WRH-ONoC: A Wavelength-Reused Hierarchical Architecture for Optical Network on Chips

Feiyang Liu, Haibo Zhang, Yawen Chen, Zhiyi Huang
Department of Computer Science
University of Otago, Dunedin, New Zealand
Email:{feiyang,haibo,yawen,hzy}@cs.otago.ac.nz

Huaxi Gu
State Key Laboratory of ISN
Xidian University, Xi'an, China
Email:hxgu@xidian.edu.cn

***Abstract*—Optical Network on Chip (ONoC) is a promising technology for the next-generation many-core chip multiprocessors owing to its tremendous advantages in low power consumption, low communication delay, and high bandwidth. In this paper we present WRH-ONoC, a novel wavelength-reused hierarchical architecture that is capable of interconnecting thousands of cores using a limited number of wavelengths while providing extremely high-throughput data communication between connected cores. In WRH-ONoC, the cores are divided into small subsystems that are interconnected using multiple λ-routers and gateways in a hierarchical manner. Each λ-router can provide non-blocking parallel communication among the directly connected cores or gateways, and all λ-routers can reuse the limited number of available wavelengths. Communications between cores in different subsystems are routed via gateways in which optical signals can change their wavelengths via optical-electrical signal conversions. For a given number of cores, we give the minimum number of levels, λ-routers, and gateways required to interconnect these cores, and derive the expected end-to-end data communication delay under the Uniform-Poisson traffic pattern. Both theoretical analysis and simulation results demonstrate that WRH-ONoC can achieve significant improvement on performance and reduction on hardware cost in comparison with the existing solutions.**

***Index Terms*—On-Chip Communication, ONoC, λ-Router.**

I. Introduction

Today's processor development has moved to many cores on a single chip, e.g., 80 cores in Intel Teraflops chip and 192 cores in CSX700 Processor. It was predicted that thousands or even more cores will be integrated on a single chip in the next decade [1]. With the rapid development of high-speed cores, the conventional electrical approaches for interconnection are no longer able to meet the requirements of high throughput and low power consumption for inter-core communications. Recent advances in CMOS-compatible on-chip photonics have led to the development of Optical Network on Chip (ONoC), a silicon-based optical interconnection among cores at the chip level, as an attractive solution to overcome the limitations of conventional electrical interconnects [2]. Recently, Intel has announced the use of silicon photonic architectures to define the next generation multi-core processors and servers [3], and demonstrated its first inexpensive optical chip in 2013. With the help of Wavelength Division Multiplexing (WDM), multiple independent signals can be simultaneously transmitted through a single optical waveguide by using different wavelengths, thereby enabling extremely low communication delay and high throughput. However, optical interconnection also suffers from some limitations, such as the lack of optical buffer and processing logics, limited chip area, limited number of wavelengths (a maximum of 62 wavelengths for an MR-based WDM interconnect assuming a 10 *Gbps* data rate with -20 *dB* noise [4]), etc. All these limitations pose great challenges to design the efficient and scalable ONoC architectures.

Current existing ONoC architectures can be classified into two categories: (i) *architectures for small-scale ONoCs using bus, ring, crossbar, and multi-stage switching*. For instance, λ-router uses 2x2 optical switching elements to construct a non-blocking all-optical interconnection [5]. Kao *et al.* proposed a photonic *clos* network for the fully interconnection of 64 cores [6]. However, these architectures cannot be easily scaled to a large system due to either severe resource contention or high hardware cost. (ii) *architectures that combine electrical and optical interconnections for large-scale ONoCs*. Generally, these optical-electrical hybrid NoCs are composed of two layers: an optical layer for communication and an electrical layer for buffering and processing. For example, in [7] Shacham *et al.* proposed a hybrid ONoC that uses an optical circuit-switched network to transmit large data messages with guaranteed bandwidth, and an electrical packet-switched network to transmit small messages and control packets without occupying optical circuits. A path setup process is required before each data transmission. Kurian *et al.* proposed ATAC [8] that uses an electrical mesh network for intra-cluster communication and an optical snake bus for inter-cluster communication. Another hybrid architecture, Corona [9], uses a photonic crossbar to fully interconnect 64 clusters with each cluster having 4 cores connected by an electrical crossbar. These architectures have better scalability, but still suffer from severe resource contention when the system size increases.

The objective of this work is to design a scalable architecture that is capable of interconnecting thousands of cores using limited resources while still being able to provide desirable performance. The key contributions are listed as follows:

- We propose a novel wavelength-reused hierarchical architecture for ONoCs that sustains the strength of λ-router in high-speed communication but offsets its weakness in scalability. By dividing cores into subsystems and connecting them using multiple λ-routers in a hierarchical manner, the limited number of wavelengths can be reused.

978-1-4799-8381-0/15/$31.00 ©2015 IEEE

- We analyse the minimum hardware requirement for connecting a given number of cores. Theoretical results demonstrate that hardware requirement can be reduced by $\sim 90\%$ in comparison with the single λ-router approach.
- We derive the expected end-to-end packet delay for the traffic generated with Uniform distribution in space and Poisson distribution in time.
- We carry out extensive simulations to evaluate the performance of WRH-ONoC, and results show that our scheme can achieve significant performance improvement in compassion with the electrical-optical hybrid approach.

The rest of this paper is organized as follows. Section II introduces some background on ONoCs. Section III presents WRH-ONoC, and Section IV analyses the hardware requirement and performance of WRH-ONoC. Section V presents the simulation results and the paper is concluded in Section VI.

II. Background

A. Optical NoC

The fundamental building blocks of an ONoC include laser sources, waveguides, microring resonators, optical routers, modulators, and photodetectors. Laser sources generate optical signals on which the information is modulated, and photodetectors receive optical signals and convert them back to electrical signals. Waveguides are on-chip light transmitting medium whose propagation loss can be less than 1 *dB*/cm [10]. Microring Resonator (MR) is a compact and energy-efficient optical filter that is designed to filter optical signals with a specific wavelength (a wavelength can be thought of as a color of light). As shown in Fig. 1(a), when the wavelength of the input signal λ_i equals to the resonant wavelength λ_r of the MR, the optical signal couples into the MR and changes its direction; otherwise the optical signal keeps the original direction. MR is the basic component in the design of optical routers and modulators. As illustrated in Fig. 1(b), optical signals with different wavelengths can be filtered and routed to different output ports based on the resonant wavelengths.

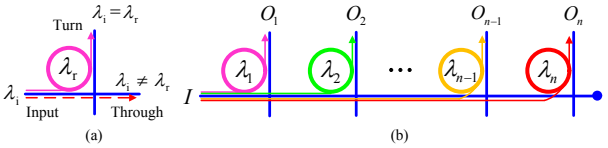

Fig. 1. (a) Microring resonator; (b) A simple optical router.

Wavelength division multiplexing (WDM) is a technology that can multiplex multiple optical carrier signals onto a single waveguide by using different wavelengths, and the multiplexed signal can be demultiplexed at the receivers using specific MR filters. For a WDM-based photonic circuit, the aggregate data rate can reach 200 *Gbps* over a 25 *Gbps* optical channel using 8 wavelengths [11].

B. λ-Router

The λ-router is a WDM-based all-optical network that can provide non-blocking communication among the connected ports [5]. Fig. 2(b) shows an 8-inputs×8-outputs λ-router. The key component in a λ-router is the 2-inputs×2-outputs optical switching element (OSE) designed based on MRs, as shown in Fig. 2(a). When the wavelength of the input signal λ_i equals to the resonant wavelength λ_r of the OSE, the optical signal will couple into the MR and output from one port; otherwise the signal will pass the OSE and output from another port.

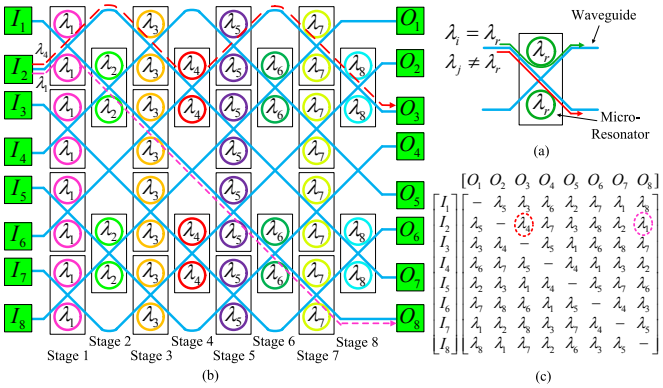

Fig. 2. λ-router: (a) 4-port optical switching element (OSE); (b) 8-input 8-output connection architecture; (c) Wavelength matrix.

To achieve non-blocking communication among the connected ports, an $N \times N$ λ-router needs N waveguides and N wavelengths. Assume that there is no communication between input port I_i and output port O_i as they generally connect to the same core. Therefore, the minimum number of OSEs required for non-blocking communication among N connected cores is $\lceil \frac{N(N-2)}{2} \rceil$. There are several different kinds of OSE layouts for non-blocking communication. As illustrated by Fig. 2(b), a straightforward approach is to place the OSEs in N stages where the number of OSEs in stage i is $\lfloor \frac{N}{2} \rfloor$ if i is odd and $\lfloor \frac{N-3}{2} \rfloor$ if i is even [12]. In each stage, all the OSEs share the same resonant wavelength. The wavelength used for communication between I_i and O_j is determined by $M_{i,j}$, where $\boldsymbol{M}$ is the wavelength matrix. For example, according to the matrix given in Fig. 2(c), $M_{2,3} = \lambda_4$ and $M_{2,8} = \lambda_1$. Thus the corresponding paths for communications from I_2 to O_3 and I_2 to O_8 use λ_4 and λ_1 respectively, as highlighted by dashed lines in Fig. 2(b). The wavelength matrix should guarantee that each input port uses different wavelengths to communicate with different output ports, and each output port receives signals from different input ports using different wavelengths. Hence, the λ-router is a non-blocking passive optical network since the wavelength matrix is predefined and the communication path can be automatically determined based on the wavelength used for communication.

The λ-router has several advantages that make it promising for ONoCs. (i) *fully non-blocking communication*: with N waveguides and N wavelengths, there are $N\times(N-1)$ physical channels that fully connect N cores (no self-to-self communication), thus maximizing the utilisation of waveguides and wavelengths. (ii) *broadcasting and multicasting*: with WDM technique, one core can broadcast/multicast packets to the connected cores by using different wavelengths. (iii) *low latency and high bandwidth*: each waveguide contains $N-1$ optical channels each using a different resonant wavelength, as shown in Fig. 2(c). Thus, the aggregate bandwidth per waveguide can achieve $10 \times (N-1)$ *Gbps* if the bandwidth for each

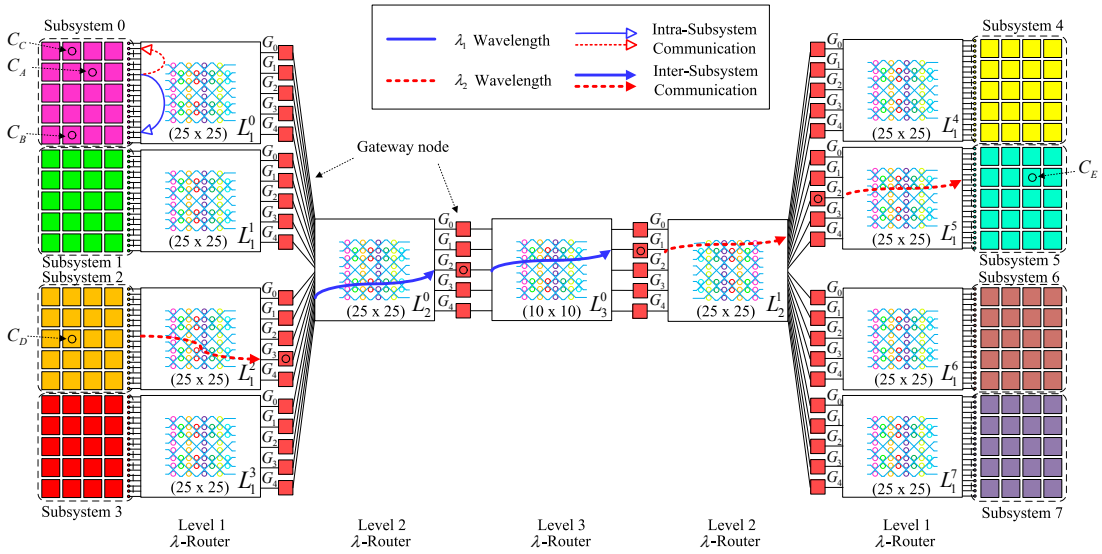

Fig. 3. An example of the wavelength-reused hierarchical ONoC architecture for connecting 160 cores using 25 wavelengths using 3 levels λ-router.

channel is 10 *Gbps* [11]. (iv) *customizability*: a λ-router can be customized based on the application's communication pattern to reduce the hardware cost and energy consumption [13].

The major drawback of the λ-router is its poor scalability. The number of required waveguides and wavelengths is linearly proportional to the router size, but the number of required OSEs increases quadratically. This makes λ-router not suitable even for medium-sized ONoCs due to the limitation on the number of available wavelengths and the difficulty in integrating so many optical components in a limited chip area.

III. Principle and Architecture

The key idea of our scheme is to combine hierarchical routing with wavelength reuse to sustain the strength of λ-router but offset its weakness. As shown in Fig. 3, all the cores are grouped into multiple subsystems. In the same subsystem, the cores are connected using one λ-router, thereby providing non-blocking direct communication within each subsystem. The communications between two cores in different subsystems are done via the gateways that are connected using multiple λ-routers and organised in a hierarchical manner. In our scheme, the available wavelengths are reused by all λ-routers, thus overcoming the drawback of λ-router in scalability.

A. Hierarchical Interconnection

We assume that an ONoC is to be designed to connect N cores using $W_{\max}$ wavelengths where $N \gg W_{\max}$. Thus, it is impossible to interconnect all N cores using one λ-router. We divide the N cores into several subsystems according to the available wavelengths, and interconnect them using multiple λ-routers and gateways in a hierarchical manner, as illustrated in Fig. 3. The gateways serve as bridges between λ-routers, where an optical signal can be converted to electric signal, temporarily buffered, and then retransmitted in optical signal using a different wavelength. The number of cores and gateways connected by one λ-router should be no larger than $W_{\max}$, and all λ-routers can reuse these $W_{\max}$ wavelengths. Fig. 3 gives an example for interconnecting 160 cores using only 25 wavelengths via a three-level hierarchical λ-router network. The 160 cores are divided into 8 subsystems with each having 20 cores. Suppose that 5 sibling gateways are used to interconnect one λ-router to a next-level λ-router. Each λ-router in level 1 is a 25×25 λ-router that interconnects all 20 cores in one subsystem and 5 gateways. Each λ-router in level 2 is also a 25×25 λ-router but interconnects 20 gateways from 4 subsystems and 5 gateways for higher level interconnection. The level-3 λ-router is a 10×10 router that interconnects with the level-2 λ-routers using 10 gateways. Another advantage of this hierarchical interconnection is the ability to achieve high fault-tolerance and load-balancing, as there are multiple redundant paths between cores in different subsystems. In the example given in Fig. 3, there are 625 paths between C_D and C_E, depending on which gateways are selected for communication between these two cores.

B. Gateway Architecture

Since all λ-routers reuse the same set of available wavelengths, communications between cores in different subsystems may use different wavelengths to pass through different λ-routers in the route, e.g., C_D to C_E in Fig. 3. To change the wavelength, optical signals need to be converted to electrical signals and then regenerated using different wavelengths for the next-hop. To reduce the overhead at gateways, we present the following design for the gateway, as illustrated in Fig. 4.

A gateway is composed of the following key components: input/output ports, Optical-to-Electrical (O/E) and Electrical-to-Optical (E/O) converters, buffer queues, packet dispatchers, and wavelength matrices. Each gateway has two pairs of input and output ports: one pair for upward traffic from a lower-level λ-router to a higher-level λ-router, and the other for downward traffic from a higher-level λ-router to a lower-level λ-router. Each pair of input/output ports has independent buffer queues. Such a design guarantees that upward and downward traffics can be processed in parallel. Let w_l and w_h be the number

of wavelengths used by the lower-level and higher-level λ-routers connected by the gateway, respectively. As shown in Fig. 4, there are w_l O/E converters for port *Input1* with each connected to an MR and an input queue. The optical signal filtered by the MR with a resonant wavelength λ_i is converted to electrical signal by the associated O/E converter and buffered in the input queue for λ_i. Similarly, there are w_h E/O converters for port *Output1* with each connected to an output buffer and an MR. The packet in the output buffer allocated for λ_j is modulated by the MR with wavelength λ_j to the optical signal and transmitted. The input queues and output buffers are fully connected using an internal crossbar. Each input queue is associated with a packet dispatcher that is responsible for dispatching packets from the input queue to the corresponding output buffers based on the wavelength to be used for the next hop. The design for connecting another pair of input and output ports is the same.

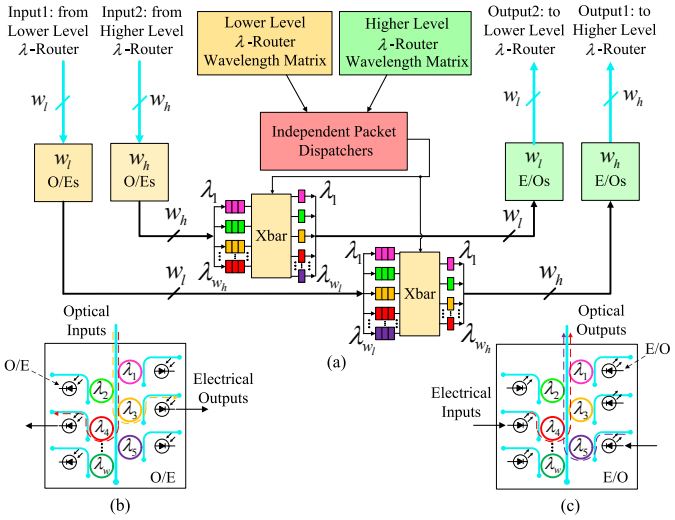

Fig. 4. (a) Architecture of gateway; (b) O/E converters; (c) E/O converters.

The gateway works as follows: when a set of WDM optical signals enter an input port, each uni-wavelength signal is filtered by a specific MR, converted into electrical signal, and written to the corresponding input queue based on the receiving wavelength, as illustrated in Fig. 4(b). The packet dispatchers continually dispatch packets from the input queues to the output buffers. Based on the destination of a packet, the dispatcher determines the next-hop λ-router for each packet (details on how to choose the next-hop will be given in Section III-C), and looks up either the lower-level or higher-level wavelength matrix to determine the wavelength to be used for next-hop. As illustrated in Fig. 4(c), the gateway continuously injects all packets in the output buffers to the E/O converters in parallel, where multiple optical signals are multiplexed and transmitted over one waveguide using different wavelengths.

The advantage of our design is that multiple optical signals using different wavelengths can be processed concurrently in the gateway, and there is no blocking on each optical communication channel, thereby maximizing the channel utilization. Although the signal conversion and the packet buffering will introduce some delay, they will not significantly degrade the system performance because: (i) E/O and O/E conversions can be done at very high speed (10*Gbps*/wavelength) [14]; (ii) a large portion of the communications in an ONoC occurs locally due to task mapping [15]. If a task is assigned to cores only in one subsystem or in neighbouring subsystems, communications, such as cache coherence messages, will not experience frequent E/O and O/E conversions and buffering.

C. Communication Scheme

The foundation for packet routing in our design is the positional prefix address. Each core in the system has a unique address in the form of $\{$*networkID; coreID*$\}$. The *coreID* represents the unique identification of a core in a subsystem, and it has $\lceil \log_2 n \rceil$ bits where n is the maximum number of cores in a subsystem. The *networkID* is composed of several fields for subnetworks $\{s_l, ..., s_2, s_1\}$ where l is the number of levels of λ-routers in the hierarchy. Fig. 5 illustrates the address assignment for the network given in Fig. 3. The number of bits for *coreID* field is 5 as there are 20 cores in each subsystem. The routing structure is a tree rooted at the top-level λ-router, and all cores and gateways in a subtree rooted at a λ-router forms a subnetwork. Let $|s_i|$ be the number of bits in the s_i field where $0 < i \leq l$. $|s_l|$ is 1 as there must be one λ-router in the top level. $|s_i|$ is $\lceil \log_2 l_i \rceil$ for $i < l$ where l_i is the maximum number of level-i λ-routers connected to a λ-router in level $i+1$. This guarantees that different level-i λ-routers that connect to the same level-$(i+1)$ λ-router can be represented using different codes, as illustrated in Fig. 5.

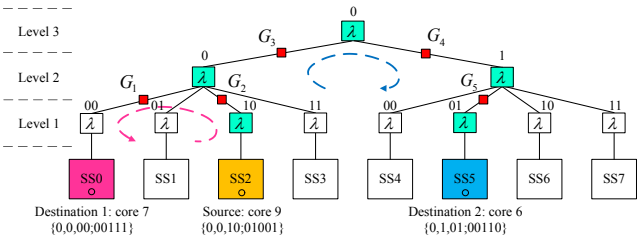

Fig. 5. Addressing approach.

The communications can be classified into two categories:

1) Intra-subsystem Communication: The *networkID* fields of the source and destination addresses must be the same. The source core can look up the local wavelength matrix to determine the wavelength for this communication, and send the packet to the destination via the connected level-1 λ-router.

2) Inter-subsystem Communication: The *networkID* fields of the source and destination addresses must not be the same, and the packet needs to be routed at gateways and delivered to the destination in a multi-hop manner. The key task for routing at gateways is to determine the next-hop as well as the wavelength to be used for the next-hop communication. We design separate rules for routing the upward and downward packets as they are processed independently at the gateways.

For an upward packet, it can be forwarded either further upward to a gateway connected to the higher-level λ-router or to a gateway connected to another λ-router at the same level, depending on the destination of the packet. For example, in Fig. 5 gateway G_1 can route a packet to either G_3 or G_2.

Consider a packet at a gateway that connects to a level-i λ-router and a level-$(i+1)$ λ-router. If the $\{s_l, ..., s_{i+1}\}$ fields for the source and destination addresses are not the same, the destination must be located outside of the subnetwork rooted at the level-$(i+1)$ λ-router, and the packet will be routed further upward to a gateway that connects the level-$(i+1)$ λ-router with a router in level $i+2$; otherwise the destination must be located in the subnetwork rooted at the level-$(i+1)$ λ-router, and the packet needs to be routed to a sibling gateway that connects to a level-i λ-router encoded with $\{s_i\}$. In the example given in Fig. 5, suppose that a packet generated by core 9 in $SS2$ is to be routed at gateway G_2. If the destination of the packet is core 6 in SS_5, the $\{s_3, s_2\}$ fields of the source and destination addresses are different, the packet is routed upward to a gateway G_3. If the destination of the packet is core 7 in $SS0$, the $\{s_3, s_2\}$ fields of the source and destination addresses are the same, and the packet is routed to a sibling gateway G_1 that connects to a λ-router in level 1 encoded as $\{s_1\} = \{00\}$. In our design, two λ-routers in adjacent levels are connected using multiple redundant gateways. To achieve load balancing, the specific gateway for next-hop is randomly chosen with an uniform distribution from the group of gateways that connect the next two λ-routers in the routing path. Once the next-hop gateway is determined, the current gateway can check the higher-level wavelength matrix to get the wavelength for next-hop communication, and send the packet to the next gateway with chosen wavelength.

For a downward packet, the next-hop λ-router that the packet will traverse must be unique. The rules for forwarding a downward packet are: (1). If the gateway is directly connected to a subsystem, it can send the packet directly to the destination via the level-1 λ-router; (2). Suppose the gateway is connected to a level-i λ-router and a level-$(i-1)$ λ-router where $i>2$, the next-hop is randomly chosen with an uniform distribution from the group of gateways that connect to the level-$(i-1)$ λ-router with the λ-router in level $(i-2)$ according to $\{s_{i-2}\}$ in the destination address for load-balance. For example, if G_4 needs to route a packet to a core in $SS5$, it will randomly choose a gateway following the uniform distribution from the group of gateways that connect to the right λ-router in level 2 and the λ-router in level 1 with $\{s_1\}=\{01\}$.

IV. Theoretical Modelling and Analysis

In this section, we first model the hardware requirement for our architecture, and then analyse the average communication delay for the traffic generated with Uniform distribution in space and Poisson distribution in time. In our analysis we assume that all subsystems have the same number of cores.

A. Hardware Requirement

The key components in our architecture for an ONoC include core interfaces, λ-routers, and gateways. We analyse the hardware requirements for these key components separately.

1) Core Interface: Each core has a network interface which consists of one transmitter and one receiver. Since each core can directly communicate with up to $W_{\max}-1$ cores/gateways in the same subsystem, the transmitter must have $W_{\max}-1$ E/O converters with each having a narrow-band on-chip laser and an MR filter to generate and modulate optical signal with a specific wavelength. Similarly, the receiver needs $W_{\max}-1$ O/E converters with each having an MR filter and a photodetector to receive an optical signal with a specific wavelength.

2) λ-Router: Assume that each λ-router is connected to a next-level λ-router using g gateways, and a hierarchy with L levels is required to interconnect all the cores and gateways. The following theorem gives the minimum number of λ-routers required at each level in the hierarchy.

Theorem 1. *The minimum number of λ-routers required at level i, denoted by R_i, is*

$$R_i = \begin{cases} \lceil \frac{N}{W_{\max}-g} \rceil, & i = 1; \\ \lceil \frac{gR_{i-1}}{W_{\max}-g} \rceil, & i \in [2, L-1]; \\ 1, & i = L. \end{cases} \quad (1)$$

Proof: Since each λ-router can directly connect to $W_{\max}$ cores and gateways using $W_{\max}$ wavelengths, the minimum number of λ-routers required at level 1 is $\lceil \frac{N}{W_{\max}-g} \rceil$. For all λ-routers in level i, they need to connect R_{i-1} λ-routers in level $i-1$ with gR_{i-1} gateways. If $gR_{i-1} \leq W_{\max}$, only one λ-router is required at level i to interconnect the gR_{i-1} gateways, and level i is the top level; otherwise more than one λ-routers are required at level i, and each λ-router in level i also needs to connect one λ-router in level $(i+1)$ via g gateways. Thus, $g(R_{i-1}+R_i) \leq R_i W_{\max}$, and we have $R_i \geq \frac{gR_{i-1}}{W_{\max}-g}$. Hence, the minimum number of λ-routers at level i is $\lceil \frac{gR_{i-1}}{W_{\max}-g} \rceil$. □

Let R_{sum} and G_{sum} be the minimum number of λ-routers and gateways required to connect N cores using $W_{\max}$ wavelengths, respectively. Then L, R_{sum} and G_{sum} can be computed using Algorithm 1.

Algorithm 1: HARDWARE COST(L,R_{sum}, G_{sum})

Input: N, $W_{\max}$, g;
Output: L, R_{sum}, G_{sum};
1 $R_1 \leftarrow \lceil \frac{N}{W_{\max}-g} \rceil$; $i \leftarrow 2$;
2 $R_{sum} \leftarrow R_1$;
3 **while** $gR_{i-1} > W_{\max}$ **do**
4 $R_i \leftarrow \lceil \frac{gR_{i-1}}{W_{\max}-g} \rceil$
5 $R_{sum} \leftarrow R_{sum} + R_i$;
6 $i \leftarrow i+1$;
7 $L \leftarrow i$; $R_L \leftarrow 1$;
8 $R_{sum} \leftarrow R_{sum} + R_L$;
9 $G_{sum} \leftarrow g(R_{sum} - 1)$;
10 **return** L, R_{sum}, G_{sum};

Since each λ-router (except the one at the top level) is connected to a higher-level λ-router via g gateways, thus $G_{sum} = g(R_{sum} - 1)$ (line 9 in Algorithm 1). Since each $m \times m$ λ-router needs at least $\lceil \frac{m(m-2)}{2} \rceil$ OSEs and each OSE has two MRs, each $m \times m$ λ-router needs $m(m-2)$ MRs. In our architecture there is no communication between the set of g gateways that connect the same two λ-routers. Thus, $g(g-1)$ MRs can be removed from each of the

two connected λ-routers. Let N_{rm}^i represents the number of MRs used by a level-i λ-router. If $i = 1$, each level-1 λ-router is connected to a level-2 λ-router via g gateways. Thus $N_{rm}^1 \leq W_{\max}(W_{\max}-2)-g(g-1)$. If $i > 1$, each level-i λ-router can connect to at most $\lfloor \frac{W_{\max}}{g} \rfloor$ λ-routers in other levels. Hence, $N_{rm}^i \leq W_{\max}(W_{\max}-2)-\lfloor \frac{W_{\max}}{g} \rfloor g(g-1)$. The actual number of required MRs depends on the detailed interconnections, and can be computed in the same way as Algorithm 1. Due to space limitation, we do not include the detailed algorithm to compute the actual number of MRs.

3) Gateway: Each gateway has two separate data paths for routing upward and downward traffics. In each direction, a gateway can receive optical signals from one previous-hop λ-router with up to $W_{\max}-g$ wavelengths, and send optical signals to one next-hop λ-router with up to $W_{\max}-g$ wavelengths. Thus, each data path should have up to $W_{\max}-g$ pairs of E/O and O/E converters, $W_{\max}-g$ input queues, $W_{\max}-g$ output buffers. Each path should also have up to $W_{\max}-g$ parallel wavelength dispatchers, and a $W_{\max}\times W_{\max}$ crossbar that fully connecting input queues and output buffers.

4) Hardware Cost Comparison: Table I compares the requirements of optical devices for interconnecting the same number of cores using WRH-ONoC and a single λ-router, denoted as 1-λ-router. It can be seen that our scheme can achieve a significant reduction ($\sim 90\%$) in comparison with the 1-λ-router. For instance, a 1-λ-router with $N = 320$ cores requires $N(N-1) = 102080$ pairs of E/O and O/E converters since each core is able to communicate with the other $N-1$ cores simultaneously. As illustrated in Fig. 4, each E/O or O/E converter needs one MR to modulate or filter the signal. Thus, $2N(N-1) = 204160$ MRs are required in core interfaces. Together with the $N(N-2) = 101760$ MRs used in λ-router, the total MRs needed for a 1-λ-router to connect 320 cores is 305920. In our scheme, each core only needs to directly communicate with the cores and gateways connected by the same level-1 λ-router. Thus, only $N(W_{\max}-1) = 6080$ pairs of E/O and O/E converters and $2N(W_{\max}-1) = 12160$ MRs are required in core interfaces. According to Algorithm 1, 26 λ-routers and 100 gateways are required to connect the 320 cores. Hence, another 8670 MRs are required for the 26 λ-routers, and 3200 pairs (1600 pairs for each direction) of E/O and O/E converters and 6400 MRs are required for the 100 gateways. In total, WRH-ONoC requires 9280 pairs of E/O and O/E converters and 27320 MRs, which is only $\sim 9\%$ of the hardware requirement for the single λ-router approach.

TABLE I. OPTICAL HARDWARE REQUIREMENTS COMPARISON

Architecture	Configuration			Hardware Requirements			
	N	$W_{\max}$	g	E/O&O/E	Reduction	MR	Reduction
1-λ-router	320	320	-	102080	-	305920	-
WRH-ONoC	320	20	4	9280	90.91%	27320	91.07%
1-λ-router	400	400	-	159600	-	478400	-
WRH-ONoC	400	25	5	14600	90.85%	43150	90.98%
1-λ-router	480	480	-	229920	-	689280	-
WRH-ONoC	480	30	6	21120	90.81%	62580	90.92%
1-λ-router	640	640	-	408960	-	1226240	-
WRH-ONoC	640	40	8	37760	90.77%	112240	90.85%

B. Communication Delay

In our analysis, packet communication delay is defined as the amount of time taken for a packet to be transmitted across the hierarchical λ-router network from the source core to the destination core. Depending on the destinations, different packets may traverse different number of hops. We use $\overline{D}$ to denote the average packet delay, which is modelled as follows:

$$\overline{D} = \alpha \overline{D}_{intra} + (1-\alpha)\overline{D}_{inter}, \tag{2}$$

where $\overline{D}_{intra}$ and $\overline{D}_{inter}$ represent the average delivery delay of intra-subsystem traffics and inter-subsystem traffics, respectively, and α is the proportion of the intra-subsystem traffics.

For intra-subsystem traffics, each packet traverses only one specific λ-router in level 1. Due to the high speed of optical communication, we assume the delay for a packet to pass each λ-router is the same. Hence, the delay for each intra-subsystem communication is constant and can be computed as follows:

$$\overline{D}_{intra} = D_{E/O} + D_{\lambda R} + D_{O/E}, \tag{3}$$

where $D_{E/O}$ and $D_{O/E}$ are the delay incurred by the O/E and E/O conversions, respectively, and $D_{\lambda R}$ is the packet propagation delay over one λ-router.

For inter-subsystem communications, $\overline{D}_{inter}$ can be computed as follows:

$$\overline{D}_{inter} = D_{E/O} + \underbrace{\overline{N}_{hop} \times D_{\lambda R}}_{I} + \underbrace{(\overline{N}_{hop}-1)\times D_{GW}}_{II} + \overline{D}_Q + D_{O/E}, \tag{4}$$

where $\overline{N}_{hop}$ is the expected number of hops that a packet traverses. Part I is the expected accumulated delay for the packet to traverse the en-route λ-routers. Part II is the accumulated delay incurred by packet processing at gateways excluding the packet queuing delay, where D_{GW} is the packet processing delay at one gateway and can be modelled as follows:

$$D_{GW} = D_{E/O} + 2D_{buf} + D_{xbar} + D_{wl} + D_{O/E}, \tag{5}$$

where D_{buf} is the delay for reading and writing the electrical sliced buffer, D_{xbar} is the delay for copying a packet from one input queue to one output buffer through the internal crossbar, and D_{wl} is the delay incurred by looking up the wavelength matrix to get the wavelength for the next hop. $\overline{D}_Q$ is the expected accumulated queuing delay in all en-route gateways. Actually $\overline{D}_Q$ is part of the delay incurred by gateways. We model it separately so that Part II only depends on $\overline{N}_{hop}$.

According to Eq. (4), $\overline{D}_{inter}$ is a function of $\overline{N}_{hop}$ and $\overline{D}_Q$ which depend on the traffic pattern. In the following, we model $\overline{D}_{inter}$ based on the Uniform-Poisson traffic pattern, with uniform distribution in space and Poisson distribution in time. Assume that each core generates packets following a Poisson distribution with the same injection rate of θ, and each core sends packets to all the other cores with the same probability. Hence, the traffic rate from any core i to any other core j is $\frac{\theta}{N-1}$, and the traffic follows a Poisson distribution. Thus, the proportion of intra-subsystem traffic α is $\frac{W_{\max}-g-1}{N-1}$.

Lemma 1. *The probability that an inter-subsystem packet traverses $2i-1$ hops of λ-routers is $P(2i-1)=(\frac{1}{R_i}-\frac{1}{R_{i-1}})\times\frac{N}{N-1}$, where $i=2,...,L$. The expected number of hops an inter-subsystem packet traverses is $\overline{N}_{hop}=\sum_{i=2}^{L}(2i-1)P(2i-1)$.*

Proof: Each inter-subsystem communication can be divided into two periods: upward communication and downward communication. There is always a turnover λ-router at which the communication changes from upward direction to downward direction. For example, in Fig. 3 L_3^0 is the turnover λ-router for the communication from core C_D to core C_E. Suppose that the turnover λ-router for an inter-subsystem communication is located in level i. Then the number of hops that this inter-subsystem communication traversed must be $2i-1$. Obviously, the probability that an inter-subsystem packet traverses $2i-1$ hops equals to the probability that the turnover λ-router for this packet is located in level i. If a level-i λ-router is a turnover router for a packet, the last λ-router from which the packet was received and the next λ-router to which the packet will be forwarded must be two different routers in level $i-1$. Let $P_h(i)$ represent the probability that a packet passes a λ-router in level i, and $P_h(\cap|i)$ denote the probability that the packet changes its direction from upward to downward at a level-i λ-router given that the packet passes through a level-i λ-router. Then $P(2i-1)=P_h(\cap|i)\times P_h(i)$.

In our architecture, each level-1 λ-router aggregates the upward traffic from $n=W_{\max}$–g cores, and segregates downward traffic to n cores. Each level-2 λ-router aggregates/segregates traffic from/to $\frac{R_1}{R_2}\times n=\frac{N}{R_2}$ cores. Each level-3 λ-router aggregates/segregates traffic from/to $\frac{R_2}{R_3}\times\frac{R_1}{R_2}\times n=\frac{N}{R_3}$ cores. Similarly, each level-i λ-router aggregates/segregates traffic from/to $\frac{N}{R_i}$ cores. If a packet passes a level-i λ-router, the destination of the packet must not be any core from which the level-i λ-router can aggregate its upward traffic. Since any packet generated by any core has the same probability to be sent to all the other cores, $P_h(i)=(N-\frac{N}{R_i})\times\frac{1}{N-1}$. For any packet that passes through a level-i λ-router, it can either keep going upward, or change its direction from upward to downward, depending on the destination of the packet. Hence, $P_h(\cap|i)=(\frac{N}{R_i}-\frac{N}{R_{i-1}})\times\frac{1}{N-\frac{N}{R_i}}$. Thus, $P(2i-1)=P_h(\cap|i)\times P_h(i)=(\frac{1}{R_i}-\frac{1}{R_{i-1}})\times\frac{N}{N-1}$, and the expected number of hops that an inter-subsystem packet traverses is $\sum_{i=2}^{L}(2i-1)P(2i-1)$. □

Assume that the routing structure as illustrated in Fig. 5 is a balanced and complete tree. The following theorem gives the expected en-route queuing delay for an inter-system packet. If the routing tree is not balanced and complete, $\overline{D}_Q$ in Theorem 2 is an upper bound on the expected en-route queuing delay.

Theorem 2. *The expected packet en-route queuing delay is*

$$\overline{D}_Q=\sum_{i=2}^{L}\Big(P(2i-1)\times\sum_{j=1}^{i-1}\frac{2\theta_j}{\mu_j(\mu_j-\theta_j)}\Big), \tag{6}$$

where $\theta_j=\frac{N^2}{N-1}\times\frac{R_{j-1}-1}{R_{j-1}^2}\times\frac{\theta}{g(W_{\max}-g)}$ and $\mu_j=\frac{s_p}{t_d}$. s_p is the packet size and $t_d=2D_{buf}+D_{xbar}+D_{wl}$.

Proof: For each inter-subsystem packet that traverses $2i-1$ hops, the packet will be routed by $i-1$ gateways in both upward and downward directions. Let T_i be the queuing delay at the j^{th} en-route gateway. According to Lemma 1, the probability that a packet traverses $2i-1$ hops is $P(2i-1)$. The expected accumulated queuing delay $\overline{D}_Q$ can be computed by

$$\overline{D}_Q=\sum_{i=2}^{L}\Big(P(2i-1)\times\sum_{j=1}^{2i-2}T_j\Big). \tag{7}$$

Since upward and downward traffics are processed separately using two independent paths, we analyse the queuing delay T_i in upward and downward directions, respectively.

(1) T_i in upward direction: Since each core generates traffic following a uniform distribution in space and a Poisson distribution in time with the same injection rate θ, the traffic injection rate at a gateway that bridges a level-$(j-1)$ λ-router and a level-j λ-router can be computed as follows: the traffic rate from a level-$(j-1)$ λ-router to a level-j λ-router is $\theta\times\frac{N}{R_{j-1}}\times\frac{N-\frac{N}{R_{j-1}}}{N-1}=\theta\times\frac{N^2}{N-1}\times\frac{R_{j-1}-1}{R_{j-1}^2}$. We use the example given in Fig. 5 to illustrate this computation. Consider the λ-router at level-3 and the left level-2 λ-router. $\frac{N}{R_{j-1}}=\frac{N}{R_2}$ is the number of cores whose upward traffic can pass the left level-2 λ-router. Since only the packets with destinations outside of the subtree rooted at the left level-2 λ-router will be routed upward to the level-3 λ-router, $\frac{N-\frac{N}{R_{j-1}}}{N-1}=\frac{N-\frac{N}{R_2}}{N-1}$ is the proportion of the packets that will be routed from the left level-2 λ-router to the level-3 λ-router. Since the packets from a level-$(j-1)$ λ-router to a level-j λ-router will be evenly routed via the g gateways that connect these two λ-routers, the traffic injection rate at each gateway that connects a level-$(j-1)$ λ-router to a level-j λ-router is $\frac{N^2}{N-1}\times\frac{R_{j-1}-1}{R_{j-1}^2}\times\frac{\theta}{g}$. According to our gateway design, each gateway needs to have $W_{\max}-g$ input queues to buffer the packets received from the lower-level λ-routers using $W_{\max}-g$ wavelengths since there is no communication between the g gateways. Thus, the upward traffic injected to each gateway will be dispatched to the $W_{\max}-g$ input queues following a uniform distribution. Let θ_j denote the traffic injection rate at one input queue in a gateway that connects a level-$(j-1)$ λ-router to a level-j λ-router. $\theta_j=\frac{N^2}{N-1}\times\frac{R_{j-1}-1}{R_{j-1}^2}\times\frac{\theta}{g(W_{\max}-g)}$.

In our gateway design, the input queues are fully connected with the output buffers, and each input queue has an independent packet dispatcher. Let t_d represent the packet dispatch delay which is defined as the average time interval between two adjacent packets that are sent out from the same output buffer. Then $t_d=2D_{buf}+D_{xbar}+D_{wl}$. Each input queue can be modelled as a FIFO queue system with injection rate of θ_j and service rate of $\mu_j=s_p/t_d$ where s_p is the average packet size. Each input queue subjects to the Birth-Death process [16] according to the queuing theory. Assuming the probability that q packets stay in the queuing system is $P_j(q)$ with an initial state of $P_j(0)$, and $P_j(q)=P_j(0)\times(\frac{\theta_j}{\mu_j})^q$ for $q\geq 0$. To achieve a stable queuing system, $\frac{\theta_j}{\mu_j}<1$, and the stable average

queue length, denoted by Q_j, is $Q_j = \sum_{q=2}^{\infty}[(q-1)P_q^j] = \sum_{q=2}^{\infty}[(q-1)(1-\frac{\theta_j}{\mu_j})(\frac{\theta_j}{\mu_j})^q] = \frac{\theta_j^2}{\mu_j(\mu_j-\theta_j)}$. According to the Little's Law theorem in queuing theory [16], the stable queue delay of each input queue is $T_j = \frac{Q_j}{\theta_j} = \frac{\theta_j}{\mu_j(\mu_j-\theta_j)}$.

(2) T_i in downward direction: The traffic rate from a level-j λ-router to a level-$(j-1)$ λ-router is $\theta \times (N - \frac{N}{R_{j-1}}) \times \frac{N}{N-1} = \theta \times \frac{N^2}{N-1} \times \frac{R_{j-1}-1}{R_{j-1}^2}$, which is similar to the traffic rate from a level-$(j-1)$ λ-router to a level-j λ-router because the downward process is symmetrical to the upward process. Since the routing of downward packets is the same as the upward packets through another independent path, we have $T_k = T_{2i-k-1}$, $k \in [1, i-1]$. Hence, $\overline{D}_Q = \sum_{i=2}^{L}\left(P(2i-1) \times \sum_{j=1}^{i-1} 2T_j\right) = \sum_{i=2}^{L}\left(P(2i-1) \times \sum_{j=1}^{i-1} \frac{2\theta_j}{\mu_j(\mu_j-\theta_j)}\right)$, where $\theta_j = \frac{N^2}{N-1} \times \frac{R_{j-1}-1}{R_{j-1}^2} \times \frac{\theta}{g(W_{\max}-g)}$, and $\mu_j = s_p/t_d$. □

Corollary 1. *To guarantee network stability, the maximum packet injection rate at each core should be smaller than* $\frac{s_p g(W_{\max}-g)}{t_d} \times \frac{(N-1)R_{L-1}^2}{N^2(R_{L-1}-1)}$.

Proof: To ensure network stability, each input queue in a gateway between levels j–1 and j should satisfy $\frac{\theta_j}{\mu_j} < 1$, i.e., $\frac{N^2}{N-1} \times \frac{R_{j-1}-1}{R_{j-1}^2} \times \frac{\theta}{g(W_{\max}-g)} < \frac{s_p}{t_d}$. Thus $\theta < \min_{j\in[2,L]} \frac{s_p g(W_{\max}-g)}{t_d} \times \frac{(N-1)R_{j-1}^2}{N^2(R_{j-1}-1)} = \frac{s_p g(W_{\max}-g)}{t_d} \times \frac{(N-1)R_{L-1}^2}{N^2(R_{L-1}-1)}$. □

V. Performance Evaluation

In this section, we evaluate the performance of WRH-ONoC through extensive simulations with different network sizes and traffic distributions. We also compare WRH-ONoC with the traditional hybrid (i.e., optical+electrical) architecture in terms of average packet communication delay and throughput.

A. Simulation Setup

We implement WRH-ONoC based on an open-source NoC simulator, Noxim [18]. The simulation settings are summarized in Table II. All optical devices including E/O&O/E converters and λ-routers work at a rate of 10 *Gbps* for each wavelength. All electrical devices use a system clock of 1 GHz, and thus one system cycle is 1 ns. The gateway processing delay incurred by buffering ($2D_{buf}$), wavelength look-up (D_{wl}), and packet dispatching (D_{xbar}) takes 4 cycles per packet. Each packet has a size of 64 bits.

TABLE II. Simulation Setting

Optical		Electrical	
E/O conversion rate	10 *Gbps*/wl	Clock frequency	1 GHz
O/E conversion rate	10 *Gbps*/wl	WRH-ONoC gateway delay	4 cycles
channel bandwidth	10 *Gbps*/wl	Mesh router delay	2 cycles
Packet size		64 bits	

We compare the performance of WRH-ONoC with the hybrid architecture proposed in [17] that combines an optical circuit-switched network with an electrical packet-switched network. In our simulations, the delay for a packet to pass a router in the hybrid architecture is set to the minimum, i.e., 2 cycles, and the electrical network is only used to transmit control messages (i.e., path setup/teardown). We also use Noxim [18] to simulate the hybrid architecture. Since Noxim is a simulator for electrical NoCs, we use it to simulate the path setup delay. The end-to-end packet delay is the path setup delay plus one cycle, assuming that the optical signal can be transmitted within one cycle once the path is setup. In our simulations, the buffer size for the input queues at both gateways in WRH-ONoC and electrical routers in the hybrid architecture are set to be infinite to compare their maximum transmission capacity. Each simulation lasts for 20,000 cycles with a warmup period of 1000 cycles.

B. Comparison with theoretical results

Fig. 6 compares the average end-to-end delay obtained from theoretical analysis and simulation for WRH-ONoCs with $\{N, W_{\max}, g\}$ configured to $\{400, 25, 5\}$ and $\{480, 30, 6\}$. It can be seen that, in both cases, the average delay measured in simulations keeps close to the theoretical results. When the injection data rate θ is small (e.g., $\theta \leq 10$ Gps/core), the average end-to-end delay is small (∼25ns) and remains stable because most packets do not experience much queuing delay at the gateways. When the injection data rate approaches to the maximum injection rate $\theta_{\max}$, the average delay increases dramatically because the network is becoming saturated.

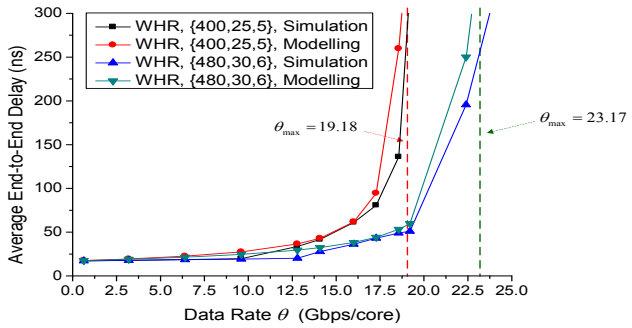

Fig. 6. Average end-to-end delay from theoretical analysis and simulation.

C. Impact of Network Size

In this set of simulations, we evaluate the performance of WRH-ONoC with the variation of the network size, and compare it with the hybrid architecture. Four groups of simulations were carried out with 320, 400, 480, and 640 cores, respectively. In the hybrid architecture, the cores are interconnected into mesh topologies in the form of 16x20, 20x20, 20x24, and 20x32. In WRH-ONoC, the cores are organized in a way that $\{N, W_{\max}, g\}$ are configured to {320,20,4}, {400,25,5}, {480,30,6}, and {640,40,8}, respectively.

Fig. 7(a) shows the average end-to-end delay with the variation of data injection rate. It can be seen that our scheme can achieve much lower end-to-end delay and much larger saturation data rate than the hybrid approach. When the traffic rate θ is very small (e.g., 0.1 *Gbps*/core), most of the packets will not suffer from queuing delay at the gateways in our scheme, and the average end-to-end delay achieved by our scheme is reduced by $\sim 50\%$ in comparison with the hybrid approach due to the less number of hops that the packets traversed. Moreover, in our scheme the maximum injection rate increases with the increase of the number of gateways

and wavelengths. The maximum injection rate achieved by our scheme is more than 3 times of that for the hybrid approach, owing to the non-blocking and WDM properties of λ-router. Fig. 7(b) shows the average throughput per core with the variation of the data rate. It can be seen that our scheme can achieve much higher throughput than the hybrid approach. When the number of wavelength is increased from 20 to 40 and the number of gateways is increased from 4 to 8, the per-core throughput is doubled even though the number of cores is also doubled (from 320 to 640). This is because that the proportion of intra-subsystem ($\frac{W_{\max}-g-1}{N-1}$) increases with the increase of $W_{\max}$, and the parallel data paths between two λ-routers increases with the increase of g. On the other hand, the performance of the hybrid approach deteriorates as the system size expands due to the linearly increased average hops and the more severe resource contention.

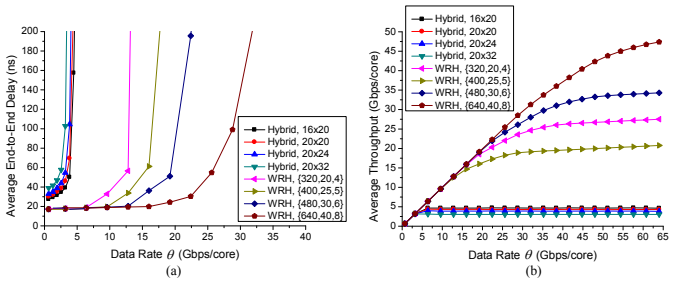

Fig. 7. (a) Average end-to-end packet delay; (b) Average throughput per core.

D. Impact of Traffic Pattern

In multi-core systems, communications tend to occur locally as tasks can be scheduled to be executed using only a subset of cores instead of all cores. In this set of simulations, we evaluate our scheme using a locality traffic pattern in which the proportion of intra-subsystem traffic α can be tuned manually. For example, 30% of packets will be intra-subsystem packets when α is 0.3. For each type of traffic (intra-subsystem or inter-subsystem), it still subjects to uniform distribution in space and Poisson distribution in time. Fig. 8 illustrates the impact of traffic distribution where $\{N, W_{\max}, g\} = \{400, 25, 5\}$. It can be seen that the gap between uniform traffic and locality traffic is more apparent with larger α and θ. This is because that, for uniform traffic, intra-subsystem communication only takes $\frac{n-1}{N-1} = 4.76\%$. Intra-subsystem packets only traverse one level-1 λ-router with low latency, while inter-subsystem packets need traverse several hops of λ-routers and gateways, and they may compete for the gateway resources along the route. Hence, the average end-to-end delay increases with the increase of θ, but decreases with the increase of α. The average throughput increases with both the increase of θ and α.

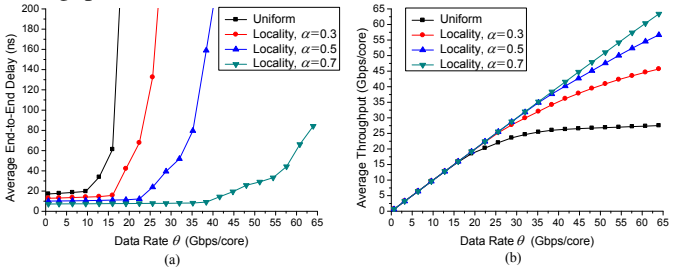

Fig. 8. (a) Average end-to-end packet delay; (b) Average throughput per core.

VI. Conclusions

This paper proposes a novel wavelength-reused hierarchical architecture, WRH-ONoC, for Optical Network on Chips. WRH-ONoC has three properties that ensure good scalability: (1) limited available wavelengths are reused in large systems; (2) performance increases with the increase of the number of available wavelengths and number of gateways; (3) hardware requirement is significantly reduced ($\sim$90% compared with the single λ-router architecture). Simulation results demonstrate that WRH-ONoC can achieve much lower average end-to-end delay and much higher throughput compared with the hybrid ONoC architecture by sacrificing a little bit hardware cost. Our future work is to further explore the tradeoff between hardware cost and performance, that is, to minimize the hardware cost subjecting to user-specified requirement on performance. Other future work includes broadcasting/multicasting scheme, fault-tolerant routing, analysis and evaluation of energy efficiency and communication reliability, etc.

References

[1] G. Nychis, C. Fallin, T. Moscibroda, O. Mutlu, S. Seshan, "On-Chip Networks from a Networking Perspective: Congestion and Scalability in Many-Core Interconnects," *SIGCOMM Conference*. pp. 407-418. 2012.
[2] International Technology Roadmap for Semiconductors Reports. 2013.
[3] Intel Newsroom, "Intel Re-architects the Fundamental Building Block for High-Performance Computing," http://newsroom.intel.com/, 2014.
[4] K. Preston, N. Scherwood-Droz, J. Levy, M. Lipson, "Performance Guidelines for WDM Interconnects Based on Silicon Microring Resonators," *CLEO Conference*. pp. 1-6. 2011.
[5] I. O. Connor, "Optical Solutions for System-Level Interconnect," *SLIP Conference*, pp. 79-88. 2004.
[6] Y. Kao, H. J. Chao, "Design of a Bufferless Photonic Clos Network-on-Chip Architecture," *IEEE Trans. Comput.*, vol. 63, pp. 764-776. 2014.
[7] A. Shacham, K. Bergman, L. Carloni, "On the Design of a Photonic Network-on-Chip," *NOCS Conference*, pp. 53-64. 2007.
[8] G. Kurian, J. Miller, J. Psota, J. Eastep, J. Liu, J. Michel, L. Kimerling, A. Agarwal, "ATAC: A 1000-Core Cache-Coherent Processor with On-Chip Optical Network," *PACT Conference*, pp. 124-133. 2010.
[9] D. Vantrease, R. Schreiber, M. Monchiero, M. McLaren, N. Jouppi, M. Fiorentino, A. Davis, N. Binkert, R. Beausolei, J. Ahh, "Corona: System Implications of Emerging Nanophotonic Technology," *ISCA Conference*, pp. 153-164. 2008.
[10] R. Morris, A. Kodi, A. Louri, "Three-Dimensional Stacked Nanophotonic Network-on-Chip Architecture with Minimal Reconfiguration," *IEEE Trans. Comput.*, vol. 63, no. 1, pp. 243-255. 2014.
[11] A. Liu, L. Liao, Y. Chetrit, J. Basak, H. Nguyen, D. Rubin, M. Paniccia, "Wavelength Division Multiplexing Based Photonic Integrated Circuits on Silicon-on-Insulator Platform," *J. Select. Topics Quantum Electron.*, vol. 16, no. 1, pp. 23-32. 2010.
[12] L. Zhang, M. Yang, Y. Jiang, E. Regentova, E. Lu, "Generalized Wavelength Routed Optical Micronetwork in Network-on-Chip," *IASTED Conference*, pp. 698-703. 2006.
[13] S. Beux, I. Connor, G. Nicolescu, G. Bois, P. Paulin, "Reduction Methods for Adapting Optical Network on Chip Topologies to 3D Architectures," *J. Micpro.*, vol. 37, no. 1, pp. 87-98. 2013.
[14] L. Chen, M. Lipson, "Ultra-Low Capacitance and High Speed Germanium Photodetectors on Silicon," *Opt. Express.*, vol. 17, no. 10, pp. 7901-7906. 2009.
[15] A. Guerre, N. Ventroux, R. David, A. Merigot, "Hierarchical Network-on-Chip for Embedded Many-Core Architectures," *NOCS Conference*, pp. 189-196. 2010.
[16] J. Sztrik, "Basic Queueing Theory," http://irh.inf.unideb.hu/jsztrik/education/16/SOR_Main_Angol.pdf. 2012.
[17] H. Gu, J. Xu, Z. Wang, "A Novel Optical Mesh Network-on-Chip for Gigascale Systems-on-Chip," *APCCAS Conference*, pp. 1728-1731. 2008.
[18] F. Fazzino, M. Palesi, D. Patti, "Noxim: Network-on-chip simulator." http://noxim.sourceforge.net/. 2014.