Three-Dimensional Stacked Nanophotonic Network-on-Chip Architecture with Minimal Reconfiguration

Randy W. Morris Jr., *Student Member*, *IEEE*, Avinash Karanth Kodi, *Senior Member*, *IEEE*, Ahmed Louri, *Fellow*, *IEEE*, and Ralph D. Whaley Jr.

Abstract—As throughput, scalability, and energy efficiency in network-on-chips (NoCs) are becoming critical, there is a growing impetus to explore emerging technologies for implementing NoCs in future multicore and many-core architectures. Two disruptive technologies on the horizon are nanophotonic interconnects (NIs) and 3D stacking. NIs can deliver high on-chip bandwidth while delivering low energy/bit, thereby providing a reasonable performance-per-watt in the future. Three-dimensional stacking can reduce the interconnect distance and increase the bandwidth density by incorporating multiple communication layers. In this paper, we propose an architecture that combines NIs and 3D stacking to design an energy-efficient and reconfigurable NoC. We quantitatively compare the hardware complexity of the proposed topology to other nanophotonic networks in terms of hop count, network diameter, radix, and photonic parameters. To maximize performance, we also propose an efficient reconfiguration algorithm that dynamically reallocates channel bandwidth by adapting to traffic fluctuations. For 64-core reconfigured network, our simulation results indicate that the execution time can be reduced up to 25 percent for Splash-2, PARSEC, and SPEC CPU2006 benchmarks. Moreover, for a 256-core version of the proposed architecture, our simulation results indicate a throughput improvement of more than 25 percent and energy savings of 23 percent on synthetic traffic when compared to competitive on-chip electrical and optical networks.

Index Terms—Nanophotonics, CMP, 3D stacking, reconfigurable, NoC

1 Introduction

The ITRS roadmap predicts that complementary metal oxide semiconductor (CMOS) feature sizes will shrink from 32 nm to sub-17 nm within the next few years, thereby increasing the number of cores that can be integrated on a single chip [1]. Recent projections have shown that it will be possible to have as many as 256 cores on-chip by 2017 [2], [3]. As the design of the communication fabric has become more challenging, a growing number of multicore designs have adopted the network-on-chips (NoCs) design paradigm for enhancing scalability and improving reliability. While metallic interconnects can provide the required bandwidth due to shorter wires between cores as seen in NoCs, ensuring high-speed intercore communication within the allocated power budget in the face of technology scaling (and increased leakage currents) will become a major bottleneck for future multicore designs [4], [5]. Moreover, fundamental signaling limitations (reflections, crosstalk), electromagnetic interference (EMI), clock skew, and other problems associated with metallic interconnects will only exacerbate the power dissipation problem and thereby limit the performance of future multicores [5].

Nanophotonic interconnects (NIs) are under serious consideration for meeting the communication requirements of future multicores [2], [3]. NIs can provide several power-performance advantages that could prove to be critical for future on-chip communication such as higher bandwidth by multiplexing wavelengths on the same fiber/waveguide (wavelength-division multiplexing), increasing the bandwidth density by having multiple waveguides/fibers (space-division multiplexing), and reducing the energy/bit by dissipating energy only at the endpoints of the communication channel [2], [3], [6], [7], [8]. Although most prior designs have focused on 2D designs, signal paths can have a large number of waveguide crossings or require long waveguides for routing of packets from the source to destination. Waveguide crossings or intersections can result in significant power loss and backreflections due to the changes in the refractive index [7]. Another technology that is at the forefront for improving performance and reducing power consumption is 3D stacking. A common way to connect these layers vertically is using through silicon vias (TSVs) [9], [10], [11], [12]. The pitch of these vertical vias is very small ($4\ \mu$m-$10\ \mu$m), and delays of the order of 20 ps for a 20-layer stack. One of the critical challenges facing 3D integration is higher thermal and power dissipation density, which can be overcome with strategic placement of components and advanced cooling techniques [11].

In this paper, we exploit the advantages of two emerging technologies, NIs and 3D stacking with reconfiguration to

- *R.W. Morris Jr. is at 20420 NW Colonnade Dr., Hillsboro, OR 97124. E-mail: randy.w.morris@intel.com.*
- *A.K. Kodi and R.D. Whaley Jr. are with the Department of Electrical Engineering and Computer Science, Ohio University, Stocker Center, Athens, OH 45701. E-mail: {kodi, whaleyr}@ohio.edu.*
- *A. Louri is with the Department of Electrical and Computer Engineering University of Arizona, 1230 E Speedway Blvd., Tucson, AZ 85721. E-mail: louri@email.arizona.edu.*

Manuscript received 7 Apr. 2012; revised 9 July 2012; accepted 12 July 2012; published online 24 July 2012.
Recommended for acceptance by R. Melhem.
For information on obtaining reprints of this article, please send e-mail to: tc@computer.org, and reference IEEECS Log Number TC-2012-04-0255.
Digital Object Identifier no. 10.1109/TC.2012.183.

design a high-bandwidth and energy-efficient interconnect architecture called OCMP, an on-chip multilayer photonic NoC architecture. OCMP consists of 16 decomposed NI-based crossbars placed on four optical communication layers, thereby, eliminating waveguide crossings and the need for long snake-like waveguides resulting in a reduction in optical power losses. In addition, the static channel allocation (wavelengths, waveguides) proposed for OCMP can provide good performance for uniform traffic; however, for nonuniform and varying traffic, the static allocation could lead to network congestion. Therefore, to maximize the performance for varying applications with limited resources, we also propose dynamic reconfiguration by reallocating the available network bandwidth based on application demands. To limit the complexity of the reconfiguration algorithm, we restrict the bandwidth re-allocation only to the adjacent layers in our proposed architecture. This is accomplished by monitoring the traffic load and applying a reconfiguration algorithm that works in the background without disrupting the on-going communication. Our simulation results on 64-cores and 256-cores using synthetic traffic, SPEC CPU2006, Splash-2 [13], and PARSEC [14] benchmarks provide energy savings up to 23 percent and outperform other leading NIs up to 25 percent for adversial traffic via reconfiguration. Contributions of this work are as follows:

1. We propose a 3D stacked NIs that eliminate waveguide crossing and the need for long snake-like waveguides. The optical signals traverse multiple layers in OCMP using micro-ring resonators arranged in racetrack configuration that reduces TSVs and electrical power dissipation.
2. We quantitatively compare various nanophotonic topologies such as Corona, Firefly, and OCMP in terms of router complexity and photonic components.
3. We propose a reconfiguration algorithm that maximizes the available bandwidth by reconfiguring the network at runtime.
4. We evaluate our reconfiguration algorithm on synthetic traffic (uniform, permutation) as well as on real application traces collected via SIMICS and GEMS [15].

2 Related Work

In this section, we focus on the prior work on two important areas using NIs: 1) *architectures* and 2) *technology*. At the architecture level, there have been several NIs proposed that tackle different issues such as intercore communication, memory communication, and arbitration protocols [2], [3], [6], [8], [16], [17], [18], [19]. As this work is related to intercore communication, we restrict the discussion to few NIs. Shacham et al. [6] proposed a circuit-switch NI with electronic setup and photonic tear-down to optimize the power and performance for large-size packets seen in scientific applications. Vantrease et al. [2] proposed a 3D stacked 256-core NI to completely remove all electrical interconnect by designing an optical crossbar and token control. Due to sharing of resources, contention can be high as well as the cost and complexity of designing an optical crossbar for very high core counts. Firefly is an optoelectronic NI [8] that reduces the crossbar complexity of [2] by designing smaller optical crossbars connecting select clusters and implementing electrical interconnect within the cluster. While Firefly was able to reduce hardware cost, energy consumption increased due to increased electrical hops. In the more recent "macrochip" NI [16], multiple many-core chips have been integrated in a single package and multiphase arbitration protocols for communication have been proposed. Kirman et al. [19] have proposed an oblivious multilayer NI using torus topology that uses multiple nanophotonic layers for increased bandwidth. FlexiShare [18] is an optical crossbar that combines the advantages of both Corona (single-read, multiple-write) and Firefly (multiple-read, single-write). OCMP differs from FlexiShares as OCMP creates new communication channels within an existing framework. While FlexiShare is concerned with improving bandwidth in the time domain (more slots on more channels), R-OCMP (reconfigured-OCMP) improves performance on both space and time domains with a gradient of bandwidth (different percentages), which enables better overall performance. Recently, MPNoCs, a 3D NI [20], was proposed that uses multiple layers to create a crossbar with no optical waveguide crossover points. While MPNoCs uses TSVs for interlayer communication which consumes more power, OCMP is designed using microring resonators to traverse multiple layers. In addition, we uniquely improve the performance by implementing dynamic reconfiguration for synthetic as well as real applications.

On the technology front, most NIs adopt an external laser (whose power is included in the total power budget) and on-chip modulators. Micro-ring resonators (MRRs) have become a favorable choice due to smaller footprint ($10\ \mu$m), lower power dissipation (0.1 mW), high bandwidth (>10 Gbps), and low insertion loss (1 dB) [21], [3], [22], [23]. MRRs serve as modulators at the transmitter side and as filters at the receiver side. Complementary metal-oxide semiconductor (CMOS) compatible waveguides allow for signal propagation of on-chip light. Waveguides with micron-size cross-sections ($5.5\ \mu$m) and low-loss (1.3 dB/cm) have been demonstrated [21], [22], [23]. Recent work has shown the possibility of multiplexing 64 wavelengths within a single waveguide with 60 GHz spacing between wavelengths, although the demonstration was restricted to four wavelengths [3], [21]. An optical receiver performs the optical-to-electrical conversion of data, and consists of a photodetector, a transimpedance amplifier (TIA), and a voltage amplifier [24], [25]. Recently, a Si-CMOS-Amplifier with energy dissipation of about 100 fJ/bit and a data rate of 10 Gbps was demonstrated [24]. Recent advances have opened up the door to design 3D on-chip nanophotonic interconnects. Jalali's group at UCLA has fabricated a SIMOX (Separation by IMplantation of Oxygen) 3D sculpting to stack optical devices on top of each other [26] to create multilayer optical interconnects. Lipson group at Cornell has successfully buried active optical ring modulator in polycrystalline silicon [27]. Moreover, recent work on using silicon nitride has shown the possibility of designing multilayer 3D integration of photonic layers [28]. Clearly, there are several techniques of integrating optical components on multiple layers; one such technique will be described in the following section.

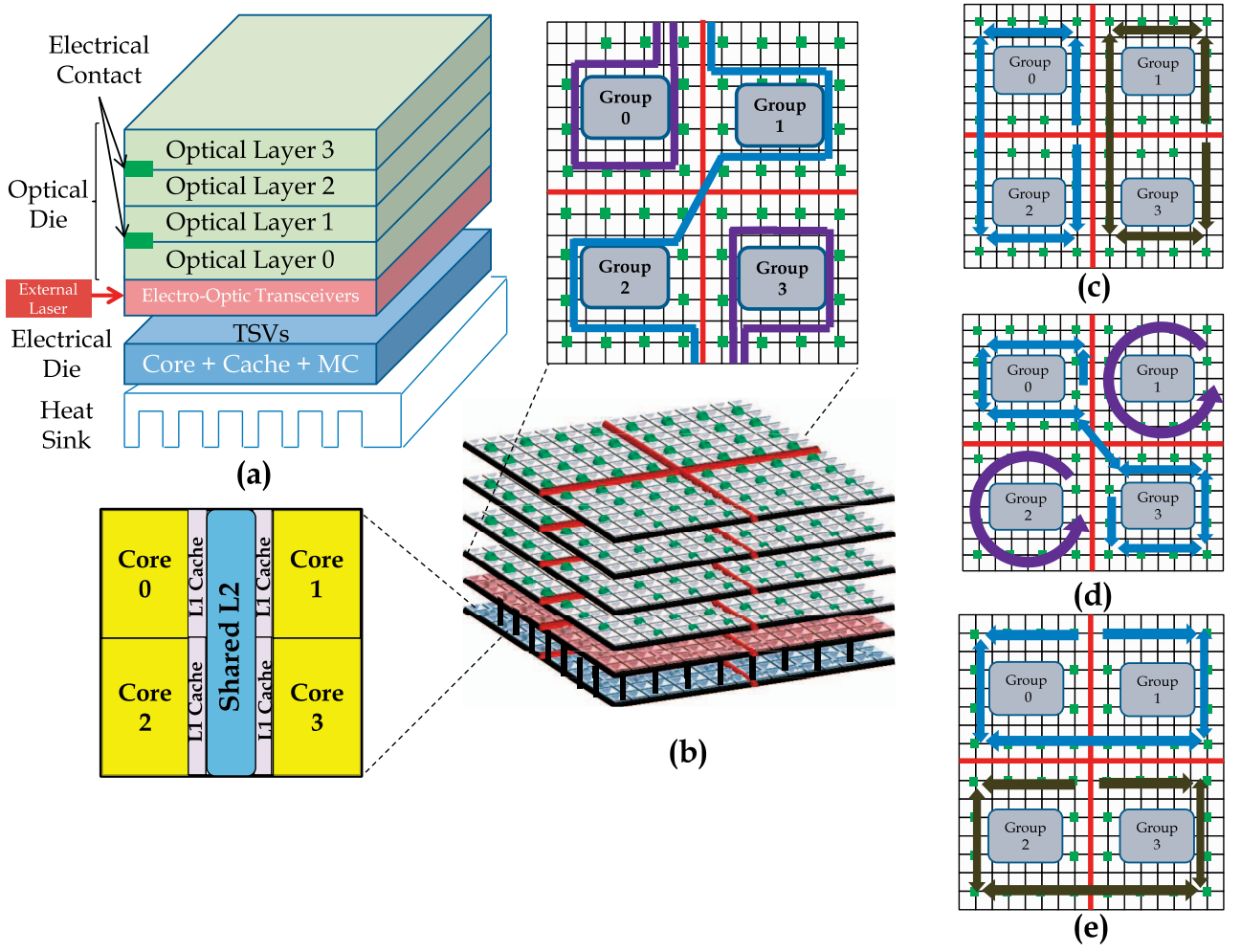

Fig. 1. Proposed 256-core 3D chip layout. (a) Electrical die consists of the core, caches, the memory controllers, and TSVs to transmit signals between the two dies. The optical die on the lower most layer contains the electro-optic transceivers and four optical layers. (b) 3D chip with four decomposed nanophotonic crossbars with the top inset showing the communication among one group (layer 0) and the bottom inset showing the tile with a shared cache and four cores. The decomposition, slicing, and mapping of the three additional optical layers: (c) optical layer 1, (d) optical layer 2, and (e) optical layer 3.

3 On-Chip Multilayer Photonic Architecture

The proposed OCMP architecture consists of 256 cores in 64 tile configuration on a $400\ \text{mm}^2$ 3D IC. As shown in Fig. 1, 256 cores are mapped on a 8×8 network with a concentration factor of 4, called a *tile* [29]. From Fig. 1a, the bottom layer, called the *electrical* die (adjacent to the heat sink), contains the cores, caches, and the memory controllers. Each core has its own private L1 cache and shared-L2 cache which connects four cores together to create a *tile*. The left inset shown in Fig. 1b illustrates a *tile*. The grouping of cores allows for a reduction in the cost of NIs as every core does not require lasers attached and more importantly, facilitates local tile communication through a shared-L2 cache. Each tile has a slice of shared-L2 cache along with directory information; memory addresses are interleaved across shared-L2. For 64-core version, we have 16 memory controllers located within the chip; as we scale to 256 cores, we can increase the number of memory controllers (this has not been modeled, as we assumed synthetic traffic for 256 cores). There are two key motivations for designing decomposed NI-based crossbars. First, an optical crossbar is desired to retain a one-hop network; however, a long winding waveguide connecting all the processors increases the signal attenuation (and thereby requires higher laser power to compensate). Second, decomposed crossbars on multiple layers eliminate waveguide crossings; this naturally reduces signal attenuation when compared to 2D networks where waveguide intersection losses can be a substantial overhead.

3.1 Proposed Implementation

To utilize the advantage of a vertical implementation of signal routing, we propose the use of separate optical and core/cache systems unified by a single set of connector vias. The upper die, called the *optical* die, consists of the electro-optic transceiver layer which is driven by the cores via TSVs and four decomposed nanophotonic crossbar layers. To this extent, electro-optic layer of the optical system contains all the optoelectronic components (modulators, detectors) required for the optical routing as well as the off-chip optical source coupling elements. Layers 0-3 contain optical signal routing elements, composed almost exclusively of MRRs and bus waveguides, with the exception of the electrical contacts required for ring heatings and reconfiguration (these are explained later). The top inset of Fig. 1b shows the interconnect for layer 0,

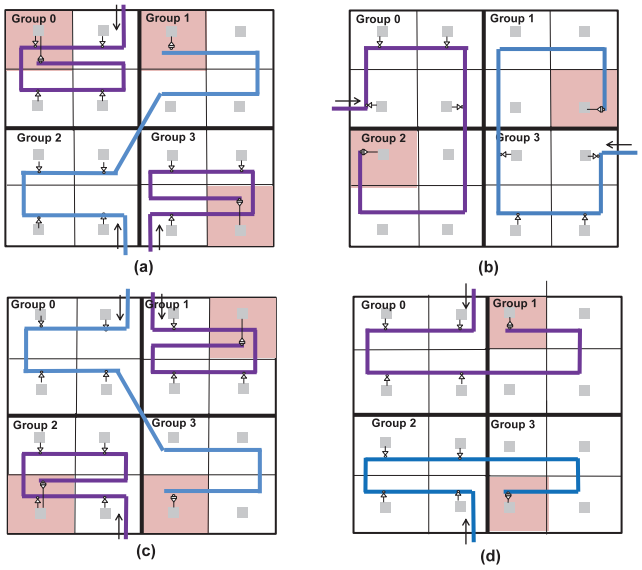

Fig. 2. The layout of different waveguides for (a) layer 0 communication, (b) layer 1 communication, (c) layer 2 communication, and (d) layer 3 communication.

whereas Figs. 1c, 1d, and 1e show layers 1-3. We determine the optimum number of optical layers by analyzing the requirement such that all groups can communicate while preventing waveguide crossing. For example, if Group 0/ Group 3 are connected together, then only intragroup communication between Group 1 and Group 2 can take place without waveguide crossing. Therefore, based on this reasoning, we concluded that we will require at least four optical layers. We also provide electrical contact between layers 0/1 and 2/3 to tune ring resonators required for reconfiguration. Vertical coupling of resonators can be very well controlled as intermediate layer thicknesses can be controlled to tens of nanometers [30].

The core region of the optical layer is composed of ZnO which is chosen due to its extreme low optical loss in the C-band region, high crystal quality at low deposition temperatures on amorphous substrates [31], high index of refraction (n-2 for $1.55\ \mu$m), high electro-optical coefficient for efficient modulation [32], and high process selectivity to standard CMOS materials. The fabrication of optical layers 0-3 follows a similar process of PECVD deposition of SiO_2, RF sputtering of ZnO, photolithography and etching to define resonators and waveguides, spin deposition of a planarizing compound (such as spin on glass (SOG), benzocyclobutene (BCB), or Cytop), and an O_2-plasma etchback for planarization. An electro-optic layer will also require the additional steps of e-beam Ge deposition and photolithography for the definition of the photodetectors, as well as the RF sputtering of indium gallium zinc oxide (IGZO) contacts for both the modulated rings and Ge detectors.

3.2 Quantitative Comparison: Corona, Firefly, and OCMP

In the proposed 3D layout, we divide tiles into four groups based on their physical location. Each group contains 16 tiles. Unlike the global 64×64 nanophotonic crossbar design in [2] and the hierarchical architecture in [8], OCMP consists of 16 decomposed individual nanophotonic crossbars mapped

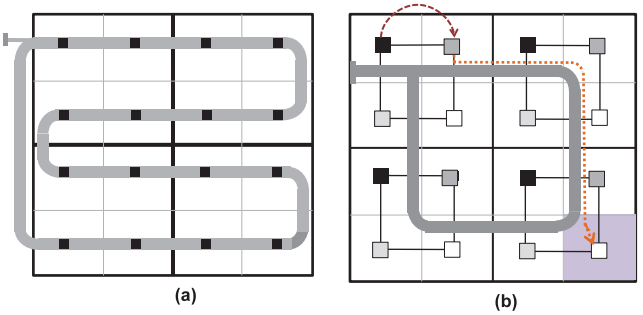

Fig. 3. The network layout for (a) Corona [2] and (b) Firefly [8].

on four optical layers as shown in Figs. 2a, 2b, 2c, and 2d. Each nanophotonic crossbar is a 16×16 crossbar connecting all tiles from one group to another (12 intergroup and 4 intragroup). It should be noted that the proposed architecture cannot be designed for arbitrary number of cores; OCMP is restricted to 64- and 256-core versions only. It is composed of many multiple-write-single-read (MWSR) nanophotonic channels. A MWSR nanophotonic channel allows multiple nodes the ability to write on the channel but only one node can read the channel, and therefore requires arbitration. If the arbitration is not fair (early nodes have more priority than later nodes), then latency and starvation could become a problem. On the other hand, a single-write-multiple-read (SWMR) channel allows only one node the ability to write on the channel but multiple nodes can read the data, and therefore requires efficient signal splitters and more power. A reservation-assisted subnetwork has been proposed in Firefly [8], where MRRs divert light only to those nodes that require the data; thus, SWMR can reduce the power but comes at a price of higher complexity and cost. Therefore, in this paper, we adopt MWSR combined with a token slot [2] to improve the arbitration efficiency and implement a fair-sharing of the communication channels.

To further illustrate the difference between OCMP and other leading nanophotonic networks, Fig. 3 illustrates the topologies of Corona and Firefly. In Fig. 3a, the optical crossbar topology for a 64-core version of Corona is shown [2]. Each waveguide in Corona traverses around all the tiles, where every tile can write onto a waveguide but only one tile can read a waveguide. In Fig. 3b, the optical topology for a 64-core version of Firefly is shown [8]. Firefly concentrates few local tiles into a group using an electrical Mesh network and then the same numbered tiles in different grffoups are connected using photonic SWMR interconnects. Table 1 shows the optical device requirements for 64- and 256-core versions of Corona, Firefly, and OCMP. As Firefly is an optoelectronic network, the hop count is more than either Corona or OCMP, but reduces the radix of the network from 8 to 6. OCMP requires the maximum number of ring resonators due to multiple layers that are designed to prevent waveguide crossings. Firefly requires the most photodetectors because each tile can receive data from four other tiles simultaneously. The last rows indicate the performance metric of average hop count and energy from running uniform traffic (more details are given in Section 5). OCMP and Corona are single hop networks; however, because of the decomposition, OCMP offers lower energy for communication.

TABLE 1
Topology Comparison for Different Network Sizes, Radices, Concentration, and Diameter, Where w and k Indicate the Number of Wavelengths and Radix of the Switch, Respectively

Component	Corona			Firefly			OCMP		
Network size (N)	-	64	256	-	64	256		64	256
Network Radix (k)	-	4	8	-	4	6	-	4	8
Concentration	-	4	4	-	4	4	-	4	4
Network Diameter	1	1	1	(k/2)+1	3	5	1	1	1
Wavelengths (w)	-	64	64		64	64	-	64	64
MRRs	$4wk^4$	65K	1048K	$4wk^3$	16K	131K	$4wk^4 + 12wk^2$	77K	1097K
Photo-detectors	$4wk^2$	4K	16K	$4w(k-1)k^2$	12K	114K	$16wk^2$	16K	65K
Electrical Links	-	-	-	k^2	16	64	-	-	-
Bisection Bandwidth	$4wk^2$	4K	16K	$4wk^2$	4K	16K	$16wk^2$	16K	65K
Bandwidth/channel	-	256	256	-	256	256	-	256	256
Average Hops	-	1	1	-	1.75	3.63	-	1	1
Average Energy	-	1.26	1.41	-	1.28	1.74	-	1.175	1.25

3.3 Intra- and Intergroup Communication

Each waveguide used within a nanophotonic crossbar has only one receiver which we define as receivers *home channel*. During communication, the source tile sends packets to their destination tile by modulating the light on the home channel of the destination tile. An off-chip laser generates the required 64 continuous wavelengths, $\bigwedge = \lambda_0, \lambda_1, \lambda_2....\lambda_{63}$. For optical layer 0, a 32 waveguide bundle is used for communication between Groups 1 and 2 and two 16 waveguide bundles are used for communication within Groups 0 and 3. For intergroup communication between 1 and 2, the first 16 waveguide bundle is routed past Group 1 tiles so that any tile within Group 1 can transmit data to any destination tile in Group 2. Similarly, the next 16 waveguide bundle is routed past Group 2, so that any tile within Group 2 can communicate with a destination tile located within Group 1. The bidirectional arrows illustrate that light travels in both directions and depends on which group is the source and the destination. The remaining two independent waveguide bundles (16 waveguides) are used for intragroup communication for Groups 0 and 3, respectively. Therefore, we require a total of 64 waveguide bundle per layer. A detailed decomposition and slicing of the crossbar on the other three layers is shown in Figs. 1c, 1d, and 1e. To further illustrate a decomposed crossbar optical layer, Figs. 2a, 2b, 2c, and 2d shows three waveguide layouts for layer 0-layer 3. Each group in the figure only has four tiles and only three waveguides are shown for clarity. The two intragroup waveguides are used for the four tiles in Group 0 and the four tiles in Group 3 to communicate with each other. These waveguides are first routed past the four tiles in the group which allows the tiles to write onto the waveguide. Then the waveguide is routed back to the tile that will read the optical data. For the waveguide in Group 0, the top left tile reads the data and for the waveguide in Group 3, the bottom right tile reads the data. The third waveguide in the figure shows the intergroup communication, where Group 2 tiles can communicate with the top left tile in Group 1.

4 Reconfiguration

As future multicores will run diverse scientific and commercial applications, networks that can adapt to communication traffic at runtime will maximize the available resources while simultaneously improving the performance. To implement reconfiguration, we propose to include additional MRRs that can switch the wavelengths from different layers to create a reconfigurable network. These MRRs are placed at points where the waveguide bundles from different layers will be in close proximity to each other. Furthermore, we also propose a reconfiguration algorithm to monitor traffic load and dynamically adjust the bandwidth by reallocating excess bandwidth from underutilized links to overutilized links.

4.1 Implementation

While reconfiguration can improve performance, it is essential to reduce the redundancy of components that are needed to achieve reconfiguration. Therefore, dynamic reconfiguration in R-OCMP will be limited to adjacent communication layers where bandwidth from one layer will be routed to another under different traffic and load conditions. Due to hardware and reconfiguration complexity, we restrict the reconfiguration that can take place in layers 0/1 and layers 2/3. MRRs are placed between the two layers (0/1 and 2/3) at locations where the waveguides are routed above each other. These micro-ring resonators, when activated, will switch data from one waveguide to another in a racetrack configuration. At every switch point, we require the entire wavelength bundle that can switch from one layer to the next.

To illustrate with an example, consider a situation where tiles in Group 0 communicate only with tiles in Group 3. Fig. 4 shows the reconfiguration mechanism. The *static* allocation of channel for communication are in layer 2 as shown in Fig. 4a. Suppose no tile within Group 1 (in layer 1) communicates with Group 3, then we can reallocate the bandwidth from Group 1 to Group 0 to communicate with Group 3. To implement reconfiguration, however, we need to satisfy two important requirements: 1) There should be a

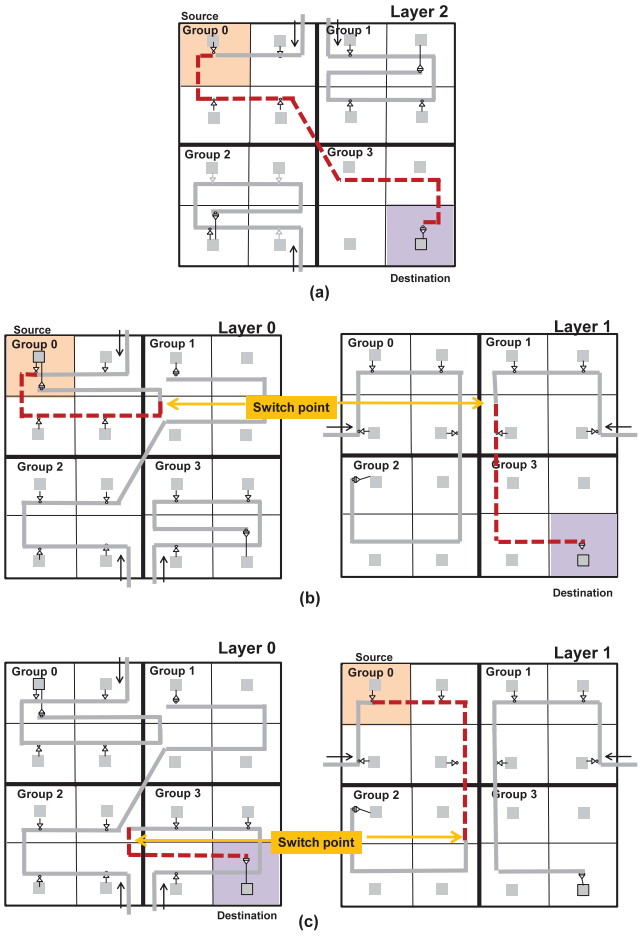

Fig. 4. (a) Static communication between the source in Group 0 and destination in Group 3. (b) Illustration of reconfiguration between Groups 0 and 3 using partial waveguides from layers 0 and 1, and (c) alternate reconfiguration between Groups 0 and 3 using partial waveguides from layers 1 and 0.

source waveguide which should be freely available to start the communication on a source layer, and 2) there should be a destination waveguide which also should be freely available to receive the extra packets. As shown in Fig. 4b, as the two Groups 0 and 3 talk only to each other, we have the first set of waveguides on layer 0 (generally used to communicate within the group) available; therefore, this satisfies the first condition. As Group 1 does not communicate with Group 3, we can utilize the destination waveguide available in layer 1 and this satisfies the second condition. The signal originates on layer 0, and then switches to layer 1 to reach the destination. Note that this additional channel is available in addition to layer 2 static configuration, thereby doubling the bandwidth. In addition, if need be, one more reconfiguration can occur as shown in Fig. 4c where Group 0 can partially use the waveguide statically used to communicate with Group 2 on layer 1 and switch to Group 3 (communicating with itself) on layer 0. Therefore, at most, we can increase the bandwidth to $3\times$ between two layers. As we restrict the reconfiguration to adjacent layers of 0/1 and 2/3, we are limited to two additional reconfiguration possibilities, i.e., Group 0 communicates with Group 3 on layer 2, which removes layer 3 as a reconfiguration choice, and confines reconfiguration between Group 0 and 3 only to layers $0->1$ and $1->0$ as shown in Figs. 4b and 4c. The only way to include layer 3 into the reconfiguration scenario will be to remove some of the bandwidth from layer 2 (static) and allocate which does not actually improve the communication bandwidth. Therefore, the maximum bandwidth that we can reallocate is restricted to $3\times$ (one static and two dynamic). Table 2 shows all combinations that are possible between layers 0 and 1 in R-OCMP. The first column represents the nodes within the group that are requesting additional bandwidth. The second column represents the possible destination group. The third column indicates the availability of source waveguides and the fourth column indicates the destination waveguides required to implement reconfiguration. In future work, we will expand the problem size to include any-group to any-layer reconfiguration.

4.2 Dynamic Reconfiguration Technique

In R-OCMP, reconfiguration algorithm reallocates bandwidth based on historical information. Historical statistics such as link utilization (Link_{util}) and buffer utilization (Buffer_{util}) are collected at the optical receiver of every communication channel by hardware counters [33]. This implies that each tile within a group will have four

TABLE 2
Reconfiguration Combinations for Layer 0/Layer 1

Source	Destination	Source Waveguides's Destination	Destination Waveguide's Source
Layer 0 $\longrightarrow$ Layer 1			
Group 0	Group 1/Group 3	Group 0	Group 3/Group 1
Group 1	Group 1/Group 0	Group 2	Group 3/Group 2
Group 2	Group 2/Group 3	Group 1	Group 0/Group 1
Group 3	Group 0/Group 2	Group 3	Group 2/Group 0
Layer 1 $\longrightarrow$ Layer 0			
Group 0	Group 3/Group 1	Group 2	Group 3/Group 2
Group 1	Group 1/Group 0	Group 3	Group 2/Group 0
Group 2	Group 2/Group 3	Group 0	Group 1/Group 3
Group 3	Group 2/Group 0	Group 1	Group 1/Group 0

TABLE 3
Reconfiguration Algorithm for OCMP

Step 1:	Wait for Reconfiguration window, $R_W{}^t$
Step 2:	RC_i sends a request packet to all local tiles requesting $Link_{Util}$ and $Buffer_{Util}$ for previous $R_W{}^{t-1}$
Step 3:	Each hardware counter sends $Link_{Util}$ and $Buffer_{Util}$ statistics from the pervious $R_W{}^{t-1}$ to RC_i
Step 4:	RC_i classifies the link statistic for each hardware counter as: If $Link_{util}$ = 0.0 Not-Utilized: Use β_4 If $Link_{util} \leq L_{min}$ Under-Utilized: Use β_3 If $Link_{util} \geq L_{min}$ and $Buffer_{util} < B_{con}$ Normal-Utilized: Use β_2 If $Buffer_{util} > B_{con}$ Over-Utilized: Use β_1
Step 5:	Each RC_i sends bandwidth available information to RC_j, (i≠j).
Step 6:	If RC_j can use any of the free links then notify RC_i of their use, else RC_j will forward to next RC_j
Step 7a:	RC_i receives response back from RC_j and activates corresponding microrings
Step 7b:	RC_j notifies the tiles of additional bandwidth and RC_i notifies RC_j that the additional bandwidth is now available
Step 8:	Goto Step 1

hardware counters (one for each of the three groups) that will monitor traffic utilization. Both link and buffer utilization are used, as link utilization provides accurate information at low-medium network loads and buffer utilization provides accurate information regarding high network loads [33]. All these statistics are measured over a sampling time window called *reconfiguration window* or phase, R_W^t, where t represents the reconfiguration time number t. This sampling window impacts performance, as reconfiguring finely incurs latency penalty and reconfiguring coarsely may not adapt in time for traffic fluctuations. For calculation of $Link_{util}$ at configuration window t, we use the following equation:

$$Link_{util}^t = \frac{\sum_{cycle=1}^{R_W} Activity(cycle)}{R_W}, \quad (1)$$

where *Activity(cycle)* is 1 if a flit is transmitted on the link or 0 if no flit is transmitted on the link for a given cycle. For calculation of $Buffer_{util}$ at configuration window t, we use the following equation:

$$Buffer_{util}^t = \frac{\sum_{cycle=1}^{R_W} Occupy(cycle)/Total_{buffers}}{R_W}, \quad (2)$$

where *Occupy(cycle)* is the number of buffers occupied at each cycle and $Total_{buffers}$ is the total number of buffers available for the given link. When traffic fluctuates dynamically due to short-term bursty behavior, the buffers could fill up instantly. This can adversely impact the reconfiguration algorithm as it tries to reallocate the bandwidth faster leading to fluctuating bandwidth allocation. To prevent temporal and spatial traffic fluctuations affecting performance, we take a weighted average of current network statistics ($Link_{util}$ and $Buffer_{util}$). We calculate the $Buffer_{util}$ as follows:

$$Buffer_w^t = \frac{\sum_{Buffer_{util}^t \times weight + Buffer_{util}^{t-1}}}{weight + 1}, \quad (3)$$

where $weight$ is a weighting factor and we set this to 3 in our simulations [34].

After each R_W^t, each tile will gather its link statistics ($Link_{util}$ and $Buffer_{util}$) from the previous window R_W^{t-1} and send it to its local reconfiguration controller (RC) for analysis. We assume that tile 0 of every group gathers the statistics from the remaining tiles and this can be few bytes of information that is periodically transmitted. Next, when each $RC_i, (\forall\, i = 0, 1, 2, 3)$, has finished gathering link and buffer statistics from all its hardware controllers, each RC_i will evaluate the available bandwidth for each link depending on the $Link_{util}^{t-1}$ and $Buffer_{util}^{t-1}$ and will classify its available bandwidth into a select range of thresholds β_{1-4} corresponding to 0 percent, 25 percent, 50 percent, and 90 percent. We never allocate 100 percent of the bandwidth as the source group may have new packets to transmit when the destination tile before the next R_W. RC_i will send link information (availability) to its neighbor $RC_j (j \neq i)$. If RC_j needs the available bandwidth, RC_j will notify the source and the destination RCs so that they can switch the MRRs and inform the tiles locally of the availability. Once the source/destination RCs have switched their reconfiguration MRRs, RC_i will notify RC_j that the bandwidth is available for use. On the other hand, if a node within RC_i that throttled its bandwidth requires it back due to increase in network demand, RC_i will notify that it requires the bandwidth back and afterward will deactivate the corresponding MRRs. The above reconfiguration completes a three-way handshake, where RC_i first notifies RC_j, then RC_j notifies RC_i that RC_j will use the addition bandwidth, and finally RC_i notifies RC_j that the bandwidth can be used. Table 3 shows a pseudo-reconfiguration algorithm implemented in R-OCMP. We assume $Link_{util} = 0.0$

TABLE 4
Core and Cache Parameters Used for Splash-2, PARSEC, and SPEC CPU2006 Application on SIMICS Using GEMS

Parameter	Value
L1/L2 coherence	MOESI
L2 cache size/accoc	4MB/16-way
L2 cache line size	64
L2 access latency(cycles)	4
L1 cache/accoc	64KB/4-way
L1 cache line size	64
L1 access latency(cycles)	2
Core Frequency(GHz)	5
Threads(core)	2
Issue policy	In-order
Memory Size(GB)	4
Memory Controllers	16
Memory latency(cycle)	160
Directory latency(cycle)	80

to indicate if the link is not being used, $L_{min} = 0.10$ to indicate if the link is underutilized, $L_{min} = 0.25$ and $B_{con} = 0.25$ to indicate if the link is normal-utilized, and $B_{con} = 0.5$ to indicate that the link is overutilized [33].

5 Performance Evaluation

In this section, we evaluate the performance, power efficiency, and area overhead of R-OCMP when compared to competing electrical interconnects and NIs.

5.1 Simulation Setup

We first describe the simulation setup of the proposed architecture. Our simulator models in detail the router pipeline, arbitration, switching, and flow control. An aggressive single-cycle electrical router is applied in each tile and the flit transversal time is one cycle from the local core to electrical router [35]. As the delay of optical/electrical (O/E) and electrical/optical (E/O) conversion can be reduced to less than 100 ps, the total optical transmission latency is determined by physical location of source/destination pair (1-5 cycles) and two additional clock cycles for the conversion delay. In addition, a latency of 1 to 3 cycles was assumed for a tile to capture an optical token. We assume an input buffer of 16 flits with each flit consisting of 128 bits. The packet size is 4 flits which will be sufficient to fit a complete cache line of 64 bytes. We assume a supply voltage V_{dd} of 1.0 V and a router clock frequency of 5 GHz [2], [8].

We compare OCMP architecture to two other crossbar-like NIs, Corona [2] and Firefly [8], and two electrical interconnects (Mesh and Flattened-Butterfly) [36]. We implement all architectures such that four cores (one tile) are connected to a single router. We assume a token slot for both OCMP and Corona to pipeline the arbitration process to increase the efficiency. Multiple requests can be sent from the four local cores to optical channels to increase the arbitration efficiency. We use Fly_Src routing algorithm [8] for Firefly architectures, where intragroup communication via electrical Mesh is implemented first and then intergroup via NIs. For a fair comparison, we ensure that each communication channel in either electrical or optical network is 640 Gbps with 64 wavelengths. We also evaluate by reducing the channel bandwidth to 16 wavelengths and communication bandwidth limited to 160 Gbps. For open-loop measurement, the packet injection rate is varied from 0.1 to 0.9 of the network capacity, and packets are injected according to the Bernoulli process based on the given network load. We consider both uniform as well as permutation traffic such as bit-complement (bitcomp), bit-reversal (bitrev), transpose, butterfly, neighbor, and prefect shuffle traffic patterns.

For closed-loop measurement, we collect traces from real applications using the full execution-driven simulator SIMICS from WindRiver, with the memory package GEMS enabled [15]. We evaluate the performance of 64-core versions of the networks on Splash-2 [13], PARSEC [14], and SPEC CPU2006 workloads and 256-core version on synthetic and workload completion traffic (a mixture of synthetic traces). Table 4 shows the core and cache parameters used for Splash-2, PARSEC, and SPEC2006 workloads. For Splash-2 traffic, we assume the following kernels and workloads: FFT (16K particles), LU (512×512 with a block size of 16×16), radiosity (large room), raytrace (teapot), radix (1 million integers), ocean (258×258), FMM (16K particles), and water (512 molecules). We consider six PARSEC applications with medium inputs (blackscholes, facesim, fluidanimate, freqmin, streamcluster, ferret, and swaptions) and two workloads from SPEC CPU2006 (bzip and hmmer).

The sizes of miss status holding registers (MSHRs) and length of the reconfiguration window (R_W) were extracted by running one of the PARSEC benchmarks (*blackscholes*). We varied the size of MSHRs to determine the optimum size for performance. From our simulations, a MSHR value of 4 gives the best performance because a smaller MSHR value does not inject enough traffic into the network for the reconfiguration algorithm to improve the performance and a larger MSHR saturates the network as many packets are injected into the network. We keep this constant across all applications. We choose 1,300 cycles as the reconfiguration window size for our simulations as this provides the best performance. In addition, we assumed a 100 cycle latency for the reconfiguration to take place after each R_W (three-way handshake delay).

5.2 Simulation Results

5.2.1 Splash-2: 64 Cores

Fig. 5 shows the speedup for the Splash-2 applications [13]. From Fig. 5a, OCMP has about an average speedup of about 2.5 for each benchmark over the Mesh network for 64 wavelengths. In the water application, OCMP has the highest speed-up with a factor of over 3 relative to Mesh. This is a result of OCMP's decomposed crossbars allowing for fast arbitration of network resources (less contention) and the reduced hop count relative to the Mesh network. In Raytrace and FMM benchmarks, OCMP has the lowest speedup factor of 2.2, which is contributed to the higher local (few hops) traffic. Nearest-neighbor traffic creates more contention for optical tokens with locally concentrated destinations in OCMP. When OCMP is compared to

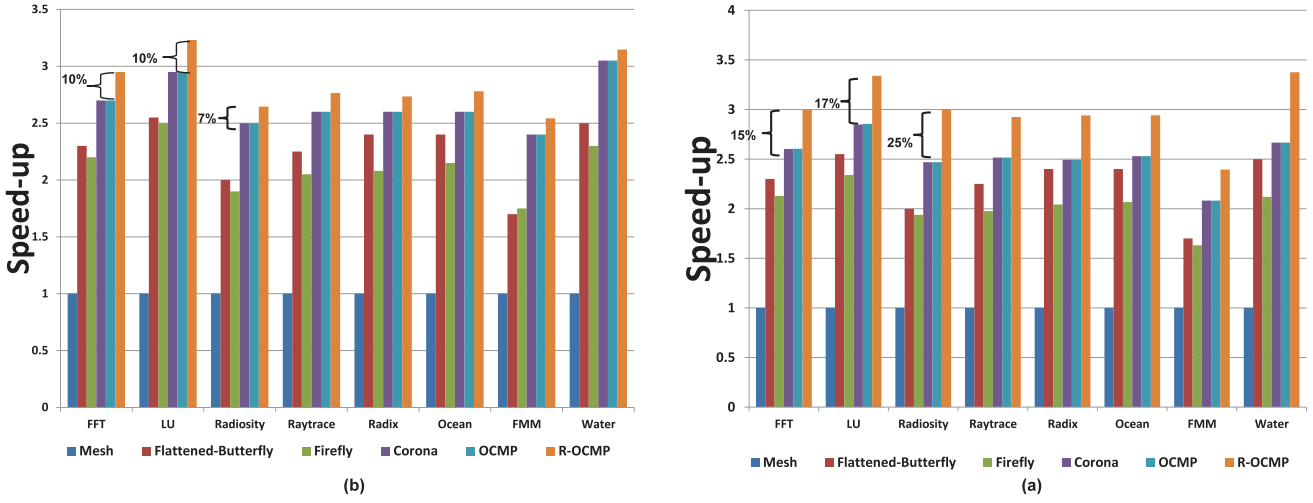

Fig. 5. Simulation speedup for 64-core using SPLASH-2 traffic traces with (a) 64 and (b) 16 wavelengths.

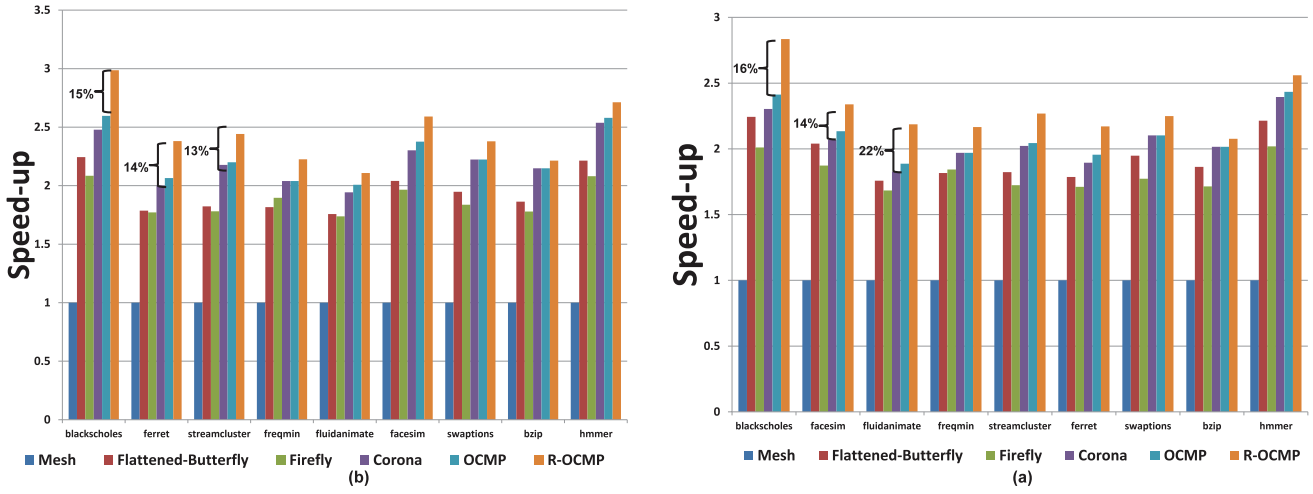

Fig. 6. Simulation speedup for 64-core using PARSEC traffic traces with (a) 64 and (b) 16 wavelengths.

Flattened-Butterfly, OCMP has a 25-30 percent improvement. As Flattened-Butterfly is a two-hop network and most traffic under Splash-2 suite is two hops, the intermediate router reduces the throughput of the network. When OCMP is compared to Firefly, OCMP outperforms Firefly by about as much as 38 percent which is a result of Firefly routing its traffic through both electrical and optical networks. As for Corona, both OCMP and Corona have similar SPLASH-2 traffic as both networks are similar in terms of their zero load latency. R-OCMP has about a 5-12 percent improvement over OCMP for the select range of Splash-2 traffic traces. For FFT and LU applications, R-OCMP has the highest performance improvement over OCMP at about 12 percent. Both FFT and LU have communication patterns that can take advantage of the reconfiguration algorithm as their communication patterns do not quickly change over time forcing the network to keep reconfiguring and improving performance. In the other applications (radiosity, raytrace, radix, ocean, rmm, and water), R-OCMP has about a 5 percent increase in performance over OCMP. This is a direct result of Splash-2 traffic traces resembling uniform traffic, which reduces the bandwidth available for reconfiguration. Moreover, as we simulate the application traces and not the actual application, the reconfigurations algorithm does not have enough time to adjust the bandwidth before the traffic patterns change again. Fig. 5b shows the results in a resource-constrained environment with only 16 wavelengths. As seen, the results show an average gain of over 25 percent across various applications. Clearly, as the available bandwidth is reduced, the performance gains increases due to more contention for limited bandwidth where our reconfiguration algorithm can allocate more spare idle channels.

5.2.2 PARSEC and SPEC CPU2006: 64 Cores

Fig. 6a shows the speedup for 64 wavelengths. OCMP shows an average of 2× speedup compared to Mesh and 10-40 percent improvement over Flattened-Butterfly and Firefly architectures. When Corona and OCMP are compared to each other, OCMP is able to outperform Corona for most applications except swaptions and bzip. The reason for improved performance over Corona is primarily due to the communication pattern which makes use of all the four decomposed crossbars to be used simultaneously, thereby sending more data on the network when compared to Corona. For swaption and bzip application traffic, their communication patterns do not take advantage of OCMP decomposed crossbars and as such there is no significant improvement. R-OCMP shows better improvement in

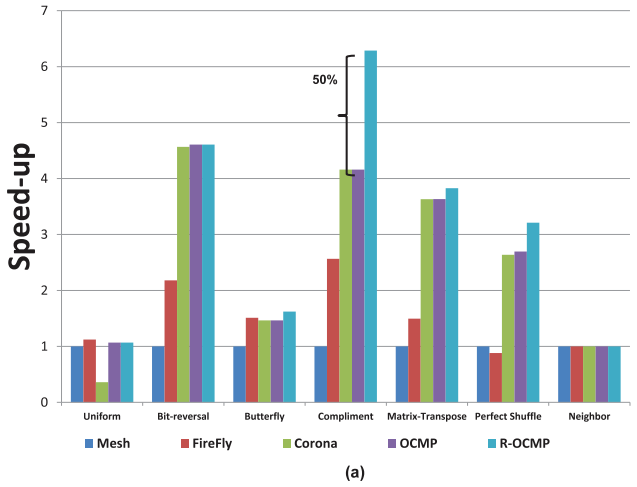

Fig. 7. Simulation results showing normalized saturation throughput for seven traffic patterns for 256 cores.

performance for PARSEC/SPEC CPU2006 benchmarks compared to Splash-2 traffic. Blackscholes has the largest jump in performance by almost 20 percent when compared to Corona and 15 percent when compared to OCMP. This large increase in performance is contributed to the nature of PARSEC applications which are more communication intensive, and, therefore, the reconfiguration algorithm maximizes the performance. PARSEC applications emphasize on emerging workloads and future shared-memory applications for the study of CMPs, rather than the network. Fig. 6b shows the results when the number of wavelengths is reduced to 16. The average speedup (geometric mean) for PARSEC benchmarks increases to a factor of 15 percent across all benchmarks when the wavelengths are reduced highlighting the importance of reconfiguration in a resource-constrained environment. It should be noted that we have focused on minimizing the design complexity and limiting the reconfiguration between only two layers. If all layers are involved in reconfiguration, then the performance improvement can be significantly higher as the entire system bandwidth can be reallocated.

5.2.3 Synthetic Traffic: 256 Cores

The throughput for all synthetic traffic traces for 256-core implementations are shown in Fig. 7 and is normalized to a Mesh network (for uniform, the Mesh has a throughput of 624 GBytes per sec). OCMP has about a $2.5\times$ increase in throughput over Corona for uniform traffic due to the decomposition of the nanophotonic crossbar. The decomposed crossbars allow for a reduction in contention for optical tokens as now a single token is shared between 16 tiles instead of 64 tiles as in Corona. Firefly slightly outperforms OCMP for uniform traffic due to the contention found in the decomposed nanophotonic crossbars. Moreover, Firefly uses a SWMR approach for communication which does not require optical arbitration. From the figure, OCMP slightly outperforms Corona for bit-reversal and complements traffic traces. This is due to lower contention for optical tokens in the decomposed crossbars. OCMP significantly outperforms Mesh for the bit-reversal, matrix-transpose, and complement traffic patterns. In these traffic patterns, packets need to traverse across multiple Mesh routers which in turn increases the packet latency and thereby reduces the throughput. When OCMP is compared to Firefly, OCMP outperforms Firefly by $2.5\times$. In Firefly, most traffic patterns will require packets to traverse across multiple electrical routers, and then traversal across an optical link resulting in a reduction in the number of packets that can be injected into the network as compared to OCMP. R-OCMP is able to outperform OCMP for complement, matrix-transpose, and perfect shuffle traffic traces. These permutation traffic traces exhibit adversial patterns which will benefit R-OCMP. In complement traffic, R-OCMP has about a 55 percent increase in performance when compared to OCMP. Complement traffic pattern showcasing the best performance as a single source tile will communicate with a single destination tile, thereby providing opportunities to improve performance via reconfiguration.

5.3 Energy Comparison

The energy consumption of an NI can be divided into two parts: electrical energy and optical energy. Optical energy consists of the off-chip laser energy and on-chip MRRs heating energy. In what follows, we first discuss the electrical energy and then optical energy consumption.

5.3.1 Electrical Energy Model

Electrical energy dissipated includes the energy of the link, router, and back-end circuit for optical transmitter and receiver. We use ORION 2.0 [37] to obtain the energy dissipation values for an electrical link and router, and modify their parameters for 22 nm technology according to ITRS. We assume all electrical links are optimized for delay and the injection rate is 0.1. Moreover, we include the energy dissipated in both planar and vertical links (communicating only with layer 1). The length of electrical links in Firefly and Mesh are $20\ \text{mm}/8 = 2.5\ \text{mm}$ and $20\ \text{mm}/16 = 1.25\ \text{mm}$, respectively. The energy for planar link is conservatively obtained as 0.15 pJ/bit for Firefly and 0.075 pJ/bit for Mesh under low swing voltage level [37]. It should be mentioned that the energy per bit per distance is the same in Firefly and Mesh network. A Mesh link dissipates half the energy as Firefly link because a Mesh link is half the distance of a Firefly link. For a 10-layer chip, the vertical via is determined as $\sim$100-200 μm [16], which is significantly less than planar links. As a result, the power consumption of vertical links is very small. We neglect it when we calculate our electrical link power model. For the electrical router power, we calculate the energy dissipated, per hop, in a 8×8 router to be 0.30 pJ/bit [37]. A 5×5 router with the same buffer size is 0.22 pJ/bit [37]. For the 8×8 router, the clock contributes 0.047 pJ/bit, the buffers contributes 0.07 pJ/bit, the crossbar contributes 0.178 pJ/bit, switch arbiter contributes 0.0025 pJ/bit, and the VA arbiter contributes 0.0025 pJ/bit. As for the 5×5 router, the clock contributes 0.034 pJ/bit, the buffers contribute 0.051 pJ/bit, the crossbar contributes 0.135 pJ/bit, switch arbiter contributes 0.0018 pJ/bit, and the VA arbiter contributes 0.0018 pJ/bit. For each optical transmitted bit, we need to provide electrical back-end circuit for the transmitter end and receiver end. We assume that the O/E and E/O converter energy is 100 fJ/b, as predicted in [38].

TABLE 5
Electrical and Optical Power Losses for Select Optical Components

Component	Value	Unit
Laser efficiency	5	dB
Coupler(Fiber to Waveguide)	1	dB
Waveguide	1	dB/cm
Splitter	0.2	dB
Non-Linearity	1	dB
Ring Insertion & scattering	1e-2 - 1e-4	dB
Ring Drop	1.0	dB
Waveguide Crossings	0.5	dB
Photo Detector	0.1	dB
Ring Heating(per ring)	26	μW
Ring Modulating(per ring)	500	μW
Receiver Sensitivity	-26	dBm

TABLE 6
Electrical Power Dissipation for Various NIs

	Corona	Fire-fly	OCMP	Mesh
Link(pJ/b)	-	0.15	-	75
Router(pJ/b)	0.22	0.30	0.22	0.22
O/E, E/O(fJ/b)	100	100	100	-
Optical loss(dB)	-25.2	-17.6	-16	-
Power(λ, mW)	0.81	0.14	0.10	-
Laser power(W)	13.6	2.4	6.1	-
Ring heating(W)	26	6.5	27.5	-

5.3.2 Optical Energy and Loss Model

The power dissipated in the optical link is determined by: $P_{laser} = P_{rx} + C_{loss} + M_s$ where P_{laser} is the laser power, P_{rx} is the receiver sensitivity, C_{loss} is the channel losses, and M_s is the system margin. To perform an accurate comparison with the other two optical architectures, we use the same optical device parameters and loss values provided in [3], as listed in Table 5.

Based on the energy model discussed in the previous section, we calculate the energy parameters of all four architectures as shown in Table 6. We test uniform, complement, and butterfly traffic patterns with 0.1 injection rate and obtain energy per-bit comparison as shown in Fig. 8. Fig. 8a shows the energy per bit for uniform traffic. OCMP is the most energy-efficient network followed by Corona and R-OCMP. R-OCMP saves 23.1 percent and 36.1 percent energy per bit when compared to Firefly and Mesh, respectively. Fig. 8b shows the energy per bit for complement traffic. For complement traffic, the average energy per bit increases for Mesh, Flattened-Butterfly, and Firefly architectures. This increase in power is contributed to the higher hop count of complement traffic for uniform traffic. Fig. 8c shows the average energy per bit for butterfly traffic. In butterfly traffic, the average energy per bit for Mesh and Flattened-Butterfly is lower than R-OCMP. As most traffic is near neighbor traffic, higher energy is dissipated in traversing through an optical link rather than the shorter electrical link. It should be noted that when the network injection rate increases, R-OCMP becomes much more energy efficient than other three architectures.

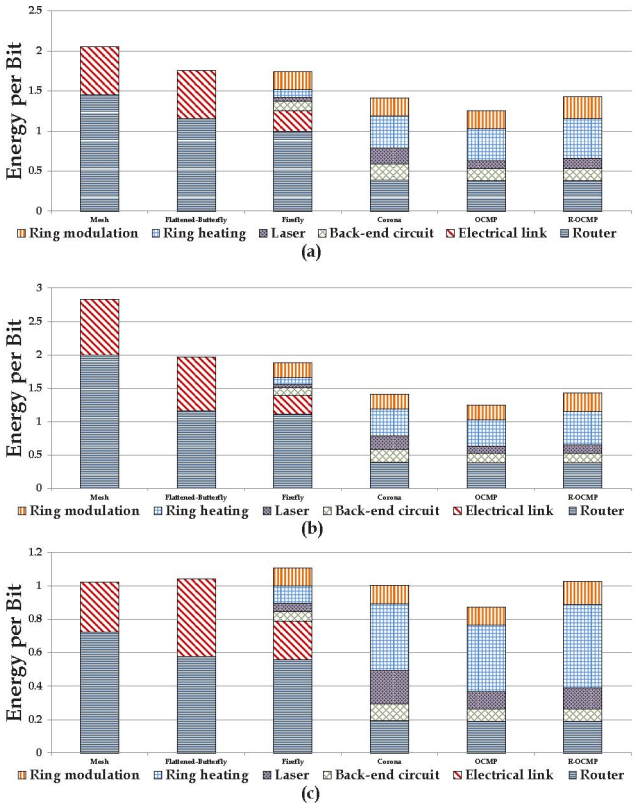

Fig. 8. Average energy per-bit for electrical and NIs: (a) uniform traffic, (b) complement traffic, and (c) butterfly traffic.

TABLE 7
Electrical and Optical Area Overhead for Select Electrical and Optical Components

Component	Area
Electrical Link	0.0085 (mm^2)
Router (8 $\times$ 8)	0.128 (mm^2)
Photodetector receiver circuitry	0.02625 (mm^2)
Microring resonator	100(μm^2)
Photodetector	100(μm^2)
Waveguide	5.5 μm

5.4 Area Analysis

In this section, we analytically compare the optical and electrical area overhead of OCMP to Firefly [8] and Corona [2] NIs. For the optical area overhead, we considered the area required for all waveguides, MRRs, and photodetectors. For the electrical layer, we considered the area required for all routers, electrical links, and electrical receiver circuitry. Table 7 shows the area overhead of both optical and electrical components used in the area calculation. In Table 7, each router and electrical link values were obtained from Orion 2.0 by directly scaling 32 nm technology values to 22 nm technology.

From our evaluation, we observe that both Corona and Firefly require 10 percent more optical area than OCMP.

This may be counter-intuitive, but OCMP uses decomposed crossbars that permit waveguides in OCMP to be shorter than the long serpentine waveguides used in both Corona and Firefly. In terms of electrical layer area overhead, OCMP consumes 4× more electrical area than Corona. As each tile is connected to four optical layers to facilitate intergroup communication, each tile in turn should have the ability to receive four signals instead of one as in Corona. However, when OCMP is compared to Firefly in terms of electrical area overhead, Firefly consumes about 75 percent more area. The proposed decomposed crossbars allow each tile to receive data from four other tiles instead of just one, thereby increasing the communication bandwidth to each tile while reducing the optical area overhead.

6 Conclusions

In this paper, we propose a 3D-stacked NI called OCMP. OCMP uses emerging NIs and 3D integration to reduce the optical power losses found in 2D planar NoCs by decomposing a large 2D nanophotonic crossbar into multiple smaller nanophotonic crossbar layers. In addition, we proposed a reconfiguration algorithm that maximizes the available bandwidth through runtime monitoring of network resources and dynamically reallocating channel bandwidth. Our simulation results indicate that the proposed OCMP architecture with reconfiguration reduces the execution time up to 25 percent for Splash2, PARSEC, and SPEC CPU2006 benchmarks when compared to electrical networks (Mesh and Flattened-Butterfly) and photonic networks (Corona and Firefly). Moreover, 256-core version of OCMP provides an energy savings of 23 percent when compared to state-of-the-art electrical and photonic networks. The proposed reconfigurable OCMP architecture that combines 3D-stacking with NI has several advantages that can translate into reduced execution time and energy savings for future many-core and multicore architectures.

Acknowlegements

This work was partially supported in part by the US National Science Foundation grants ECCS-0725765, CCF-0953398, CCF-0915418, CCF-1054339 (CAREER), and ECCS-1129010.

References

[1] ITRS, http://www.itrs.org, 2013.
[2] D. Vantrease, R. Schreiber, M. Monchiero, M. McLaren, N. Jouppi, M. Fiorentino, A. Davis, N. Binker, R. Beausoleil, and J.H. Ahn, "Corona: System Implications of Emerging Nanophotonic Technology," *Proc. 35th Int'l Symp. Computer Architecture,* pp. 153-164, June 2008.
[3] C. Batten, A. Joshi, J. Orcutt, A. Khilo, B. Moss, C. Holzwarth, M. Popovic, H. Li, H. Smith, J. Hoyt, F. Kartner, R. Ram, V. Stojanovi, and K. Asanovic, "Building Manycore Processor-to-Dram Networks with Monolithic Silicon Photonics," *Proc. 16th Ann. Symp. High-Performance Interconnects,* Aug. 2008.
[4] J.D. Owens, W.J. Dally, R. Ho, D.N. Jayasimha, S.W. Keckler, and L.S. Peh, "Research Challenges for On-Chip Interconnection Networks," *IEEE Micro,* vol. 27, no. 5, pp. 96-108, Sept./Oct. 2007.
[5] R.G. Beausoleil, P.J. Kuekes, G.S. Snider, S.-Y. Wang, and R.S. Williams, "Nanoelectronic and Nanophotonic Interconnect," *Proc. IEEE,* vol. 96, no. 2, pp. 230-247, Feb. 2008.
[6] A. Shacham, K. Bergman, and L.P. Carloni, "Photonic Networks-on-Chip for Future Generations of Chip Multiprocessors," *IEEE Trans. Computers,* vol. 57, no. 9, pp. 1246-1260, Sept. 2008.
[7] R.K. Dokania and A.B. Apsel, "Analysis of Challenges for On-Chip Optical Interconnects," *Proc. 19th ACM Great Lakes Symp. VLSI (GLSVLSI '09),* pp. 275-280, 2009.
[8] Y. Pan, P. Kumar, J. Kim, G. Memik, Y. Zhang, and A. Choudhary, "Firefly: Illuminating Future Network-on-Chip with Nanophotonics," *Proc. 36th Ann. Int'l Symp. Computer Architecture,* 2009.
[9] D. Park, E. Soumya, R. Das, M.A.K.Y. Xie, N. Vijaykrishnan, and C.R. Das, "Mira: A Multi-Layered On-Chip Interconnect Router Architecture," *Proc. 35th Int'l Symp. Computer Architecture (ISCA '08),* pp. 251-261, 2008.
[10] G.H. Loh, "3D-Stacked Memory Architectures for Multi-Core Processors," *Proc. 35th Int'l Symp. Computer Architecture (ISCA '08),* pp. 453-464, 2008.
[11] S. Souri, K. Banerjee, A. Mehrotra, and K. Saraswat, "Multiple Si Layer ICs: Motivation, Performance Analysis, and Design Implications," *Proc. 37th Design Automation Conf.,* pp. 213-220, 2000.
[12] J. Kim, C. Nicopoulos, D. Park, R. Das, Y. Xie, V. Narayanan, M.S. Yousif, and C.R. Das, "A Novel Dimensionally-Decomposed Router for On-Chip Communication in 3D Architectures," *Proc. 34th Ann. Int'l Symp. Computer Architecture (ISCA),* vol. 35, no. 2, pp. 138-149, 2007.
[13] S. Woo, M. Ohara, E. Torrie, J. Singh, and A. Gupta, "The Splash-2 Programs: Characterization and Methodological Considerations," *Proc. 22nd Ann. Int'l Symp. Computer Architecture,* pp. 24-36, 1995.
[14] C. Bienia, S. Kumar, J.P. Singh, and K. Li, "The PARSEC Benchmark Suite: Characterization and Architectural Implications," *Proc. 17th Int'l Conf. Parallel Architectures and Compilation Techniques,* Oct. 2008.
[15] M. Martin, D. Sorin, B. Beckmann, M. Marty, M. Xu, A. Alameldeen, K. Moore, M. Hill, and D. Wood, "Multifacet's General Execution-Driven Multiprocessor Simulator (GEMS) Toolset," *ACM SIGARCH Computer Architecture News,* vol. 33, no. 4, pp. 92-99, Nov. 2005.
[16] P. Koka, M.O. McCracken, H. Schwetman, X. Zheng, R. Ho, and A.V. Krishnamoorthy, "Silicon-Photonic Network Architectures for Scalable, Power-Efficient Multi-Chip Systems," *Proc. 37th Ann. Int'l Symp. Computer Architecture (ISCA),* June 2010.
[17] D. Vantrease, N. Binkert, R. Schreiber, and M.H. Lipasti, "Light Speed Arbitration and Flow Control for Nanophotonic Interconnects," *Proc. 42nd Ann. IEEE/ACM Int'l Symp. Microarchitecture (MICRO 42),* pp. 304-315, 2009.
[18] Y. Pan, J. Kim, and G. Memik, "FlexiShare: Channel Sharing for an Energy-Efficient Nanophotonic Crossbar," *Proc. 16th Int'l Symp. High Performance Computer Architecture (HPCA),* pp. 1-12, 2010.
[19] N. Kirman and J.F. Martinez, "A Power-Efficient All-Optical On-Chip Interconnect Using Wavelength," *Proc. 15th Edition of ASPLOS on Architectural Support for Programming Languages and Operating Systems (ASPLOS 10),* Mar. 2010.
[20] X. Zhang and A. Louri, "A Multilayer Nanophotonic Interconnection Network for On-Chip Many-Core Communications," *Proc. ACM/IEEE 47th Design and Automation Conf. (DAC),* June 2010.
[21] N. Sherwood-Droz, K. Preston, J.S. Levy, and M. Lipson, "Device Guidelines for WDM Interconnects Using Silicon Microring Resonators," *Proc. Workshop Interaction between Nanophotonic Devices and Systems (WINDS),* pp. 15-18, Dec. 2010.
[22] M. Petracca, B.G. Lee, K. Bergman, and L.P. Carloni, "Photonic NoCs: System-Level Design Exploration," *IEEE Micro 09,* vol. 29, no. 4, pp. 74-85, July/Aug. 2009.
[23] D.A.B. Miller, "Device Requirements for Optical Interconnects to Silicon Chips," *Proc. IEEE,* vol. 97, no. 7, pp. 1166-1185, July 2009.
[24] X. Zheng, F. Liu, J. Lexau, D. Patil, G. Li, Y. Luo, H. Thacker, I. Shubin, J. Yao, K. Raj, R. Ho, J. Cunningham, and A. Krishnamoorthy, "Ultra-Low Power Arrayed CMOS Silicon Photonic Transceivers for an 80 Gbps WDM Optical Link," *Proc. Optical Fiber Comm. Conf.,* Mar. 2011.
[25] S.J. Koester, C.L. Schow, L. Schares, and G. Dehlinger, "National Fiber Optic Engineers Conf.," *J. Lightwave Technology,* vol. 25, no. 1, pp. 46-57, Jan. 2007.
[26] P. Koonath and B. Jalali, "Multilayer 3D Photonics in Silicon," *Optics Express,* vol. 15, pp. 12686-12691, 2007.
[27] K. Preston, S. Manipatruni, A. Gondarenko, C.B. Poitras, and M. Lipson, "Deposited Silicon High-Speed Integrated Electro-Optic Modulator," *Optics Express,* vol. 17, pp. 5118-5124, 2009.

[28] A. Biberman, K. Preston, G. Hendry, N. Sherwood-Droz, J. Chan, J.S. Levy, M. Lipson, and K. Bergman, "Photonic Network-On-Chip Architectures Using Multilayer Deposited Silicon Materials for High-Performance Chip Multiprocessors," *J. Emerging Technology Computing Systems,* vol. 7, pp. 1-25, July 2011.
[29] W.J. Dally and B. Towles, *Principles and Practices of Interconnection Networks.* Morgan Kaufmann, 2004.
[30] B.E. Little and S.T. Chu, "Microring Resonators for Very Large Scale Integrated Photonics," *Proc. IEEE Ann. Meeting Lasers and Electro-Optics Soc. Conf. (LEOS),* pp. 487-8, 1999.
[31] K. Chen, K.S. Chiang, H.P. Chan, and P.L. Chu, "Growth of C-Axis Orientation ZnO Films on Polymer Substrates by Radio-Frequency Magnetron Sputtering," *Optical Materials,* vol. 30, pp. 1244-50, Apr. 2008.
[32] X.Y. Zhang, A. Dhawan, P. Wellenius, A. Suresh, and J.F. Muth, "Planar ZnO Ultraviolet Modulator," *Applied Physics Letters,* vol. 91, pp. 071107-071110, 2007.
[33] X. Chen, L.-S. Peh, G.-Y. Wei, Y.-K. Huang, and P. Pruncal, "Exploring the Design Space of Power-Aware Opto-Electronic Networked Systems," *Proc. 11th Int'l Symp. High-Performance Computer Architecture (HPCA '05),* pp. 120-131, Feb. 2005.
[34] V. Soteriou, N. Eisley, and L.-S. Peh, "Software-Directed Power-Aware Interconnection Networks," *ACM Trans. Architecture and Code Optimization,* vol. 4, Mar. 2007.
[35] A. Kumar, P. Kundu, A.P. Singh, L.-S. Peh, and N.K. Jha, "A 4.6 Tbits/s 3.6 GHz Single-Cycle NoC Router with a Novel Switch Allocator in 65 nm CMOS," *Proc. 25th Int'l Conf. Computer Design (ICCD),* Oct. 2007.
[36] J. Kim, W.J. Dally, and D. Abts, "Flattened Butterfly: Cost-Efficient Topology for High-Radix Networks," *Proc. 34th Ann. Int'l Symp. Computer Architecture (ISCA),* pp. 126-137, June 2007.
[37] A.B. Kahng, B. Li, L.-S. Peh, and K. Samadi, "ORION 2.0: A Fast and Accurate NoC Power and Area Model for Early-Stage Design Space Exploration," *Proc. Design, Automation and Test in Europe Conf. and Exhibition,* pp. 423-428, Apr. 2009.
[38] A.V. Krishnamoorthy, R. Ho, X. Zheng, H. Schwetman, J. Lexau, P. Koka, G. Li, I. Shubin, and J.E. Cunningham, "Computer Systems Based on Silicon Photonic Interconnects," *Proc. IEEE,* vol. 97, no. 7, pp. 1337-1361, June 2009.

Randy W. Morris Jr., received the BS, MS, and PhD degrees in electrical engineering and computer science from Ohio University, Athens in 2007, 2009, and 2012, respectively. He is currently working for Intel as a validation engineer for the Many Integrated Cores (M.I.C.) group in Portland, Oregon. He is a student member of the IEEE.

Avinash Karanth Kodi received the MS and PhD degrees in electrical and computer engineering from the University of Arizona, Tucson, in 2006 and 2003, respectively. He is currently an assistant professor of electrical engineering and computer science at Ohio University, Athens. His research interests include computer architecture, optical interconnects, chip multiprocessors (CMPs), and network-on-chips (NoCs). He is the recipient of the US National Science Foundation (NSF) CAREER award in 2011. He is a senior member of the IEEE.

Ahmed Louri received the PhD degree in computer engineering in 1988 from the University of Southern California (USC), Los Angeles. He is currently a full professor of electrical and computer engineering at the University of Arizona, Tucson, and the director of the High Performance Computing Architectures and Technologies (HPCAT) Laboratory (www.ece.arizona.edu/~ocppl). His research interests include computer architecture, network-on-chips (NoCs), parallel processing, power-aware parallel architectures, and optical interconnection networks. He has served as the general chair of the 2007 IEEE International Symposium on High-Performance Computer Architecture (HPCA), Phoenix, Arizona. He has also served as a member of the technical program committee of several conferences including the ANCS, HPCA, MICRO, NoCs, among others. He is a fellow of the IEEE and a member of the OSA.

Ralph D. Whaley Jr., (M'98) received the BE with a double major in electrical engineering and physics from Vanderbilt University in Nashville, Tennessee, in 1989, the ScM degree in physics from Brown University in Providence, Rhode Island, in 1991, and the PhD in electrical engineering from the University of Maryland, College Park, in 2001. In 2000, he joined the Integrated Optoelectronics group at Sarnoff Corporation in Princeton, New Jersey as a member of technical staff where he worked on InP-based active and passive photonic structures. In 2005, he joined the faculty in the School of Electrical Engineering and Computer Science at Ohio University, Athens, as an assistant professor and is currently a member of the Nanoscale and Quantum Phenomena Institute (NQPI) and the Center for Electrochemical Engineering Research (CEER). He is a member of the OSA, SPIE, APS, and MRS.