patterns and real workloads traces. Simulation results show that the hybrid network achieves up to 25% of average network latency reduction and up to 6 times energy savings, compared to conventional on-chip mesh and optical circuit-switched memory access network.

The rest of this paper is organized as follows. Section 2 presents related backgrounds on flattened butterfly and summarizes state of the art on photonic chip-scale networks. Section 3 first describes the hybrid architecture and then focuses on the design of our 3D flattened butterfly. Section 4 details our PBS memory access network. Section 5 presents the simulation results and discussions. Conclusions and future work are included in Sect. 6.

Manuscript received December 30, 2011.
Manuscript revised April 26, 2012.

[†]The authors are with the School of Computer Science, National University of Defense Technology, China.
a) E-mail: fengquanyou@nudt.edu.cn

DOI: 10.1587/transinf.E95.D.2908

2. Background and Related Work

2.1 The Flattened Butterfly Topology

The flattened butterfly (FBT) [7] is derived by flattening the routers in each row of a conventional butterfly topology while preserving inter-router connections. The major advantage of FBT over conventional mesh topologies [9] is the small average number of hops for network traversals under minimal routing. However, high-radix routers and long wires in on-chip planar FBT [7] cost too much area and power when manycore chips scale up. Moreover, unlike mesh or torus, adding nodes in the FBT requires the rewiring and layout of the entire chip, since a node has to be connected to each dimension. These drawbacks of on-chip FBT must be solved for system scalability.

2.2 Photonic Chip-Scale Network

CMOS-compatible photonic devices, including modulators, detector, waveguides and photonic switches have all been demonstrated [6] by silicon photonic technology and they have paved the way for optical chip-scale network design. For example, D. Vantrease et al. [8] used a dense wavelength division multiplexed crossbar to connect off-stack memory modules; Shacham et al. [11] proposed a hybrid photonic-electric network for both on-chip and off-chip communications; Hendry et al. [12] proposed an optical circuit-switched memory access scheme.

Generally, photonic links have a significant static energy cost in thermal tuning circuits and optical laser sources, which can be much higher than the static energy cost of electrical links. Hence, networks designed using photonic links need to have high utilization to offset the large static energy overhead. Existing photonic chip-scale network schemes, including wavelength routing [8] and optical circuit-switching [11], [12], suffer from poor bandwidth utilization since a exclusive wavelength channel must be reserved from source to destination for the whole duration of packet transmission. The large portion of static energy overhead in these networks actually degrades their energy efficiency.

3. The Hybrid Interconnection Network

In this section, we first present the rationale behind our hybrid architecture and then focus on the design of an on-chip 3D flattened butterfly. The PBS memory access network is introduced in Sect. 4.

3.1 Rationale Behind the Hybrid Architecture

On-chip network and off-chip memory access network have different constraints. Latency, not bandwidth, dominates on-chip inter-core communication while bandwidth and energy efficiency dominate processor-memory interconnection. These different requirements motivate us to employ

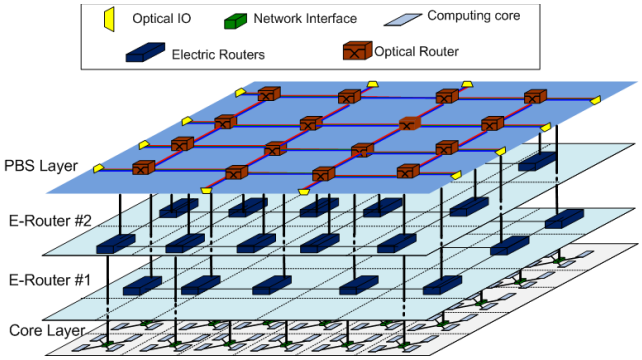


Fig. 1 The hybrid architecture.

separate networks for different use and result in our hybrid network architecture. Figure 1 illustrates the hybrid network. Computing cores take up the bottom layer, as it is close to heat sink and such floor plan is beneficial for heat dissipations. The PBS network used for off-chip memory access is placed on the top layer in a mesh topology with peripheral optical IOs. Memory reads and writes are relayed to this layer by underlying electric routers. An electric FBT for inter-core communication is embedded on multiple dedicated layers. The hybrid network adopts a 3D architecture, since 3D-I technology seems the only viable platform for heterogeneous integration [5], such as RAM, CMOS logic gates and photonic devices.

Our hybrid network can support scalable manycore chips in an energy-efficient manner because of the following three characteristics. That is,

Small-diameter Flattened Butterfly. End-to-end latency T of a packet using wormhole routing contains three parts: the header latency (T_h), the serialization latency (T_s), and the time of flight on the wires (T_w), as shown in Eq. (1),

$$T = T_h + T_s + T_w = H \times t_r + L/b + T_w \quad (1)$$

where t_r is the router delay, H is the hop count, L is the packet size, and b is the channel bandwidth. Minimizing T needs to balance header latency and serialization latency. Abundant on-chip wires will significantly reduce serialization latency by providing wide channels. However, the high hop-count in 2-D mesh/torus networks results in large latency when system scales. For example, in the 2-D mesh network used in the Intel TeraFlop [15], with uniform random traffic, header latency is approximately 10 times the serialization latency for worst case traffic. In contrast, our flattened butterfly has small network diameter since it essentially provides more links in addition to the connectivity of a mesh, assuming the bisection bandwidth is held constant. The hop count is thus greatly reduced. That is the reason why we choose flattened butterfly for inter-core communication.

Bandwidth-efficient Photonic Burst Switching. PBS is an adapted version of optical burst switching for chip-scale communication. It leverages high-speed low-power CMOS-compatible photonic devices to meet the enormous memory bandwidth demand and stringent energy constraints.

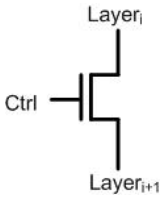


Fig. 2 Connection box.

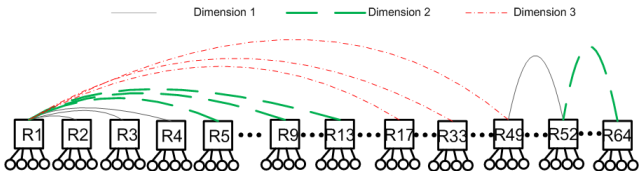


Fig. 3 A 4-ary 4-flat FBT (only the connections of R1 and the route from R1 to R64 are illustrated, others are omitted for clarity).

Compared with existing wavelength routing [8] and optical circuit-switching [11], [12], it uses sub-wavelength optical switching and one-way signaling for resource reservation and thus results in improved bandwidth efficiency. The high bandwidth utilization thus offset the static energy overhead of photonic devices. This is the primary motivation that we select PBS for power-hungry processor-memory network.

Concentration. Concentration can improve network efficiency, because the probability that more than one of the cores attached to a single router will attempt to access the network on a given cycle is relatively low. Our hybrid network thus uses concentration to aggregate traffic. In a k-ary n-flat flattened butterfly, k computing cores are connected to the same router (see Fig. 1, the concentration factor on the bottom layer is 4). PBS memory access network also uses concentration to increase bandwidth utilization, as described in Sect. 4.

The flattened butterfly is an efficient topology. However, on-chip FBT does not scale well because it suffers from high-radix routers and long wires which impose great concerns in area consumption and power overhead. Moreover, unlike mesh, adding nodes in the FBT requires the rewiring and layout of the entire chip, since a node has to be connected to each dimension (see Fig. 3). We may solve such concerns using 3D stacking technology. The reason we are able to build a low-latency 3D FBT topology lies in the great advantage of the vertical links, i.e., pillars that connect nodes on different dies (see Fig. 1). Interlayer pillars can be connected through a "connection box" (see Fig. 2) to build one-hop longer links which cross multiple layers [18]. As a result, the logical "long" wires between remote nodes are in fact physically very short in the 3D FBT topology. Furthermore, multiple stacking layers provide abundant area for high-radix routers. We developed the following integer linear programming (ILP) model to embed a FBT into the 3D architecture.

3.2 Embed a FBT into the 3D Topology Using ILP

We first present the theoretical design space. A k-ary n-flat flattened butterfly is composed of N/k radix $k' = n \times (k-1)$ +1routers where N is the size of the network. Routers are connected by links in $n'(n' = n-1)$ dimensions. To build a FBT with radix-k routers, the smallest dimension n' should meet the scaling requirement, i.e.,

$$\left\lfloor \frac{k}{n'+1} \right\rfloor^{(n'+1)} \geq N \tag{2}$$

Table 1 Solution space for a 256-core flattened butterfly.

Size	Stacking Layers(L)	Inter-router Ports(R)	Concentration Factor
256	8	7	2
256	4	9	4
256	8	12	5

Based on the value of n' selected, the resulting effective radix k' of the topology is

$$k' = \left(\left\lfloor \frac{k}{n'+1} \right\rfloor - 1 \right)(n'+1) + 1 \tag{3}$$

Therefore, given the size N fixed, a network has many combination pairs of dimensionality and radix (n', k'), which forms a group G $(G = \{(n'_1, k'_1), (n'_2, k'_2) \ldots\})$. We define two groups here, dimensionality group $G_{n'} = \{n'_1, n'_2 \ldots\}$ and radix group $Gk' = \{k'_1, k'_2 \ldots\}$ for future use.

For each (n', k'), we construct a unique $(N/k) \times (N/k) \times n'$ connection matrix $C_{n'}$, where its element $c(i, j, d)$ represents whether router i and router j is connected in dimension d. $c(i, j, d)$ is given by (4),

$$c(i,j,d) = \left(j == i + \left[m - \left(\left\lfloor \frac{i-1}{k^{d-1}} \right\rfloor \bmod k \right) \right] k^{d-1} \right) ?1 : 0 \tag{4}$$

for m from 0 to $k-1$, where the connection from i to itself is omitted. Let L be the number of stack layers, M be the FBT routers per layer, and R be the router's port for inter-router links, $R = k' - k$. Then, the total number of inter-router links must match. So, we have

$$\frac{1}{2} \sum_{d=1}^{n-1} \sum_{j=1}^{N/k} \sum_{i=1}^{N/k} c(i,j,d) = LMR/2 \tag{5}$$

Using these equations, we can obtain a set of theoretical solutions. For example, we can decompose a FBT network with 256 cores into the three entries in Table 1, assuming 16 routers per layer. They all satisfy the above equations seemingly. However, some of them are far from feasible, because many important physical constraints are neglected, such as stacking layer depth, wiring area, long wires, and vertical pillars. In this section, we try to achieve an optimum tradeoff among all the constraints by using an Integer Linear Programming model to explore the design space thoroughly.

The ILP Problem

Theoretical analysis shows that our topology can be developed given the router port constraints. In practice, area

overhead and wiring complexity of long links are more important constraints for on-chip design. Moreover, the stacking depth is also limited in 3D integration technology. The ITRS [19] projected that the number of dies that can be stacked will exceed 11 by 2012. Hence, if the required layer count is larger than this value, we can only incorporate a subset of the links. If we use less layers, then the amount of long links and wiring complexity for each layer will increase, which will lead to unaffordable thermal problem and lower working frequency. Therefore, our goal is to select the minimum stacking layers that satisfy the wiring area, router radix, and long wires constraints while supporting the given network size. This selection process can be carried systematically through ILP model, which is a powerful method for minimizing (maximizing) certain objective through determining a set of decision variables, subject to some constraints. We now discuss three main ILP components: decision variables, objective function, and constraints.

Decision variables

Our decision variables are Boolean variables $X_{l,i,n'}^{(x,y)}$ representing whether a router i is placed at (x, y) on a stacking layer l for a given combination pair (n', k'). For example, in Fig. 4, router R1 is placed on layer 2 at $(1, 1)$, so $X_{2,1,3}^{(1,1)} = 1$ and $X_{l,1,3}^{(x,y)} (l \neq 2)$ is 0. For our decision variables, we have

$$0 \leq X_{l,i,n'}^{(x,y)} \leq 1, \quad (1 \leq l, 1 \leq i \leq N/k, n' \in G_{n'}) \tag{6}$$

Once assigned to a layer, FBT routers get unique coordinates (x, y, l) each. The physical distance $D(i, j)$ between routers i and j, which contains two parts (planar distance D_{planar} and vertical distance $D_{vertical}$), determines the layout and wiring of their links. We have:

$$\begin{aligned} D(i,j) &\stackrel{\text{def}}{=} \left(D_{planar}, D_{vertical}\right) \\ &= \left(\left|x_i - x_j\right| + \left|y_i - y_j\right|, \left|l_i - l_j\right|\right) \end{aligned} \tag{7}$$

Objective function

As discussed earlier, our objective is to minimize the stacking layers used for FBT. Let $Y_{l,n'}$ denotes whether any router has been assigned on the layer l. If l is not occupied, then $Y_{l,n'}$ equals 0; otherwise, it equals 1. Therefore, we have:

$$Y_{l,n'} = 1 - \prod_{i,(x,y)} \left(1 - X_{l,i,n'}^{(x,y)}\right) \tag{8}$$

The objective function can be expressed as:

$$\min_{n' \in G_{n'}} \sum_{l=1} Y_{l,n'} \tag{9}$$

Constraints

We first discuss the area constraints for wiring long links, and then summarize mathematically the rest constraints that we mentioned earlier.

Wiring Area. The number of wires that can be routed is limited by the size of the router as well as the area taken by the wires. The wiring density refers to the maximum number of tile-to-tile wires routable across a tile edge [20]. Therefore, given the size of a router, the number of global wires that can be routed through the router is fixed. Studies show that global wires consume 4 to 8 times the area (width+spacing) of short local wires [21]. In our topology, wires between router i and j may cross multiple layers or be in one planar layer depending on $D(i, j)$. For planar wires, the link spanning one tile is definitely a short wire. The 2 and 3-hop wires are considered as short and medium wires, and the rest ($D_{planar} \geq 4$) are considered as long wires. Cross-layer wires consist of two parts, vertical pillars and the planar wires. Pillars are modeled as one-hop short wires due to the small inter-layer distance of 50 μm, but we limit the maximum layers that a pillar can penetrate as explained later.

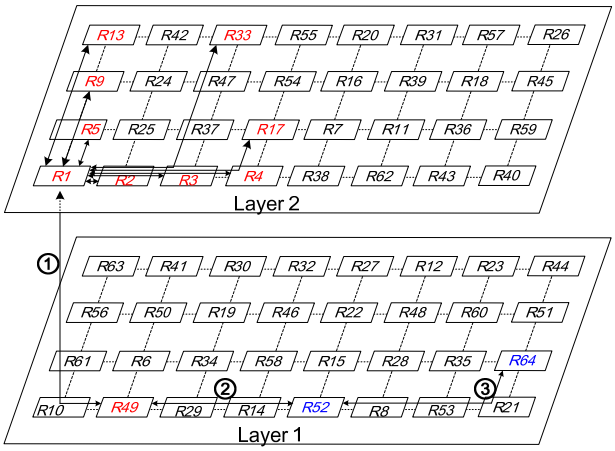


Fig. 4 A sample 3D FBT (only the wires of R1 and the route from R1 to R64 are illustrated, others are omitted for clarity).

Let $w_{i,j,l}$ denote the area taken on layer l by the wire from router i to j, $A_{\max}$ be the wiring density in unit of the area for one short wire. Let s denote a one hop segment between two neighboring routers, and S denote the union of them. Then the area constraint for wiring can be expressed as:

$$\sum_{\substack{all\ wires \\ pass\ thru.\ s}} w_{i,j,l} \leq A_{\max}, \quad \forall s \in S, \quad 1 \leq l < l_{\max} \tag{10}$$

The sum is expanded by examining the routing paths of all wires. The constant $w_{i,j,l}$ and $A_{\max}$ are determined as follows. If the distance D_{planar} between i and j is larger than 3, $w_{i,j,l}$ is 4× the area of a short wire. Otherwise it is 1×. For $A_{\max}$, we follow the model in [22] (in which $A_{\max}$ is 12 for 45 nm process) and consider the improvements of future generation process, therefore we assume $A_{\max} \leq 16$ for 22 nm process. That is, we can arrange up to 16× the area of a semi-global wire per hop.

Router Radix. Routers with small radix are preferred for on-chip design. Hence, we confine that the number of inter-router ports per router $(k' - k)$ should not exceed a reasonable maximal number $P_{\max}$, unless more pillars are used. We have

$$k' - k \leq P_{\max} \tag{11}$$

In our topology, $P_{\max}$ is 12 (8 intra-layer ports and 4 vertical ports). Previous researchers have found that the vertical communication in 3D chip is critical to network performance [18]. This also holds in our design, as we use vertical links to optimize long wires. Studies have shown that the state-of-the-art routers can have more than 30 to 100 pillars [23]. Hence, pillar density will likely not be a limiting factor for several generations. However, diminishing returns has been observed in [22] when they increased the pillars beyond 4. Therefore, we choose 4 as our maximal vertical ports for each router.

Vertical Pillars. For vertical pillars crossing multiple layers, we follow the scheme proposed in [22] to mitigate the voltage drop problem, which makes the logic 1 difficult to be recognized by the final stage logic. Hence, we assume that transmitting one bit on a vertical pillar cannot cross more than 6 stacked stages with an inter-layer wire length of 50 μm, which means:

$$D_{vertical} \leq 6 \tag{12}$$

Long wires. Global wires take longer time and more energy to carry signals than local (short) wires. This delay will affect the network clock frequency if the long wires take only one clock cycle to complete. Yi et al. [22] found that the slowest 6-hop long wire with a delay of 957 ps can well sustain a 1 GHz network, which means working at 1 GHz is sufficient to enable that every link requires only 1 clock cycle to transmit a signal. We consider this frequency reasonable since manycore chips suffer from thermal problems and higher frequency is not beneficial for heat dissipation. Hence, planar global wires should not exceed 6 hops. That is,

$$D_{planar} \leq 6 \tag{13}$$

Summary

Our ILP uses the Boolean variables $X_{l,i,n'}^{(x,y)}$ to indicate whether a router i is placed at (x, y) on a stacking layer l for a given combination pair (n', k'). The results of these variables are determined by evaluating the objective function (9) subject to the constraints (10), (11), (12), and (13). Our ILP is formulated using AMPL language [24] and solved using lpsolve [25].

3.3 A Sample 3D Flattened Butterfly

The original system contains 256 cores. After theoretical decomposition, we choose the (3, 13) dimension-radix pair as it needs less routers than others. Figure 3 shows its logical topology. According to connection matrix generated from (4), anyone of the 64 routers is connected with three other ones in each dimension, which totals up to 576 unidirectional wire bundles. Intuitively, it would be desirable and cost less if we can construct a planar flattened butterfly [7] and place all routers and wires therein. However, the planar scheme does not work in our exemplary network, because we find that a single layer under the 22 nm process can at most accommodate the FBT network for a system with around 100 cores, resulting to a shortage of space for those area-hungry high-radix routers and long wire bundles in large-scale manycore chips.

That is, the scalability limitation (as also mentioned by [7]) of single-layer flattened butterfly motivates the multi-layer 3D layout scheme for our exemplary 256-core system. Using our ILP, the FBT is embedded onto two layers, as shown in Fig. 4. There are 32 routers arranged in a 4×8 grid on each layer, which provides enough area for high-radix routers. The constraints here are in line with the previous discussion, e.g., the longest planar wire is only a 5-hop link (R1 $\rightarrow$ R33).

Our FBT uses dimension ordered routing, as shown in Fig. 4. For example, packets from R1 to R64 first pick up link ① in the third dimension and reach router R49. Then, they are forwarded through link ② in the first dimension to R52, and finally relayed to R64 through link ③ in the second dimension. In this way, anyone of the 256 cores can reach each other within at most 3 hops.

4. Photonic Burst-Switched Memory Access

4.1 Chip-Scale PBS Network

Another critical aspect for manycore processors is the off-chip memory access network. In our hybrid architecture, we use photonic burst switching for processor-memory network. As shown in Fig. 5, we arrange the PBS network into an optical mesh, as it is easy to layout and results in short wires without waveguide crossings. In the mesh, we connect the 5-port optical routers using wavelength division multiplexed waveguides. One of the ports is a PBS interface used for injection and ejection of local traffic. One port of the peripheral optical router is a memory access point (MAP), which is finally connected with off-chip DRAM modules. As explained earlier, the combination of high-speed photonic devices and bandwidth-efficient photonic burst switching offers a natural match in terms of bandwidth demand and energy budgets between increasing computing cores and memory system.

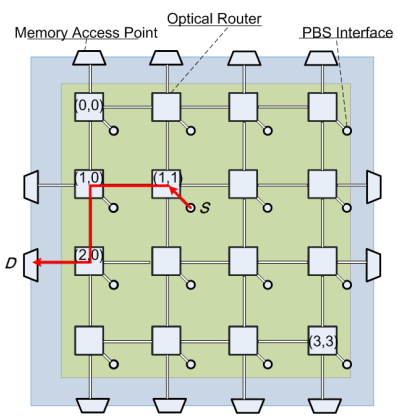


Fig. 5 The PBS network.

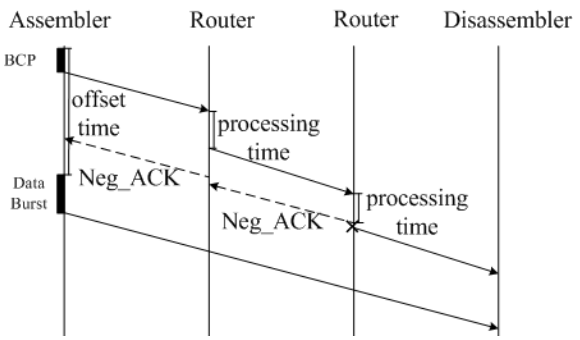


Fig. 6 The basic protocol of a PBS transaction.

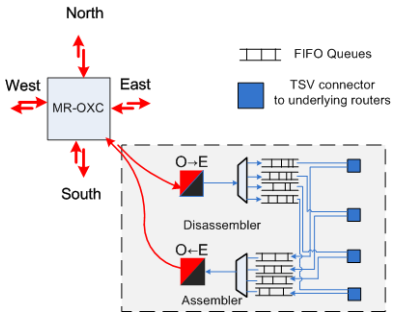


Fig. 7 A PBS interface.

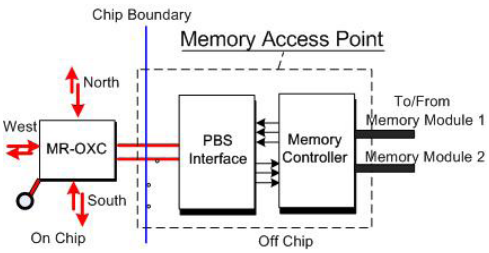


Fig. 8 A memory access point.

Our PBS scheme is an adapted version of optical burst switching [14] especially for chip-scale optical interconnection. In this scheme, for example, as shown in Fig. 5, the source PBS interface S first buffer the memory transaction packets from electric FBT routers and then assemble them into large bursts based on the address of memory module D. After that, a burst control packet (BCP) is created and sent by S an offset time before the burst (see Fig. 6). The BCP packet is electronically switched and processed at every intermediate optical router. It contains important information for resource reservation, including the burst arrival time, burst size, destination, which is used by optical routers to forward corresponding data burst. Without waiting for response about the successful reservation of a full path from S to D, data bursts are injected onto different wavelengths, optically switched by $(1,1)\rightarrow(1,0)\rightarrow(2,0)$ and ejected by the MAP D. Once failures happen, a Neg_ACK packet is transmitted back to the source, which starts the retransmission procedures. Retransmission guarantees reliable communication. Packet transmission from MAP to PBS interfaces works similarly.

In contrast to existing wavelength-routed [8] or optical circuit-switched schemes [11], [12], the major advantage of our PBS scheme comes from the one-way Just-Enough-Time (JET) [14] resource reservation signaling and sub-wavelength optical switching. First, data bursts are injected without waiting for successful reservation, which significantly reduces packets transmission latency. Second, the expensive wavelength channel does not need to be reserved exclusively from source to destination for the whole duration of packet transmission, which greatly improves bandwidth efficiency.

4.2 PBS Interface and Memory Access Point

The PBS interface (see Fig. 7) injects or ejects data bursts to/from optical network. The assembler module is responsible for assembling the incoming traffic into bursts and it uses a mixed timer-size scheme, which means that the burst is scheduled for transmission when at least one of the conditions is true: timer expiration or minimum burst length size reached. The disassembler module performs the inverse operation, breaking down the incoming bursts into packets and forwarding them.

As shown in Fig. 8, we place MAPs around the chip periphery to relay on-chip memory requests to off-chip DRAM modules and vice versa. Off-chip photonic IO signaling is achieved through lateral coupling by through inverse-taper optical mode converters, as it incurs lower insertion loss, compared to vertical coupling [6]. A MAP is essentially a memory controller augmented with a PBS interface. Upon a read request arrival, if another read transaction is currently in progress, this request is then queued up; otherwise, the memory controller issues DRAM commands to the memory module to fetch data. Data packets are then encapsulated into bursts by the MAP and sent back to the request issuer following PBS protocol. Writes begin by a core initiating a request to a MAP. Upon a write request arrival, if the memory module is servicing a read, this request is discarded and a negative acknowledgement is returned back; otherwise, the memory controller issues the normal DRAM commands sequence. Livelock can be avoided by using random backoff at the source core. However, starvation is possible, especially for writes in the presence of many reads. Addressing starvation remains a topic for future work.

4.3 A 5-Port Nonblocking Optical Router

The design of an efficient optical router is vital for our network because it implements the PBS protocol and routing functions. As shown in Fig. 9, our optical router, MR-OXC, is a strictly non-blocking 5x5 optical router with multi-wavelength routing capability. It consists of an optical switching fabric and a control unit. The switching fabric is based on two basic 1x2 optical switching elements, i.e., the crossing one and the parallel one (see Fig. 10). It uses 16

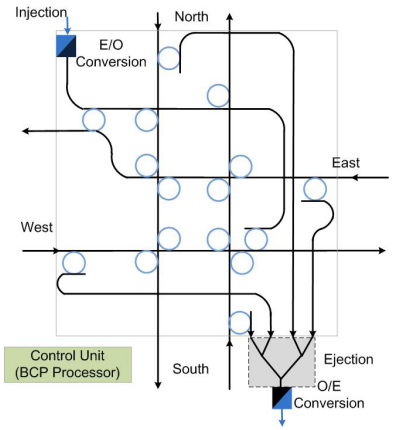


Fig. 9 Micro-architecture of the MR-OXC.

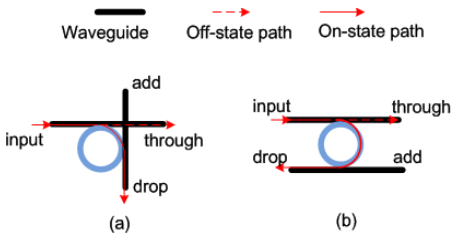


Fig. 10 1×2 optical switches. (a) Crossing. (b) Parallel.

Table 2 Performance comparison of optical routers.

Optical Router	Microring Resonators	Power efficiency (fj/bit)	Average Loss(dB)	Worst-case Loss (dB)
PR router	22	8.29	0.99	1.60
Optimized crossbar	25	9.60	1.08	1.58
MR-OXC	16	4.80	0.62	1.06

microresonators, 9 waveguides and 1 passive multiplexer to fulfill the 5x5 non-blocking optical switching function. The control unit is essentially a BCP packet processor, which uses electrical signals to configure the switching fabric according to the routing requirement of each BCP packet.

We compare MR-OXC with optimized crossbar router [17], and router proposed in [11] (we refer to it as PR router). As shown in Table 2, MR-OXC consumes fewer devices and has lower insertion loss. These advantages result from two design choices of MR-OXC. First, we prefer parallel 1x2 switching elements to crossing ones. This choice thus significant saves waveguide crossing losses. For example, the switching from north, east, west and south to the ejection port is only based on parallel 1x2 switches. Second, the diagonal layout of injection and ejection port further reduces crossings.

MR-OXC is especially beneficial to network scaling. With dimension order routing, no microresonator in the PBS mesh has to power on for data bursts that travel between south and north or between east and west. Only one microresonator is powered on when a burst is injected, ejected or makes a turn. This feature guarantees that the maximum power to route data bursts through the PBS mesh is a small constant number, regardless of the network size. This is because that networks built from MR-OXC only need to power on at most three microresonators to inject, turn, and eject a packet with dimension order routing.

5. Experimental Results

We used synthetic traffic and real workloads traces to evaluate the performance of the hybrid network. The simulation results highlight its advantages in terms of network latency and energy efficiency.

5.1 Simulation Setup

We extended the PhoenixSim simulator [13], which accurately captures the physical-layer aspects of the photonic devices. We named our hybrid network FBT-PBS, which was implemented and integrated into PhoenixSim.

Two typical mesh architectures, CMesh-PS and PMesh-CS, are selected as baseline network. CMesh-PS uses an electric packet-switched network for both on-chip and off-chip communication. PMesh-CS differs from CMesh-PS in that it uses optical circuit switching for memory access, which is similar to the one proposed in [11], [12]. In our FBT-PBS, the 3D flattened butterfly for intra-chip communication is built using the ILP in Sect. 3.2; the PBS memory access network uses the optical router in Sect. 4.3. The network size in our simulation is 256. In all three networks, 4 cores are connected to a single router.

Table 3 shows the more important simulation parameters that are used for simulations. CMesh-PS and FBT-PBS use DRAMsim [10] to model off-chip memory behaviors. PMesh-CS uses circuit-switched memory model DRAM_LRL [12]. Memory module parameters are extracted from a Micron 1-Gb DDR3 chip DRAM. For power dissipation modeling, the ORION 2.0 electronic router model [26] is integrated, in which the target technology generation is 22 nm. To study the advantages of photonic burst switching over optical circuit switching, the same parameters as the ones in [12] are used for silicon photonic devices.

Our architecture provides effective communication infrastructures for latency-sensitive and memory-intensive workloads, but few workloads exhibit both characteristics. Therefore, we first use synthetic traffic patterns to evaluate the on-chip inter-core network and the off-chip processor-memory network respectively, and then study the hybrid architecture as a whole with real application traces.

5.2 Results for Synthetic Traffic Patterns

The traffic patterns used in the simulations include the uniform, the hotspot and the tornado traffic [16]. In the uniform traffic pattern, each node sends packets to all other nodes with the same probability. In the hotspot traffic pattern, one or more nodes are designated as the hotspot nodes, which

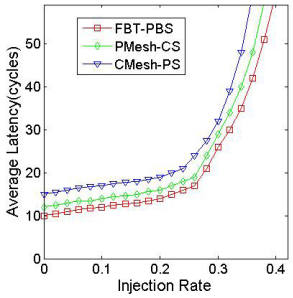
Table 3 Some important simulation parameters.

Parameter	CMesh-PS	PMesh-CS	FBT-PBS
Chip IO Parameters			
Physical I/O per MAP	64	2(w/128 λ)	2(w/32 λ)
I/O bit rate	1.6GTps	2.5Gbps	2.5Gbps
Electronic Inter-core Network Parameters			
Packet switched Clock Freq (GHz)	1.6	1	1
Buffer Size (b)	1024	1024	1024 (for FBT)
Virtual Channels	2	2	2(for FBT)
Electronic Channel Width	32	32	32
DRAM Parameters			
Base DRAM Frequency (MHz)	1066	1066	1066
Total Memory Per MAP(GB)	2	2	2
Bandwidth per DIMM (Gb/s)	128	320	320
Photonic Devices Parameters			
Microresonator switching time(ps)	NA	30	30
Dynamic energy of a switch element	NA	375 fJ	375 fJ
Static energy of a switch element	NA	400 μ J/sec	400 μ J/sec
Detector energy	NA	50fJ/bit	50fJ/bit

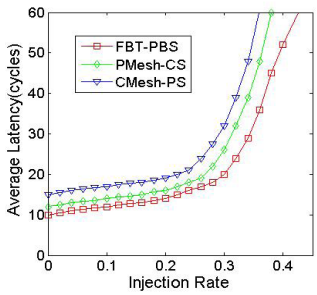
receive hotspot traffic in addition to the regular uniform traffic. In the tornado traffic pattern, the node (i, j) only sends packets to $(i, (j + \lceil k/2 \rceil - 1) \bmod k)$, where k is the network size. In our simulation, a node can be a computing core or a DRAM module.

We first generated synthetic traffics to evaluate the on-chip inter-core network. That is, no processor-memory traffic exists in this scenario. Here, we focus on the system throughput and latency. Figure 11 plots the average packet latencies for the three networks. The results show that FBT-PBS has noticeable improvement in both network latency and throughput. The zero-load latency for the Uniform Random traffic sees a 29.3% and 45.7% improvement over CMesh-PS and PMesh-CS respectively. The saturation point of our design is 7.8% (Tornado) and 11% (HotSpot) later than CMesh-PS, indicating a throughput improvement as well. These improvements result from the facts that: a) flattened butterfly essentially provides more links in addition to the connectivity of a mesh; b) our 3D FBT further reduces inter-core latency by using the express one-hop vertical interconnects.

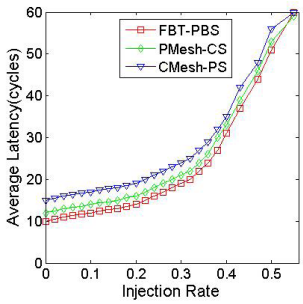
Next, we generated synthetic traffics to study the memory access network. Since computing cores and DRAM nodes have different addressing schemes, we used an address translator to perform the translation. Here, we focus on power because processor-memory network generally has stringent power budget. Figure 12 shows the metric of energy efficiency: performance gained for every unit of energy spent, which is effectively a measure of a network's efficiency. We selected the injection rate around the saturation point to increase link utilization.



(a) Hotspot



(b) Tornado

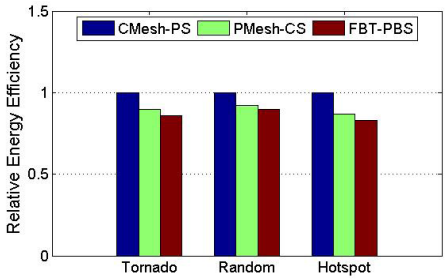


(c) Uniform Random.

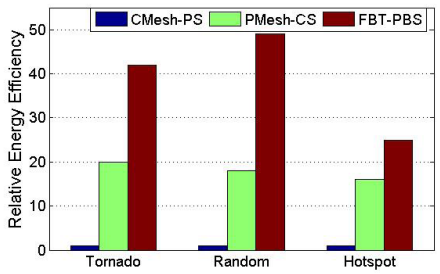
Fig. 11 Latency improvements for synthetic patterns (a) Hotspot (b) Tornado (c) Uniform Random.

As shown in Fig. 12 (a), the traffics with small messages perform poorly on both photonic processor-memory networks. The reason is that the static energy overhead of photonic devices cannot be compensated by small-size message transmission even we used a high injection rate. So, we enlarged the message size and simulated again. As a result, our photonic network, FBT-PBS, achieves the most noticeable improvements in energy efficiency (more than 40 times) for uniform random traffic (see Fig. 12 (b)), while its average transmission delay is very comparable to the results of packet-switched CMesh-PS (see Fig. 12 (c)). In contrast, the optical circuit-switched memory access scheme, PMesh-CS, also shows impressive improvements in energy savings, but it suffers the longest delay in all three traffic patterns.

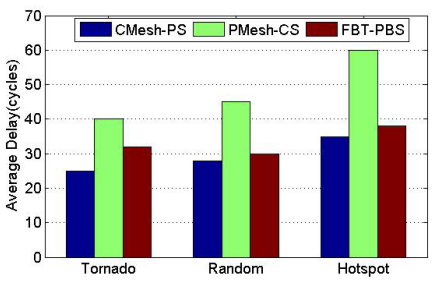
The reason for this is twofold. First, large bulk



(a) Relative energy savings for small size message.



(b) Relative energy savings for large size message.

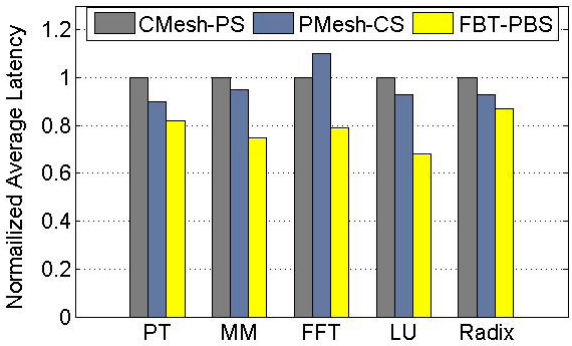


(c) Average transmission delay for large size message.

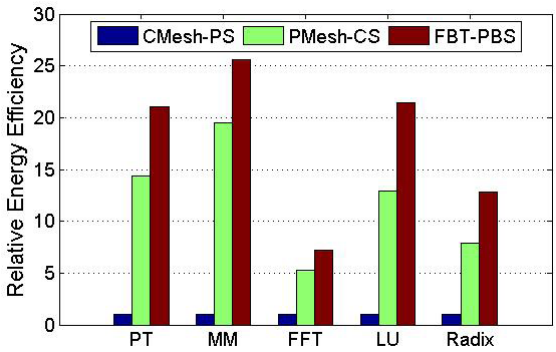
Fig. 12 Experiment results with synthetic patterns.

data movement potentially offsets the static power overhead discussed in Sect. 2.2, so both photonic memory access schemes, FBT-PBS and PMesh-CS, show higher energy efficiency than the electric scheme of CMesh-PS. Second, PMesh-CS uses the two-way resource reservation scheme in circuit switching which incurs large latency overhead due to resources contention; in contrast, FBT-PBS leverages the one-way signaling scheme and sub-wavelength switching. That is, unlike PMesh-CS, no wavelength channel in FBT-PBS has to be reserved exclusively from source to destination for the whole duration of packet transmission, resulting in improved bandwidth utilization and considerable reduction of latency overhead of control messages.

These simulation results demonstrate the possible appeal of PBS memory access network for many classes of applications, as random traffic is a common communication pattern.



(a) Latency reduction.



(b) Relative energy efficiency.

Fig. 13 Simulation results of real workloads (a) Latency reduction (b) Relative energy efficiency.

5.3 Results for Real Workload Traces

Five applications are considered: projective transform (PT), matrix multiply (MM), fast fourier transform (FFT), LU factorization (LU) and Radix integer sorting (Radix). FFT, LU and Radix come from the SPLASH-2 benchmark. We followed the methodology in [12] and used the MORE system to collect traces from the execution of these five kernel applications. MORE maps a user program written in Matlab onto a distributed or parallel architecture and translates application code into a dependency-based instruction trace, which captures the individual operations performed as well as their interdependencies. By reading the instruction trace into PhoenixSim, we were able to accurately model their executions on the three architectures. We also scaled up the default dataset appropriately to ensure that the statistical values of normalized latency and energy efficiency converge to a stable state after long simulation duration.

Figure 13 shows the results of normalized average packet latency and relative energy efficiency for the five workload traces. Interestingly, we found that among the three architectures, FBT-PBS consistently achieves the lowest packet latency and the highest energy efficiency on all kernel applications. Particularly, FBT-PBS shows a mod-

erate amount of latency reduction, i.e., 17.5% ∼ 25.9% against CMesh-PS and 5.3% ∼ 16.2% against PMesh-CS. It also shows impressive improvements in energy efficiency, i.e., 7 × ∼ 25× against CMesh-PS and 1.2 × ∼ 2.3× against PMesh-CS. However, these performance gains are not as profound as the simulation results of synthetic traffics. This is because that, although they also present the near uniform random characteristic, real workload traffics have lower injection rate which results in lower link utilization. Such findings confirm the need for large concentration factor in photonic networks.

Generally speaking, our simulation methodology suffers from some drawbacks. That is, cache hierarchies are not modeled, which are found to have important impacts on processor-memory network [3]. In our simulation, since cache hierarchies for manycore chips are still under study and we have no paradigms to follow, we used a simplified assumption that memory data are directed transmitted between cores and interconnection network. Nevertheless, using our physically-accurate network-level simulation environment, the simulation results still highlight the advantages of our hybrid architecture over conventional on-chip mesh and optical circuit-switched schemes.

6. Conclusions and Future Work

In this work, we study the problem of designing a interconnection network for manycore processors that supports low-latency on-chip communication and power-efficient off-chip memory access. We accomplish this by proposing a hybrid photonic burst-switched architecture, which leverages the recent advent of three dimensional integration [5] and silicon photonics [6] technology. The new architecture provides express inter-core communication by embedding a low-diameter flattened butterfly into the 3D topology. In addition, it uses the bandwidth-efficient photonic burst switching for memory access. We evaluated the new architecture using synthetic traffics and real workload traces on a physically-accurate network-level simulator. The results show that our hybrid network achieves considerable improvements in terms of network latency and power consumption, when compared to conventional on-chip mesh network and optical circuit-switched memory access scheme [12]. Other important aspects of the hybrid network, including heat dissipation, cache hierarchies and optimal read/write operations of DRAM interface for optical burst traffic flows are now under study and will be reported in our future work.

References

[1] T. Agerwala, "Exascale computing: The challenges and opportunities in the next decade," Proc. 16th IEEE International Symposium on High Performance Computer Architecture, p.1, 2010.

[2] Tilera Corporation, "TILE-Gx Processor Family," http://www.tilera.com/products/TILE-Gx.php, accessed Dec. 29, 2011.

[3] D. Sanchez, G. Michelogiannakis, and C. Kozyrakis, "An analysis of on-chip interconnection networks for large-scale chip multiprocessors," ACM Trans. Archit. Code Optim., vol.7, no.1, pp.4:1–4:28, April 2010.

[4] S. Borkar, "Thousand Core ChipsA Technology Perspective," Proc. 44th ACM/IEEE Design Automation Conference, pp.746–749, 2007.

[5] M. Koyanagi, T. Fukushima, and T. Tanaka, "Three-dimensional integration technology and integrated systems," Proc. Asia and South Pacific Design Automation Conference, pp.409–415, 2009.

[6] T. Barwicz, H. Byun, F. Gan, C.W. Holzwarth, M.A. Popovic, P.T. Rakich, M.R. Watts, E.P. Ippen, F.X. Kartner, H.I. Smith, J.S. Orcutt, R.J. Ram, V. Stojanovic, O.O. Olubuyide, J.L. Hoyt, S. Spector, M. Geis, M. Grein, T. Lyszczarz, and J.U. Yoon, "Silicon photonics for compact, energy-efficient interconnects," J. Opt. Netw., vol.6, no.1, pp.63–73, Jan. 2007.

[7] J. Kim, J. Balfour, and W.J. Dally, "Flattened butterfly topology for on-chip networks," Proc. 40th IEEE/ACM International Symposium on Microarchitecture, pp.172–182, 2007.

[8] D. Vantrease, R. Schreiber, M. Monchiero, M. McLaren, N.P. Jouppi, M. Fiorentino, A. Davis, N. Binkert, R.G. Beausoleil, and J.H. Ahn, "Corona: System implications of emerging nanophotonic technology," Proc. 35th International Symposium on Computer Architecture, pp.153–164, 2008.

[9] S. Bell, B. Edwards, J. Amann, R. Conlin, K. Joyce, V. Leung, J. MacKay, M. Reif, B. Liewei, J. Brown, M. Mattina, M. Chyi-Chang, C. Ramey, D. Wentzlaff, W. Anderson, E. Berger, N. Fairbanks, D. Khan, F. Montenegro, J. Stickney, and J. Zook, "Tile64 - processor: A 64-core soc with mesh interconnect," Proc. IEEE International Solid-State Circuits Conference, pp.88–89, 2008.

[10] D. Wang, B. Ganesh, N. Tuaycharoen, K. Baynes, A. Jaleel, and B. Jacob, "Dramsim: A memory system simulator," SIGARCH Comput. Archit. News, vol.33, no.4, pp.100–107, 2005.

[11] A. Shacham, K. Bergman, and L.P. Carloni, "Photonic networks-on-chip for future generations of chip multiprocessors," IEEE Trans. Comput., vol.57, no.9, pp.1246–1260, 2008.

[12] G. Hendry, E. Robinson, V. Gleyzer, J. Chan, L. Carloni, N. Bliss, and K. Bergman, "Circuit-switched memory access in photonic interconnection networks for high-performance embedded computing," Proc. International Conference for High Performance Computing, Networking, Storage and Analysis, pp.1–12, 2010.

[13] J. Chan, G. Hendry, A. Biberman, K. Bergman, and L.P. Carloni, "Phoenixsim: A simulator for physical-layer analysis of chip-scale photonic interconnection networks," Proc. Conference on Design, Automation & Test in Europe, pp.691–696, 2010.

[14] C.M. Qiao and M.S. Yoo, "Optical burst switching (obs) - a new paradigm for an optical internet," J. High Speed Netw., vol.8, no.1, pp.69–84, 1999.

[15] S. Vangal, J. Howard, G. Ruhl, S. Dighe, H. Wilson, J. Tschanz, D. Finan, P. Iyer, A. Singh, T. Jacob, S. Jain, S. Venkataraman, Y. Hoskote, and N. Borkar, "An 80-tile 1.28tflops network-on-chip in 65 nm cmos," Proc. IEEE International Solid-State Circuits Conference, pp.98–99, 2007.

[16] W.J. Dally and B. Towles, Principles and Practices of Interconnection Networks, Morgan Kaufmann Publishers, 2004.

[17] A.W. Poon, F. Xu, and X.S. Luo, "Cascaded active silicon microresonator array cross-connect circuits for wdm networks-on-chip," Proc. SPIE Int'l Soc. Opt. Eng., 2008.

[18] J. Kim, C. Nicopoulos, D. Park, R. Das, Y. Xie, N. Vijaykrishnan, M.S. Yousif, C.R. Das, and Acm, "A novel dimensionally-decomposed router for on-chip communication in 3D architectures," Proc. 34th Annual International Symposium on Computer Architecture, pp.138–149, 2007.

[19] ITRS 2009, "The International Technology Roadmap for Semiconductors," http://www.itrs.net/Links/2009ITRS/Home2009.htm, accessed Dec. 29, 2011.

[20] D.N. Jayasimha, B. Zafar, and Y. Hoskote, "On-Chip Interconnection Networks: Why They are Different and How to Compare them," http://blogs.intel.com/research/terascale/ODI_why-different.pdf, ac-

cessed Dec. 29, 2011.
[21] L.Q. Cheng, N. Muralimanohar, K. Ramani, R. Balasubramonian, J.B. Carter, and IEEE, "Interconnect-aware coherence protocols for chip multiprocessors," Proc. 33rd International Symposium on Computer Archtiecture, pp.339–350, 2006.
[22] X. Yi, D. Yu, Z. Bo, Z. Xiuyi, Z. Youtao, and Y. Jun, "A low-radix and low-diameter 3D interconnection network design," Proc. IEEE 15th International Symposium on High Performance Computer Architecture, pp.30–42, 2009.
[23] L. Feihui, C. Nicopoulos, T. Richardson, X. Yuan, V. Narayanan, and M. Kandemir, "Design and management of 3D chip multiprocessors using network-in-memory," Proc. 33rd International Symposium on Computer Architecture, pp.130–141, 2006.
[24] R. Fourer, D.M. Gay, and B.W. Kernighan. AMPL: A Modeling Language for Mathematical Programming, 2nd ed., Duxbury Press Publishing Company, 2002.
[25] M. Berkelaar, K. Eikland, and P. Notebeat, "LP solve: Opern Source (Mixed-Integer) Linear Programming System (2007)," http://lpsolve.sourceforge.net/5.5/, accessed Dec. 29, 2011.
[26] A.B. Kahng, L. Bin, P. Li-Shiuan, and K. Samadi, "Orion 2.0: A fast and accurate noc power and area model for early-stage design space exploration," Proc. Conference on Design, Automation & Test in Europe, pp.423–428, 2009.

Quanyou Feng was born in 1982. He received the B.S. degree in mechanics from Tsinghua University, Beijing, China, in 2005 and the M.S. degree in computer science and technology from National University of Defense Technology (NUDT), China, in 2008. He is currently working toward the Ph.D. degree in the School of Computer Science, NUDT. His research focuses on Network-on-Chip and high performance computing.

Huanzhong Li was born in 1984. He received the B.S. degree in computer science and technology from National University of Defense Technology (NUDT), China. He is currently working toward the Ph.D. degree in the School of Computer Science, NUDT. His research focuses on wireless network and network calculus.

Wenhua Dou was born in 1946. He received the B.S. degree in computer science from Harbin Military Engineering College in 1970. He has been working at National University of Defense Technology (NUDT), China since 1970. He was vice dean of School of Computer Science, NUDT from 1999 to 2003. He is currently a professor of School of Computer Science, NUDT with research focusing on computer architectures and computer networks.