A Novel Approach to Optical Switching for Intradatacenter Networking

Limei Peng, Chan-Hyun Youn, *Member, IEEE*, Wan Tang, and Chunming Qiao, *Fellow, IEEE*

***Abstract*—In this paper, we propose to apply a novel paradigm called *labeled optical burst switching with home circuit* (LOBS-HC) for intradatacenter networking to provide a high bisection bandwidth and significantly reduce the cost and energy consumption associated with electronic packet switching. The unique features of LOBS-HC that make it more suitable than either optical circuit switching (OCS) or optical packet/burst switching are exploited to enable all-to-all communications with a guaranteed lossless transmission bandwidth between any given pair of pods, while also supporting bursty transmissions through wavelength-sharing among *home circuits* (HCs) and statistical multiplexing. As a case study, hypercube-like topologies are considered for the interconnection among the pods within a datacenter. In particular, we first propose a simple but efficient HC assignment scheme called *complementary HC* for 2-D cube or ring, and then extend our works to *n*-cube and *generalized hypercube* by applying the concept of *spanning balanced tree* (SBT) for their HC assignment. Our analysis results show that with such datacenters, the minimum number of wavelengths needed in each case is significantly reduced from that needed with OCS and also, the network cost in terms of wires and transceivers needed is considerably reduced from that incurs with datacenters using electronic packet switching. We then evaluate the traffic performance of such hypercube-based datacenters using LOBS-HC through simulation experiments via the OPNET simulator. The performance results obtained for a variety of communication patterns and traffic models within a datacenter demonstrate the feasibility of the proposed approach.**

***Index Terms*—Datacenter, labeled optical burst switching with home circuit (LOBS-HC), home circuit (HC) assignment, hypercube.**

Manuscript received July 14, 2011; revised November 01, 2011 and December 13, 2011; accepted December 13, 2011. Date of publication December 21, 2011; date of current version January 25, 2012. This research was supported in part by the Future-Based Technology Development Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science, and Technology (NRF 2011-00230522), KAIST, Korea, in part by the R&D program of MKE/KEIT [10039260], KAIST, Korea, and in part by the State Key Laboratory of Advanced Optical Communication Systems and Networks (2009SH02, 2011GZKF031110), Shanghai JiaoTong University, China. A part of this work appeared in ICC'2011 and the Workshop on Cloud Computing (colocated with INFOCOM 2011) [30], [31].

L. Peng is with the School of Electronic and Information, Soochow University, Suzhou 215006, China (e-mail: auroraplm@suda.edu.cn).

C.-H. Youn is with the Department of Electrical Engineering, GRID Middleware Research Center, KAIST, Daejon 305, Korea (e-mail: chyoun@kaist.ac.kr).

W. Tang is with Computer Intelligence Lab, College of Computer Science, South-Central University for Nationalities, Wuhan 430074, China (e-mail: tangwan@scuec.edu.cn).

C. Qiao is with the Department of Computer Science and Engineering, State University of New York at Buffalo, NY 14228 USA (e-mail: qiao@computer.org).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/JLT.2011.2180888

I. Motivation

A LARGE datacenter consists of tens to hundreds of *pods*, each containing about a thousand servers and each server is equipped with a 1 Gbps or 10 Gbps Ethernet port. A server may communicate with any other sever and consequently require a huge capacity in the core of an intradatacenter network (iDCN) in order to provide the full bisection bandwidth for communications among all the pods. Having efficient interconnection topologies may help decrease network complexity and increase communication efficiency. A great deal of efforts has been made to design scalable interconnection topologies using electronic packet switching, such as Fat-Tree, BCube, and DCell in [1]–[3]. However, using these electronic switching based interconnections makes it difficult to provide the full bisection bandwidth for all-to-all communications since a huge number of wires and transceivers would be needed, which also implies a high cost and power consumption. Although existing datacenters using electronic switching may perform reasonably well with a high oversubscription ratio, as the number of servers in each pod and/or the number of pods increase further, the potential communication bottleneck and a huge cost and power consumption associated with a large number of transceivers in the core are still major concerns.

Optical switching networks are known to be able to provide high bandwidth, and reduce the cost and energy consumption associated with electronic switching. However, it is not trivial to apply optical switching to iDCN. On one hand, with an iDCN, the short distance (of about few hundred meters) between the pods provides a friendly environment for optical switching of data since there is little need to worry about physical layer impairments, which are typical in a long-haul optical network. On the other hand, within an iDCN, traffic in the core is much more bursty and unpredictable [6], [8] than that in the Internet Core and in addition, reducing the delay between two pods plays a very critical role in increasing the performance and utilization of an iDCN.

To alleviate the scalability problem of electronic switching, a hybrid optical and electrical switching architecture called Helios has been proposed in [4], wherein some optical circuit switches (OCS) [14]–[17] are used as core switches, with the intention being to switch "elephant" (i.e., long) flows optically over wavelength circuits, while "mice" (i.e., short) flows are switched electrically. A similar hybrid architecture called hybrid packet and circuit (HyPaC) has been proposed in [5]. Though such hybrid switching approaches may alleviate some of the problems of all electronic switching such as power consumption and cost, we believe they are only short-term compromises in that OCS is neither agile enough to handle bursty data, nor bandwidth efficient due to the fixed and coarse wavelength granularity.

An ultimate solution would be to replace all the existing electronic switches in the core (e.g., the Ethernet switches) with an optical switching network that converts all data coming from source pods, in the form of e.g., Ethernet frames, into optical packets/bursts, and switches such optical packets/bursts in the optical domain to the desired destination pods. However, there are two major challenges to be overcome. One is the lack of a mature technology to make such a fast optical switching fabric that is also large enough (in terms of number of inputs/outputs) for use in the core to replace the proposed slow but large micro electro mechanical systems (MEMS) in the hybrid switching approaches. Current photonic technologies may be able to produce fast but small optical switching fabrics only. The other major challenge is that, although one may use multiple small-and-fast optical switching fabrics to construct a multihop optical switching network in the core of an iDCN, how to effectively achieve lossless transmission and guarantee transmission bandwidth in the absence of optical buffer in such a multihop optical network.

II. Overview

To address the above two challenges, we apply a novel optical switching paradigm called labeled optical burst switching with home circuit (LOBS-HC), to provide all-optical interconnection among pods, and support optical switching of either mice or elephant flows at the burst granularity. LOBS-HC was first proposed to improve upon IP-over-WDM and packet-optical transport systems (P-OTS) in a core network in [18], [19]. LOBS-HC improves over OCS (which characterizes IPover-WDM or P-OTS) by efficiently support bursty traffic and statically multiplexing just as in OBS, and thus reducing the need for wavelength resources and end-to-end delay. However, LOBS-HC also improves upon OBS (which is not yet commercially available) to effectively achieve lossless transmission and bandwidth guarantee which OBS is unable to.

There are many open research problems associated with LOBS-HC in a core network that have not been addressed in [18], [19]. In this study, we apply this novel concept to iDCN and address some of the open problems such as how to construct efficient (i.e., good bisection width, good scalability with low complexity) interconnection topologies among pods/core switches, how to efficiently route and group multiple HCs so as to minimize the number of required wavelengths while achieving good traffic performance, and what are the cost and performance of the propose approach.

More specifically, in the study, we will consider a large datacenter consisting of N (i.e., tens to hundreds) core switches and N pods, where N pods are interconnected with core switches via LOBS-HC network and the N core switches are interconnected with each other forming a n-dimensional hypercube (i.e., n-cube) or generalized n-dimensional k-ary hypercube ($\mathrm{GHC}_{n,k}$). Note that hypercube is considered to be unsuitable to directly interconnect a large number of servers, via electronic packet switching, since its nodal degree is $\log_2 M$, where M is the total number of servers (which could be a thousand time larger than N) [3]. Nevertheless, since hypercube-like topologies have good bisection width, and we will use them to construct a two-layer interconnection architecture where only tens to hundreds of pods/core switches are to be interconnected, we believe they are still a promising interconnection candidate in the core of a large iDCN.

In addition to the proposed use of LOBS-HC in the two-layer hypercube-like interconnection topologies, this paper makes the following additional contributions. One is the development of optimal HC routing and grouping algorithms for such kind of hypercube-based datacenters with LOBS-HC, so as to minimize the network resources (i.e., wavelengths, transceivers and switches) needed to establish all the required HCs to support all-to-all communications. Note that the unique features of LOBS-HC (e.g., wavelength-sharing among the HCs and lossless transmission) make the problem of HC routing and grouping quite different from the traffic grooming problem in OCS (and OBS) networks, and to our best knowledge, this is the first such work on optimal HC routing and grouping.

More specifically, as a part of these contributions, we first discuss the 2-dimensional cube or ring interconnection topology as a case study and propose a simple HC assignment scheme called complementary HC assignment (CHA) to minimize the number of wavelength required. We then extend our works to n-cube and $\mathrm{GHC}_{n,k}$, and apply the concept of spanning balanced tree (SBT) to assign HC as well as minimizing the number of wavelengths required. Results from numerical analysis show that using LOBS-HC in the core of an iDCN requires much fewer wavelengths than using OCS, and much fewer wires, transceivers and switches than using electronic switching.

Another major contribution of this work is the traffic performance evaluation of such datacenters via simulation. Various communication patterns and traffic models typical in datacenters are studied and the simulation results show that LOBS-HC together with the hypercube topologies can support intradatacenter communication well.

We note that a few all-optical packet/frame based switching approaches have been proposed, see e.g., [6] and [7]. In [6], a DOS architecture using a single arrayed waveguide grating router (AWGR) was proposed and shown to perform better than not only a couple of other optical switching approaches such as OSMOSIS [9], [10] and Data Vortex [11], but also an electronic switching using a flattened butterfly topology [29]. The approach in [6] uses optical label switching (OLS) where each packet carries a label similar to a (burst) control packet in OBS, however, unlike in OBS, the approach in [6] requires an array of tunable wavelength convertors (TWCs), an array of fiber delay lines (FDLs), and a loopback shared electronic buffer (with additional O/E and E/O conversion circuits), as well as a centralized controller that determines which wavelength(s) to use and when to transmit for all the packets among all the pods.

In [7], a three-stage Clos-based architecture using AWGRs at each stage was proposed to address the scalability issue of a single AWGR. An array of TWCs are used between the first and second, and the second and third stages, respectively. Similar to [6], global (i.e., network-wide) scheduling of all the packets among all the pods is needed to determine the wavelength and transmission time of each packet transmission, although in [7], multiple schedulers, which coordinate with each other through iterative algorithms, are used to perform the global scheduling.

In contrast to [6], [7], in the proposed LOBS-HC approach, a packet goes through multiple switches, and switching is performed at each node (i.e., under distributed control).

The rest of the paper is organized as follows. In Section III, we introduce LOBS-HC ring and propose the CHA scheme for HC assignment and grouping to establish the foundation for the following presentation. In Section IV, we extend the study to n-cube and GHC, discuss a two-level hypercube-based interconnection architecture, and develop the HC routing and grouping scheme by resorting to the concept of SBT. In Section V, we evaluate the proposed approaches via numerical analysis and simulation experiments and compare the obtained results with datacenters of the same scale using optical circuit switching or electronic switching. In Section V, we give some concluding results.

III. LOBS-HC in a 2-Cube or Ring

A. Overview of LOBS-HC and Intradatacenter Networking Design

In this subsection, we first give a brief review of how LOBS-HC enhances OBS and outperforms OCS, and then discuss a few general design principles related to the application of LOBS-HC to iDCN.

With OBS, data bursts can be statistically multiplexed to achieve sharing of the wavelength resources, but such sharing also introduces resource contention among bursts and results in burst losses and a poor QoS guarantee. On the other hand, with OCS, lossless transmission and a good QoS can be achieved by dedicating a wavelength to each flow, but even a low bandwidth "mice" flow may take up an entire wavelength resulting in a waste of wavelength resources. LOBS-HC has been proposed to inherit the lossless transmission as in OCS and the efficient statistical multiplexing among the bursts as in OBS by assigning *home circuits* for all source-destination (S-D) pairs.

Fig. 3 shows a typical node architecture used in an OBS network. Each node consists of a controller, a fast optical switching fabric, a pair of optical transceivers, a burst assembly/disassemble module, an electronic switching fabric and N destination queues. Several OBS testbeds and prototypes, albeit not for DCN, have been built and reported where switching fabrics made of small (at most 8×8) PLZT and SOAs were [12], [13]. Simply put, packets from a source pod (at the bottom) to the same destination pod first go into a queue corresponding the destination, and are then assembled into a larger data burst for transmission. A similar but reverse process takes place when receiving data bursts from a core switch.

LOBS-HC uses almost the same architecture and switching fabrics as OBS, with the major differences between LOBS-HC and OBS lying in the controller and control plane.

More specifically, within LOBS-HC, each source node establishes a home circuit (HC) for each destination node with any required bandwidth first before any data transmission commences, and then will transmit traffic in a burst-by-burst manner along the assigned HC. In particular, if a traffic flow for a given S-D node pair (s, d) requires x units of bandwidth (where x is assumed to be lower than the wavelength capacity C), then network using LOBS-HC will establish a HC using one wavelength

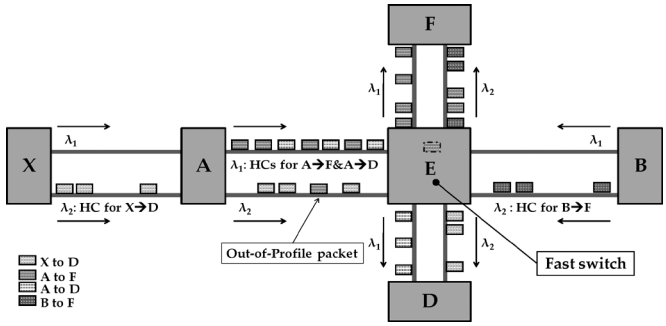

Fig. 1. LOBS-HC.

for it and ensure that flow will receive the required x units of bandwidth along the established HC. This is accomplished by allowing other HCs to use the same wavelength as long as the total bandwidth required by all the HCs sharing the same wavelength does not exceed C. These HCs are thus similar to virtual circuits (VCs) in that they are logical, however, unlike how VCs are multiplexed, only those HCs from the same source node but to different destinations may share the same wavelength. Accordingly, bursts sent from different sources will never collide at intermediate nodes (as they may in OBS). When a collision happens between an InP burst and an OoP burst, the InP burst will always win.

Take Fig. 1 as an example. The two HCs from A to F and A to D, respectively, can share the same wavelength from A to E (i.e., $\boldsymbol{\lambda_1}$) as long as each requires no greater than $C/2$ units of bandwidth. We refer to the traffic whose amount does not exceed the requested bandwidth for the HC *In-Profile (InP)* traffic, and as an example, the packets filled with leaning stripes from A to F shown in Fig. 1 are such In-Profile traffic, and these In-Profile traffic will experience lossless transmission in LOBS-HC networks.

In addition to enable statistical multiplexing on the same wavelength among multiple HCs (which cannot be done in OCS), LOBS-HC also facilitates the opportunistic transmission of the so-called *out-of-profile* (*OoP*) traffic. For example, additional bursts from A to F (filled with leaning stripes) in Fig. 1 may be injected onto a HC for a different S-D pair assigned to $\boldsymbol{\lambda_2}$. As long as such Out-of-Profile traffic is given a lower (preemptable) priority, lossless transmission of the In-Profile traffic along its HC is not affected. A preempted OoP burst is lost and may be retransmitted later by its sender. Readers interested in more details about the preemption process are referred to [18], [19]. Note that packets may arrive out-of-order, but such an issue will be handled by the transport layer protocol (e.g., TCP) above and is out of the scope of this work.

Note that the above discussion also implies that LOBS-HC can transmit bursty data as efficiently as electronic packet switching since it can guarantee lossless transmission for In-Profile traffic (as in OCS), but with much less power consumption and cost than electronic packet switching since LOBS-HC does not require any O/E/O conversions (nor optical buffers).

In our design of an iDCN of N pods, it is assumed that every pod needs to communicate with every other pod and hence a general-purpose iDCN should support is the so-called all-to-all

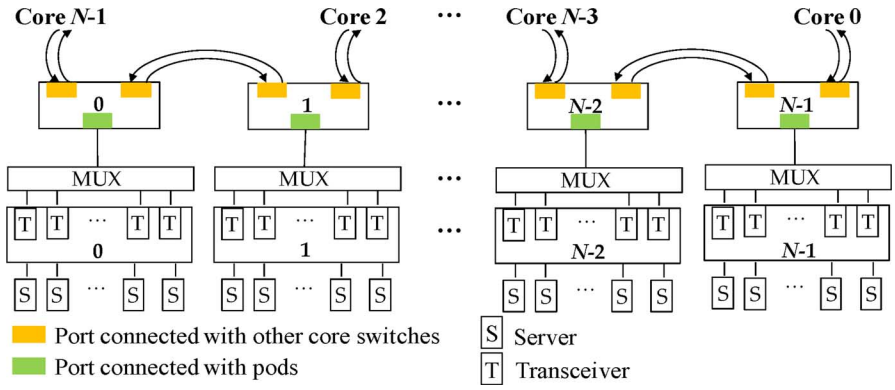

Fig. 2. LOBS-HC ring: the core switches (top) and the pods (below).

communications. In addition, although the amount of traffic between different pairs of pods may differ, that amount is unpredictable, and hence the network should *not be designed to support any specific uneven traffic distribution among the pods*. Accordingly, we will design the network for the all-to-all communications pattern with even traffic distribution, assuming that the amount between any given pair is the same. Nevertheless, since traffic is dynamic and will be unevenly distributed during the operation of a datacenter, we will evaluate the performance of the proposed datacenter design through simulations assuming various communications patterns and traffic distributions.

In particular, when designing a general-purpose intradatacenter network, we will assume that a pair of pods needs a sustained bandwidth of B_W to each of the other pods. In what follow, we denote by B the normalized bandwidth required by each HC, i.e., $B = B_W/C$, where C is the per wavelength capacity. We also denote by H the maximum number of HCs that can share one wavelength, where $H = \lfloor 1/B \rfloor$.

In order to compare with existing electronic switching based iDCN in terms of the number of wavelengths, wires, transceivers, and switches, we assume by default that $B_W = 10$ Gbps and $C = 40$ Gbps or 100 Gbps and hence $H = 4$ or 10. Note that having $B_W = 10$ Gbps means there will be one HC at 10 Gbps from any source pod to any destination pod. This number may seem low, but we note that if there are 100 pods, the total sustained interpod data throughput per pod would reach 990 Gbps (which is a decent target given that there are 1000 servers per pod, each having a 1 GE or at most 10 GE port). Besides, the total traffic through the core would reach almost 100 Tbps, which is a representative number for a current electronic switching based iDCN.

We also note that for the iDCN based on LOBS-HC built under the above default assumption, the burst rate at which a pair of pods can communicate is not limited to this sustained rate of 10 Gbps, and can be much higher as the source pod S can send additional data as out-of-profile traffic to a destination pod D using HCs other than the one established for S and D. In addition, since pods are relatively close to each other (less than 1 km or so), one may have close to a hundred of wavelengths per fiber, with each wavelength operating at $C = 400$ Gbps or 1 Tbps. Under this short-to-medium term assumption, each S-D pod pair can have a sustained rate of 100 Gbps, and the total traffic can be 1 Pbps. Since there will be no O/E/O in the core, scaling both B_W and C up by 10 times (or more) will not change H, nor the results on the number of wires, transceivers and switches needed for the proposed LOBS-HC based iDCN. In addition, such scaling can be achieved without having to further reduce the switching/reconfiguration time due to the rate transparency of the optical switching.

Since iDCN is fiber rich in that many fibers can be prewired easily, instead of using Dense WDM (DWDM) technologies (whereby a hundred of wavelengths can be multiplexed on a single fiber), it may be more cost-effective to use Coarse WDM (CWDM) whereby only a few tens of wavelengths are multiplexed onto each fiber. With CWDM, however, a few (up to ten for example) fibers need to be used to bring the total number of wavelength channels to the same number of wavelength channels on a DWDM link. To simplify our presentation below, we will assume a DWDM link between two adjacent switches when determining the number of links (fibers) between all switches, L, and the number of wavelengths on each fiber, W Nevertheless, we note that in a more cost-effective CWDM implementation, the actual number of fiber links will be a few times more, and correspondingly, the number of wavelengths per fiber will be a few times less, since the product of the two, i.e., $L \times W$, remains the same.

Below, as the first step of studying LOBS-HC in hypercube-based interconnection of core switches within a datacenter, we consider the ring topology (which is more general than a 2-dimensional cube).

B. LOBS-HC in Rings

Fig. 2 shows a datacenter with N pods interconnected in a bidirectional LOBS-HC ring, consisting of N core optical switches, each of which can switch an optical burst of an arbitrary duration without O/E/O conversion.

In a LOBS-HC ring, each source pod will be allocated one HC (providing the sustained bandwidth of $B_W = 10$ Gbps) for each destination pod. A major advantage of the proposed LOBS-HC implementation is that many (by default, $H = 10$) HCs originating from a source pod to destination pods can be effectively multiplexed onto one wavelength operating at a higher rate (of $C = 100$ Gbps by default). More specifically, as shown in Figs. 2 and 3, a possible implementation of the LOBS-HC ring is to use just a pair of WDM links connecting a pod to an

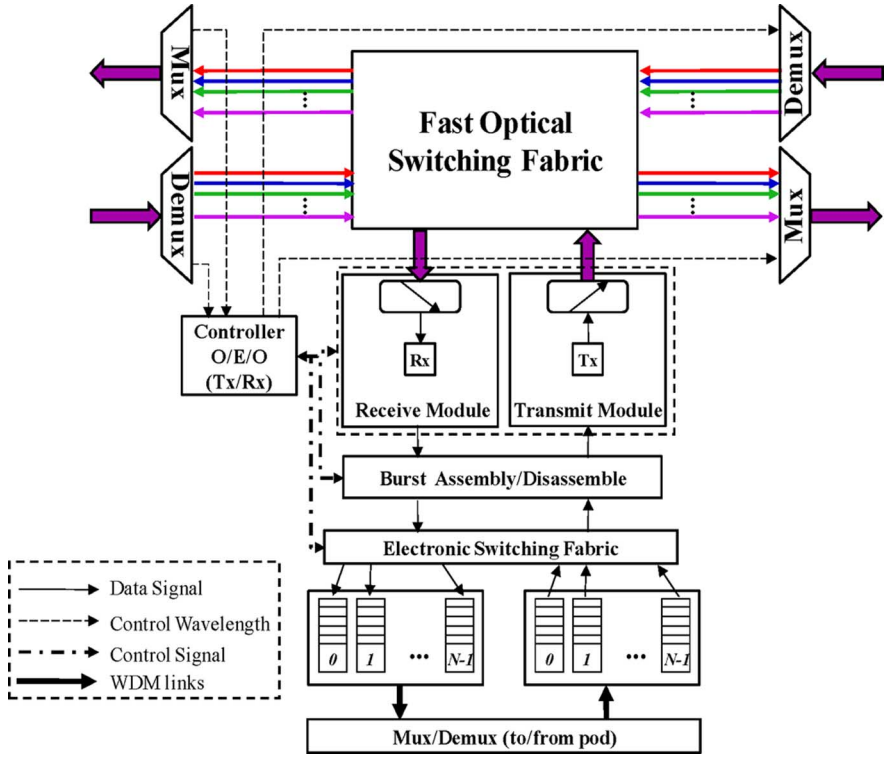

Fig. 3. Node architecture of Core Switches.

optical core switch, which in turn is connected to two neighboring core switches (for the bidirectionality). We note that the conventional wavelength routing using OCS would be quite effective if each pod has a relatively long "flow" to send to another pod at a rate of say 100 Gbps. On the other hand, if each pod does not have such a long flow to every other pod, wavelength routing would require either one dedicated wavelength for each HC, or using O/E/O conversion (barring all-optical wavelength conversion) to groom multiple HCs onto the same wavelength at the intermediate nodes. As to be shown, neither is as cost-effective as the proposed LOBS-HC approach. In addition, such a OCS-based approach cannot easily take advantage of the advance in WDM transmission in that as the per wavelength rate increases to 400 Gbps or 1 Tbps, it would be more and more difficult to find a long flow that can sustain at this higher rate to be effective.

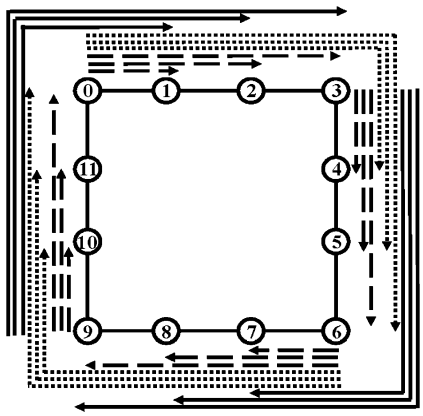

Fig. 4. Wavelength assignment for HCs using CHA.

C. Complementary HC Assignment (CHA) in LOBS-HC Rings

One basic idea of the proposed CHA is to spatially reuse the same wavelength by different nodes to originate their respective HCs. Let G be the number of such source nodes that can spatially reuse the same wavelength to establish their nonoverlapping HCs, we have $G = \lceil N/H \rceil$ and hereafter G will also be referred to as the (maximum) reuse factor.

Fig. 4 gives an example of a ring topology consisting of 12 pods using CHA. The nodes are numbered from 0 to 11 and we assume the normalized bandwidth of each flow is 0.3, i.e., $B = 0.3$. Then, $H(= 3)$ HCs originating from the same node can be established by sharing the same wavelength, and the spatial reuse factor is $G = 4$, meaning that up to four nodes can use the same wavelength to establish a total of 12 HCs on the same wavelength. For example, each of the four nodes 0, 3, 6, and 9 can originate 3 HCs, which are 1, 2, and 3 hops away from the respective sources and can share the 1st (i.e., dashed) wavelength. In addition, nodes 0 and 6 can both use the second (i.e., dotted) wavelength to construct 3 HCs to nodes that are 4, 5, and 6 hops away from them respectively. Similarly, nodes 3 and 9 can both use the third (i.e., solid) wavelength to construct 3 HCs to nodes that are 4, 5, and 6 hops away respectively. Since the maximal number of hops in the 12-node bidirectional ring is 6 hops, three wavelengths are sufficient to establish all the HCs taking nodes 0, 3, 6, and 9 as source nodes.

Based on the above example, we can describe the basic ideas of CHA as follows. For simplicity, we assume that N is a multiple of H for the time being and moreover, $G(= N/H)$ is even. First, we divide the N pods into H groups, denoted by

$G_1, G_2, \ldots, G_H$, via H-way interleaving. That is, G_i, where $i = 1, 2, \ldots, H$, consists of nodes $\{i-1, H+i-1, 2H+i-1, (G-1)H+i-1\}$. Note that the indices of two adjacent nodes within a group differ by H. For example in Fig. 4, nodes 0, 3, 6, and 9 are in G_1 while nodes 1, 4, 7, and 10 are in G_2 and so on.

Without loss of generality, we will only describe the proposed CHA for the nodes in G_i since each group needs the same number of wavelengths, although they belong to a distinct set, to establish the HCs for their member nodes. Readers can refer to the example in Fig. 4 by replacing i with 1, H with 3 and G with 4.

We will examine the HCs originating from the member nodes in G_i based on their hop lengths, starting with the establishment of all the HCs whose hop lengths are from 1 to *H in stage 1*, moving on to those with hop lengths of $H+1$ to $2H$ in stage 2, and so on, all the way to those with hop lengths of $(G/2-1)H+1$ to $G/2 * H$ in stage $G/2$.

To begin with, we use the 1st wavelength to establish HCs from its first member node $(i-1)$ to another H nodes that are 1 to H hops away from it as respective destination. The same wavelength is also used by every other member node in G_i in a similar way. This takes care of the stage 1 establishment of all the HCs originating from the member nodes in G_i that are 1 to H hops long. Next, as the first phase in stage 2, we use the second wavelength to establish HCs from its first member node $(i-1)$ to H of its destination nodes that are $H+1$ to $2H$ hops respectively. The same second wavelength is also used by some of the other member nodes $2H+i-1$, $4H+i-1$ and so on in a similar way during this phase. In the next phase of stage 2, all the remaining member nodes $H+i-1$, $3H+i-1$, and so on use the third wavelength to establish their respective HCs to their respective destination nodes that are $H+1$ to $2H$ hops away. In this way, the total number of wavelengths needed in stage 2 to establish all the HCs originating from the member nodes in G_i to all of their respective destination nodes that are $H+1$ to $2H$ hops away is 2.

We can generalize the above algorithm that establishes all the HCs originating from the member nodes in G_i of length between $(j-1)H$ to *jH* hops long, where $1 \le j \le G/2$, in stage j using $j/2$ wavelengths. Accordingly, the total number of wavelengths needed in all $G/2$ stages to establish all the HCs originating from the member nodes in G_i, (and in fact, any other group), denoted by W_G, can be calculated by (1)

$$W_G = 1 + 2 + \cdots + G/2 = G(G+2)/8, \text{when } G \text{ is even.} \tag{1}$$

From the above description, we also note that during each stage, the number of transceivers needed per pod in each of the two directions (clockwise and counter-clockwise) is one. Accordingly, the number of transceivers needed per pod per direction is $G/2$, and the total number of transceivers per pod is G, which makes the proposed approach quite feasible (given G is around 10 to 20).

Due to the space limitation, we will omit the proof that $G(G+2)/8$ is also the minimum number of wavelengths needed, but suffice it to say that CHA is optimal since that it wastes minimal wavelength resources when establishing HCs. In addition, we present the following results without further proof

$$W_G = 1 + 2 + \cdots + (G+1)/2 = (G+1)(G+3)/8, \quad \text{when } G \text{ is odd} \tag{2}$$

$$W_{\mathrm{TTL}} = \begin{cases} \frac{H(G+1)(G+3)}{8} + \frac{R(G+1)}{2}, & \text{when } G \text{ is odd} \\ \frac{HG(G+2)}{8} + \frac{RG}{2}, & \text{when } G \text{ is even} \end{cases} \tag{3}$$

where W_{TTL} denotes the total number of required wavelengths for all the H groups, and R is the remainder obtained from dividing N by H, i.e., $R = N \bmod H$.

Note that in a N-node OCS ring, a dedicated wavelength path is needed for each pair of pods even if the effective bandwidth is low (i.e., $B = 0.25$ or 0.1). The minimum number of wavelengths needed by the OCS ring can be obtained from (4) in [20] by setting $\alpha = 1$. Note that (4) can also be used to obtain the minimum number of wavelengths needed in an *O/E/O ring* where every core switch is equipped with O/E/O transceivers to perform wavelength conversion and traffic grooming by setting $\alpha = H$. In fact, that number of wavelengths is the minimum needed to provide one 10 Gbps "circuit" for each pair of pods as long as we use enough transceivers at the switches

$$W_{\mathrm{TTL}}(\mathrm{OCS}) = \begin{cases} \frac{(N+1)(N+3)}{8a}, & \text{when } N \text{ is odd} \\ \frac{N(N+2)}{8a}, & \text{when } N \text{ is even.} \end{cases} \tag{4}$$

IV. LOBS-HC in Hypercube-Based Datacenters

After studying LOBS-HC in 2-cube and its generalization (ring), we now extend our study to n-cube $(n > 2)$ and n-dimensional k-ary generalized hypercube ($\mathrm{GHC}_{n,k}$). Below, we describe how to effectively apply LOBS-HC in a n-cube and GHC to minimize the wavelengths needed to provide high bisection bandwidth for interpod communication.

A. Two-Layer Interconnection Architecture

We first describe the two-layer interconnection architecture using a n-cube/$\mathrm{GHC}_{n,k}$ to interconnect the core switches, as well as how routing is done in such a network. In a n-cube, the total number of nodes is $N = 2^n$ and each node is connected with n neighbors. Each vertex h, is represented by a binary number with the length of n, i.e., $h = a_{n-1}a_{n-2}\ldots a_1a_0$, where $a_i = 0$ or 1. In $\mathrm{GHC}_{n,k}$, the total number of nodes is $N = k^n$ and each node is connected with $n(k-1)$ nodes. Each vertex g, is represented by an n-digit k-radix number with the length of n, i.e., $g = v_{n-1}\ldots v_{p+1}v_pv_{p-1}\ldots v_0, 0 \le v_i \le k-1$.

Figs. 5 and 6 illustrate 3-cube and $\mathrm{GHC}_{2,3}$ based two-layer architectures, respectively. In Fig. 5, the core switches are interconnected in a 3-cube topology, with each core having three direct neighbors, one in each dimension. To simplify the discussion below, each pod is also associated with a binary number, and assumed to have a pair of WDM fibers to carry outgoing/incoming traffic between pods. The optimal interconnection between each pod, e.g., 000, and the core switches is to use one fiber to interconnect pod 000 to core switch 000, and the other fiber to interconnect pod 000 to core switch 111. Such a pod-to-

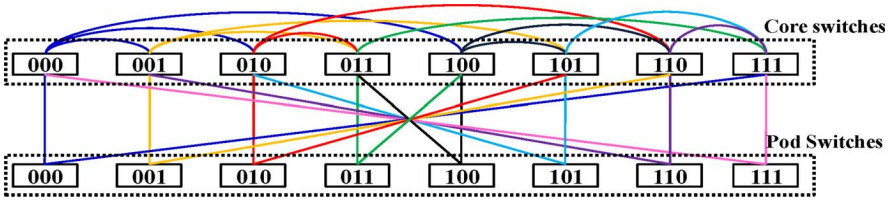

Fig. 5. Logical interconnection for 3-cube.

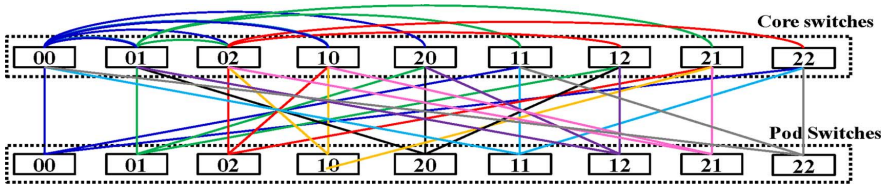

Fig. 6. Logical interconnection for $\mathrm{GHC}_{2,3}$.

core interconnection pattern can effectively increase the bisection bandwidth and reduce interpod communication latency as well as improve reliability, compared to a naïve interconnection where all the two fibers are used to interconnect pod 000 to core 000 for example.

In general, given a n-cube topology among the core switches, routing from source pod numbered $a_{n-1}a_{n-2}\ldots a_1a_0$ to destination pod numbered $a_{n-1}b_{n-2}\ldots b_1b_0$ will normally use the 1st fiber to reach the ingress core switch also numbered $a_{n-1}a_{n-2}\ldots a_1a_0$ first, while routing to destination pod numbered $\overline{a_{n-1}}b_{n-2}\ldots b_1b_0$ will use the 2nd fiber to reach the ingress core switch numbered $\overline{a_{n-1}a_{n-2}\ldots a_1a_0}$ instead (although in each case, either fiber can be used without necessarily increasing the hop lengths). Afterwards, hypercube routing among the core switches is performed until an appropriate egress core switch for the destination is reached. Due to space limit, we will omit the generalization of the above discussion to cases where there are more than two outgoing fibers per pod, or where a GHC is used instead of n-cube. Below, we will focus on routing among the core-switches within a n-cube or GHC using LOBS-HC.

B. Spanning Balanced Tree (SBT)-Based HC Assignment

In this subsection, we propose efficient HC routing, grouping and wavelength assignment algorithms in a n-cube or GHC so as to minimize the number of wavelengths needed. Unlike in the case of a LOBS-HC ring, optimal HC routing, grouping and wavelength assignment in a n-cube or GHC can be quite tricky. For one thing, in a n-cube, there are n disjoint shortest paths between two nodes whose hamming distance in their binary addresses is n, and many more nonshortest paths to choose from. Which route is optimal for a HC depends on at least how other HCs from the same source are routed since multiple of these HCs need to share a given wavelength.

Intuitively, in order to minimize the number of wavelengths, one would like to route the HCs such that no link in a n-cube or GHC is overloaded with too many HCs. In other words, one would like to *minimize the maximum* number of HCs routed over any given link. This is because if link l is used to route the highest number of HCs, say M, then it needs at least M/H wavelengths, which becomes a lower bound on the number of wavelengths needed on any link.

Considering the symmetric properties of a n-cube and GHC, and in particular, the fact that each node in a n-cube has n (and in a GHC, $n(k-1)$) outgoing links and $N-1$ destinations, the above intuition implies that any given source needs to route the $N-1$ HCs as evenly as possible using the n outgoing links. In other words, all the outgoing links should be used, and each of them *ideally* should not be used to reach more than $[(N-1)/n]$ destinations in a n-cube [and $[(N-1)/n(k-1)]$ in a GHC]. Another required feature of the optimal HC routing algorithm is that the shortest path should be used since otherwise, wavelength capability may potentially be wasted on the nonshortest path.

Fortunately, to establish all HCs from a given source node, we may adopt the concept of spanning balanced tree (SBT) proposed for n-cubes and GHCs in [21], [22] to meet the previously stated requirements. There may be multiple ways to construct a SBT in a n-cube and GHC, Figs. 7 and 8 show example SBTs rooted at node 00000 in a 5-cube and at node 000 in a $\mathrm{GHC}_{3,4}$, respectively. Note that the SBT contains 5 subtrees (STs) in Fig. 7, each providing shortest paths to 6 or 7 destinations. In addition, each of the 31 destinations is included in one and only one ST, and different STs use different links, thus they form a spanning tree without any loop.

We now briefly describe how to construct a SBT in n-cubes as follows. The idea is to construct a path from a given source s to any given destination h in a reversed order by finding the parent node for h first, and the grand-parent and so on. More specifically, we first rotate the binary representation of $h = a_{n-1}a_{n-2}\ldots a_1a_0$, to the right $(n-1)$ times, producing $(n-1)$ different values, and out of which, we record the smallest value. Let j be the least number of right rotations to produce that minimum value, which is denoted by $R^j(h)$. Next, we examine each bit in the binary representation of $R^j(h)$, starting from the most significant bit. Let k be the position of the first bit that equals 1. Finally, we set the parent node of h on the SBT to be

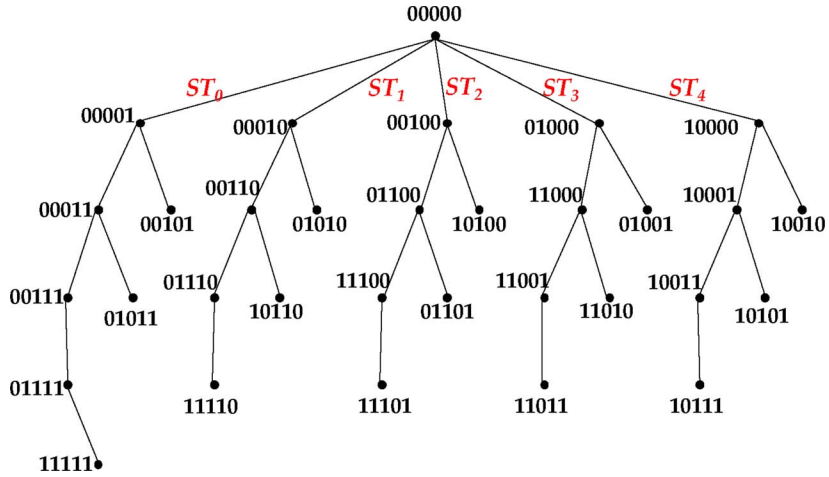

Fig. 7. SBT for node 00000 in the 5-cube (with 32 pods).

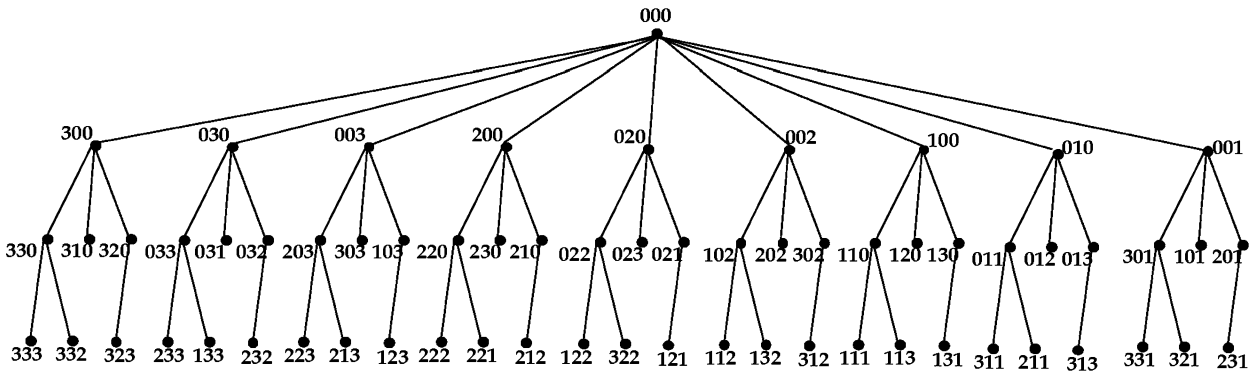

Fig. 8. SBT for node 000 in $\mathrm{GHC}_{3,4}$ (with 64 pods).

$(a_{n-1}a_{n-2}\ldots\overline{a_k}\ldots a_1a_0)$. This process is then repeated until the entire path from s to h is constructed (in a reverse order).

Figs. 7 and 8 show that in a 5-cube and $\mathrm{GHC}_{3,4}$, we can route all the HCs from node 0 (i.e., 00000 in 5-cube, 000 in $\mathrm{GHC}_{3,4}$) to all its destinations in a ST using their shortest paths dictated by the ST itself. In addition, we can reuse the same wavelengths in different STs as they do not share any common link in the 5-cube/$\mathrm{GHC}_{3,4}$. In the 5-cube, we can see that subtree ST_0 is the largest and therefore, may use the maximum number of wavelengths.

It is worth noting that despite Fig. 7, for an arbitrary n-cube, it is *not* always possible to construct a SBT such that each ST contains at most {ceiling of $(N-1)/n$} destinations. Let the maximum number of destination nodes contained in a ST (or the maximum number of HCs) be denoted by $\max(n)$, which is lower bounded by ceiling of $(N-1)/n$, the number of wavelengths required by a SBT, represented as W_{SBT}, can be calculated by (5), where H was defined in Section II-A

$$W_{\mathrm{SBT}} = \left\lceil {}^{\max(n)}/_{H} \right\rceil. \tag{5}$$

Considering the symmetric properties of hypercube again, we can see that the HCs originating from node $h = a_{n-1}a_{n-2}\ldots a_1a_0$ can reuse the same wavelengths as those originating from node $e = \overline{a_{n-1}a_{n-2}\ldots a_1a_0}$. Intuitively, this states that the two "diagonal" nodes in a n-cube can reuse the same wavelengths. We will omit the discussion on GHC due to space limit, but suffice it to say that in a $\mathrm{GHC}_{n,k}$, one can construct SBTs in a similar way and that the k nodes lying on the "diagonal" position/line can reuse the same wavelengths. The total number of wavelengths needed by N SBTs denoted by W_{TTL} is calculated by (6), where $d = 2$ for n-cube, and $d = k$ for $\mathrm{GHC}_{n,k}$, The number of transceivers per pod, denoted by P_{tx}, is given by (7) where $l = n$ for n-cube and $l = n \times (k-1)$ for $\mathrm{GHC}_{n,k}$

$$W_{\mathrm{TTL}} = d^{n-1} \times W_{\mathrm{SBT}} = d^{n-1} \times \left\lceil {}^{\max(n)}/_{H} \right\rceil \tag{6}$$

$$P_{\mathrm{tx}} = l \times \left\lceil {}^{\max(n)}/_{H} \right\rceil. \tag{7}$$

V. Results From Analysis and Simulations

In this section, we first evaluate the complexity of the hypercube-like iDCN using LOBS-HC via numerical analysis, and compare it with the complexity of a datacenter using OCS and electronic switching, respectively. Then we present the traffic performance of LOBS-HC based datacenters via simulation experiments to demonstrate the effectiveness of LOBS-HC based datacenters for various communication patterns and traffic models.

TABLE I
REQUIRED WAVELENGTHS

	Case 1 (C = 100Gbps)	Case 2 (C = 40Gbps)
OCS ring	1275	1275
LOBS-HC ring	150	364
Min. needed	128	319

A. Complexity and Cost Analysis

In this subsection, we compare the minimum number of wavelengths required using LOBS-HC and OCS, respectively, in a datacenter with a hypercube-like interconnection. We also compare the network cost in terms of the number of switches, links/wires and transceivers required using LOBS-HC and electronic packet switching.

The number of wavelengths needed in a LOBS-HC ring using CHA with that needed in an OCS ring with wavelength routing is firstly compared. For OCS, we consider two cases, one with full transient traffic grooming (TTG) at intermediate core nodes, and the other without TTG. Since TTG requires O/E/O conversion in OCS, the case with full TTG is similar to the case with electronic switching in terms of both the number of wavelengths and transceivers needed. In particular, with full TTG (i.e., TTG at every intermediate core node), each pod can multiplex outgoing traffic to different destination pods onto the smallest/minimum number of wavelengths, thus cutting down on the number of wavelengths and transceivers needed at each pod. However, such an approach is more costly than the LOBS-HC ring due to its additional O/E/O transceivers at the core switches.

Table I compares the number of wavelengths needed at each pod in a datacenter consisting of 100 pods, where the sustained interpod bandwidth is 10 Gbps. In such a datacenter, we consider the following two cases: in case 1, $C = 100$ Gbps, $H = 10$ and $G = 10$; and in case 2, $C = 40$ Gbps, $H = 4$ and $G = 25$. The results in Table I shows that the OCS ring (without TTG) requires many more wavelengths (and accordingly transceivers at the pods) than the LOBS-HC ring in both cases. In addition, even with full TTG (where the minimum number of wavelengths is needed), the reduction of the number of wavelengths needed from that needed in LOBS-HC is not significant, implying that it may not justify the increase in the cost of O/E/O at the core switches due to the use of full TTG in OCS or with electronic switching.

Table II compares the number of wavelengths needed in a 64-pod datacenter using either OCS or LOBS-HC when the interconnection topology used is a bidirectional ring, 6-cube, or $\mathrm{GHC}_{3,4}$. Assume that each pod contains 1 024 10 GE servers. And consider again the same two cases as before, the oversubscription ratios in the 6-cube and $\mathrm{GHC}_{3,4}$ are 2 and 8 in case 1, and 1.6 and 6.4 in case 2, respectively, which is below most of the oversubscription ratios in existing datacenters.

Different iDCN will result in different degrees of hardware complexity. In the case of LOBS-HC n-cube, for example, each core switch needs 6 input/output fibers to connect to other core switches and 2 fibers to a pod. Table III compares the complexity of using the proposed LOBS-HC 6-cube and $\mathrm{GHC}_{3,4}$ with that of using the existing 10 Gbps Ethernet switching based on the Fat-tree and BCube topologies, in terms of the total number of

TABLE II
REQUIRED WAVELENGTHS

	Case 1 (C = 100Gbps)	Case 2 (C = 40Gbps)
64-ring with OCS	528	528
64-ring with LOBS-HC	116	144
6-cube with OCS	416	416
6-cube with LOBS-HC	64	128
$\mathrm{GHC}_{3,4}$ with OCS	112	112
$\mathrm{GHC}_{3,4}$ with LOBS-HC	16	32

TABLE III
COST COMPARISON

	L_{TTL}	S_C	N_{tx}
6-cube	320	64	768
$\mathrm{GHC}_{3,4}$	384	64	576
Fat-Tree	8192	128	16384
BCube_3	8192	256	16384

links (denoted by L_{TTL}), core switches (denoted by S_C) and transceivers (denoted by N_{tx}) needed for interpod connections (or at the highest level in the case of BCube) to provide a *comparable* bisection bandwidth.

Their calculations are described as follows. First, L_{TTL} consists of two terms, i.e., $L_{\mathrm{TTL}} = L_C + L_{\mathrm{PC}}$. The first term L_C is the number of links connecting the core switches and is calculated as $n \times 2^{n-1}$ in n-cube and $n \times k^n$ in $\mathrm{GHC}_{n,k}$. The second term, L_{PC}, denotes the number of links connecting the core switch layer and the pod layer and can be calculated as $2^n \times 2$ in n-cube and $k^n \times k$ in $\mathrm{GHC}_{n,k}$ (i.e., the number of pods multiple the number of uplink ports per pod). Then, the required number of core switches S_C, can be obtained by examining the architectures, e.g., those shown in Figs. 5 and 6 for the n-cube and $\mathrm{GHC}_{n,k}$. Finally, the number of required transceivers, N_{tx}, for the n-cube and $\mathrm{GHC}_{n,k}$ is given by (7). Note that even though we allow the Fat-tree and BCube_3 in [1], [2] to have an oversubscription ratio of 8, the results in Table III show the great potentials in reducing the number of wires, switches and transceivers (as all of which contribute to the total cost and power consumption) when using the proposed LOBS-HC implementations. Note that in order to perform a fair cost comparison, we consider datacenters of the same scale, i.e., constructed using different switching technologies or topologies (i.e., Fat-tree, BCube_3, and LOBS-HC 6-cube) but interconnecting the same number of servers with the same amount of bandwidth demands. While it is hard to associate each approach with a single dollar amount for comparison purposes (and such direct comparison in terms of the dollar amount and energy consumption figure is out of scope of this work [25]), we note that even though the electronic switches in Fat-Tree and BCube are indeed less expensive, the total numbers of transceivers which is a key indicator of the total cost and power (and complexity) are more than 20 times higher in Fat-tree and BCube_3 than in LOBS-HC (as shown in Table III).

B. Traffic Performance Evaluation

In this subsection, we evaluate the traffic performance of a 5-cube, 32-pod datacenter network using LOBS-HC through simulation experiments. Each pod is assumed to need a 10 Gbps

HC to each and every other pod as a guaranteed connectivity, and has 5 output fiber links, each consisting of 16 100 Gbps wavelengths [according to (6)]. The routing path of each HC between each S-D pod pair is set up according to the SBT construction introduced earlier (see Fig. 7 in Section IV-B).

There are various communication patterns and traffic models in datacenters. However, only a few papers mentioned their general characteristics [26]–[28]. Here we assume that a large data file is divided into multiple segments/pieces and distributed among multiple servers across different pods. We simulate intradatacenter communication using two typical traffic models: the push model and the pull model. In the push model, we assume that some data processed by and output from the servers within a source pod (hereafter referred to as the *processing pod*) needs to be distributed to and stored in multiple destination pods (called *storing pods*). On the other hand, in the pull model, some input data needed for processing by the servers within a processing pod have to be obtained from other storing pods. In both traffic models, we only focus on interpod traffic and ignore the data (e.g., Ethernet frames) that need to be exchanged among the servers within the same pod.

Since data may be distributed among different pods in various ways, we will consider three representative communication patterns: a uniform distribution, a geometric distribution and a normal distribution. With the uniform distribution, data from a processing pod is evenly distributed among all other pods, in that for any given data frame generated, its destination pod is randomly chosen with an equal probability. With either the geometric or the normal distribution, we aim to capture some degree of communication locality, that is, the tendency that the closer a storing pod is to the processing pod, the higher the probability that the data will be stored in there or accessed from there. More specifically, in our simulation of the geometric distribution, the probabilities that a pod is selected to store or access a given data frame are about 51.6%, 25.8%, 12.9%, 6.45%, and 3.25%, respectively, when the pod are 1 hop, 2 hops, 3 hops, 4 hops, and 5 hops away, respectively, from the processing pod. Similarly, when simulating the normal distribution, we set the address of the storing pod which is numerically next to that of the requesting pod as the mean value (e.g., address 16 is set as the mean value of the processing pod 15), and set the variance as 1.

1) Results From the Push Model: When simulating the push model, it is assumed that each pod will process some data, and as such, it will generate data frames. Different subsets of these data frames will need to be sent to different storing pods according to one of the three communication patterns described above. In particular, the average size of each data frame is generated according to an exponential distribution with a mean value of 4000 bits. The process of generating these data frames for transmission to other storing pods also obeys an exponential distribution with an interval time calculated according to the desired load to be simulated. More specifically, we vary the interval time so the amount of traffic generated varies from anywhere between 0.1 to 1 with respect to the maximum network capacity. After each outgoing data frame is generated, multiple frames with the same destination (storing) pod are assembled into a burst using a minimal-size based algorithm with the threshold being 200 Kbits.

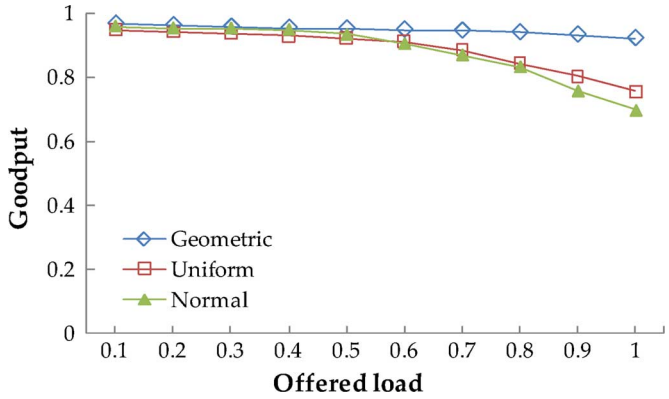

Fig. 9. Goodput under different communication patterns and threshold of queue-length is 0.

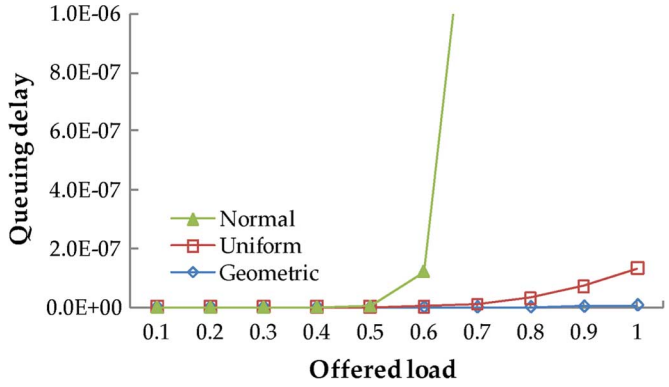

Fig. 10. Queuing delay under different communication patterns and threshold of queue-length is 0.

It is also assumed that at the interface between each processing pod and the LOBS-HC based iDCN, there is one queue for each destination storing pod. A burst will be sent out over a designated HC for that destination pod right away if no other burst is being transmitted over that HC. Otherwise, the burst will be buffered for a while for later transmission as long as the number of transmitted bursts over that HC is still within the guaranteed bandwidth for In-Profile traffic. Instead of requiring a sophisticated traffic metering mechanism, we adopt a simple queue-length based approximate approach to regulate the amount of In-Profile traffic to be transmitted over each HC. More specifically, if the queue length exceeds a certain threshold, then the head-of-line burst will be sent as an Out-of-Profile burst using a nonHC.

We have measured the goodput and queuing delay as a function of the offered load under the three traffic distributions, and the results when the queue-threshold is set to 0 are shown in Figs. 9 and Fig. 10. From the results, we can see that the best performance is achieved under the geometric distribution. That is due to the fact that in the geometric distribution, more than half of the data is transmitted to storing pods that are only 1-hop away. Overall, when the traffic load is below 0.5, the performance in all cases is sufficiently good for datacenter applications. Under geometric distribution, both the goodput and delay achieved are still respectable even at 100% load. However, under

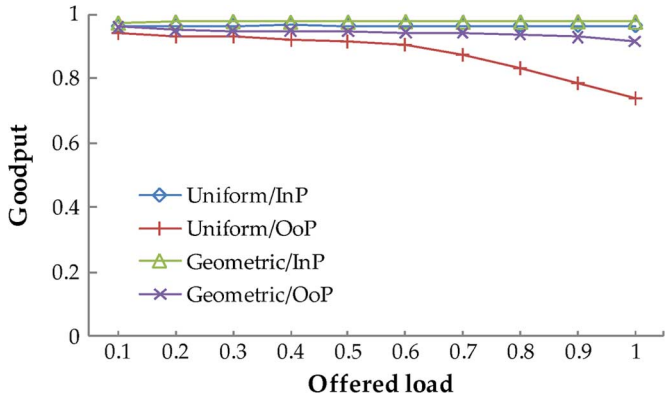

Fig. 11. Goodput for In-Profile and Out-of-Profile traffic and threshold of queue-length is 0.

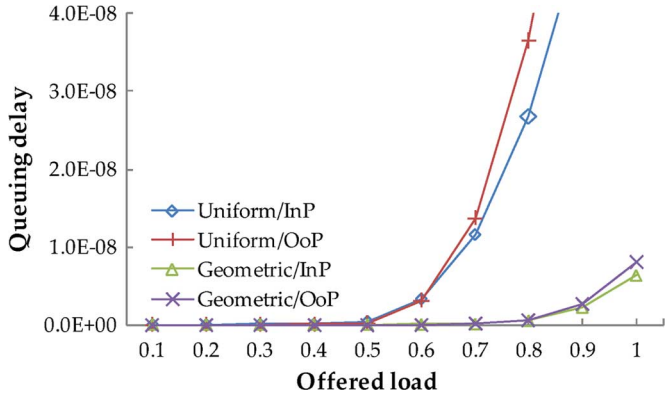

Fig. 12. Queuing delay for In-Profile and Out-of-Profile traffic and threshold of queue-length is 0.

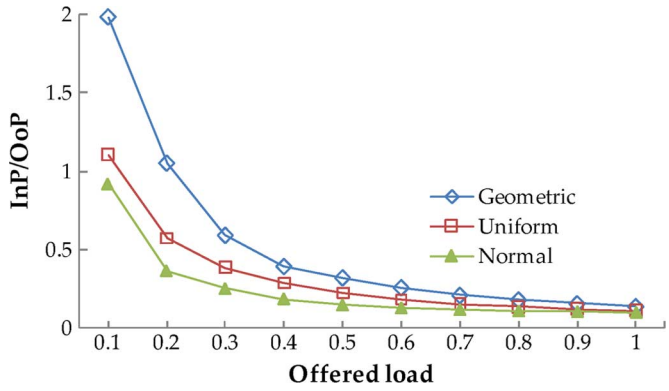

Fig. 13. Ratio of In-Profile traffic to Out-of-Profile traffic under different communication patterns and threshold of queue-length is 0.

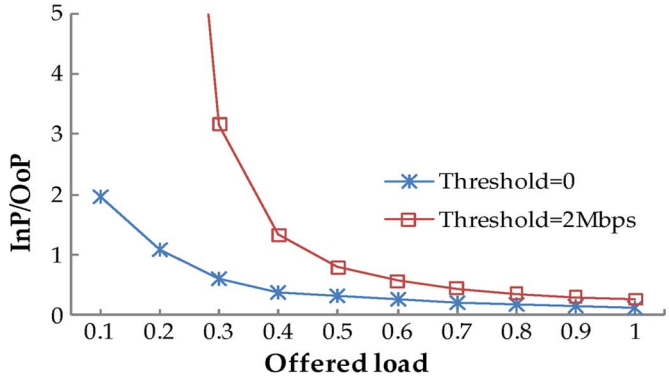

Fig. 14. Ratio of In-Profile traffic to Out-of-Profile traffic with different queue threshold lengths under Geometric distribution.

the normal distribution, the delay performance is poor when the load exceeds 50%. This is mainly due to the fact that, under the normal distribution, more data is stored at pods whose addresses are numerically next to that of the requesting pod but who are physically far away (e.g., pod 16 is 5 hops away from pod 15 in the 5-cube).

To obtain insights into LOBS-HC's unique capability of opportunistically transmitting overflowing data as Out-of-Profile bursts using non-HCs, we have also measured and compared the performance of the In-Profile (InP) and Out-of-Profile (OoP) traffic under the uniform and geometric distributions with the queue-length threshold varying from 0 to 2 Mbits. The results shown in Figs. 11 and 12 are obtained by setting the queue-length threshold to 0. That is, when a new burst is generated and the HC is not available (i.e., busying in transmitting a previous burst), the new burst will be sent out as Out-of-Profile traffic immediately in an opportunity fashion.

As can be seen and expected, the In-Profile traffic under both uniform and geometric distributions performs well, and better than the Out-of-Profile traffic in terms of both goodput and queuing delay. In particular, the goodput of In-Profile traffic under both traffic distributions are very close to 1. However, the queuing delay of In-Profile traffic is not that much smaller than that of the Out-of-Profile traffic since In-Profile bursts need to be buffered in the same queue as Out-of-Profile bursts. The amount of traffic sent as In-Profile versus that sent as Out-of-Profile is also studied in Fig. 13. Fig. 14 shows that, under the geometric distribution, when the queue-length threshold is set to 2 Mbits, the ratio of traffic sent as In-Profile traffic to that sent as Out-of-Profile traffic is higher than that when the queue-length threshold is zero. In particular, at a low load (e.g., around 0.2), all traffic is sent as In-Profile traffic with a threshold equal to 2 Mbits whereas about half of the total traffic is sent as In-Profile traffic when the threshold is zero. However, in both cases, the ratio decreases with network traffic load. This is because as the network load increases, more and more traffic has to be sent as Out-of-Profile traffic given the fixed and limited HC capacity. Note that the fact that there is a healthy amount of Out-of-Profile traffic under a high traffic load implies that LOBS-HC can be more effective than OCS, since OCS would have not been able to send these Out-of-Profile traffic using someone else's circuit at all (without O/E/O at intermediate nodes).

2) Results From the Pull Model: When simulating the pull model, it is assumed that each file has already been partitioned into a certain number of pieces, and these pieces are stored at storing pods according to one of the three distributions mentioned above relative to the requesting pod. In addition, (the servers within) each pod may randomly generate a request for a data file needed for processing according to exponential distribution. For each request, all the pieces of the requested data file will have to be fetched from their storing pods. Accordingly,

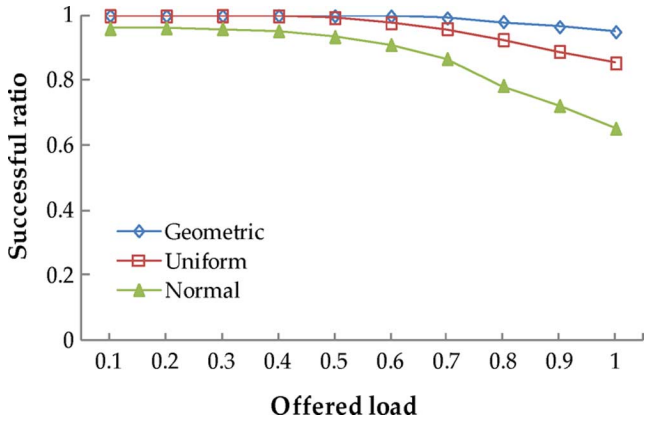

Fig. 15. Successful ratio of fetched files under different communication patterns.

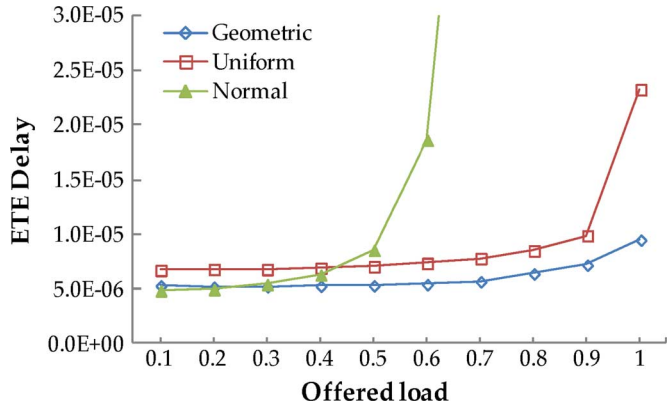

Fig. 16. ETE delay of fetched files under different communication patterns.

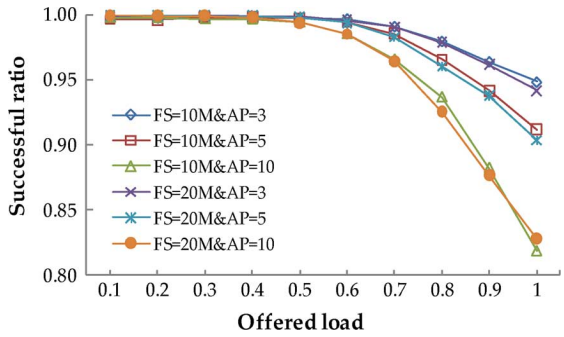

Fig. 17. Successful ratio when having different file sizes and different number of file pieces under geometric distribution.

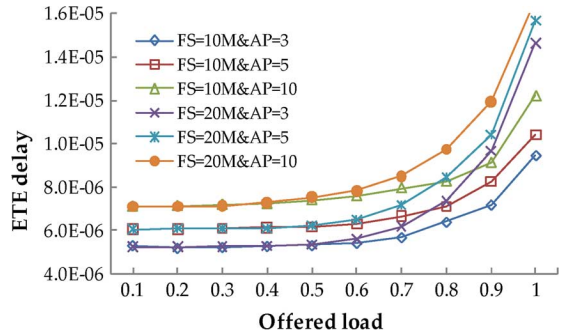

Fig. 18. ETE delay when having different file sizes and different number of file pieces under geometric distribution.

different from the push model, here, a performance metric of interest is the time it would take for the processing (or requesting) pod to obtain all the pieces of the file needed for processing. Since not all files can be fetched in their entity within a given simulation time, we are also interested in the ratio of the number of files completely fetched to that only partially fetched.

In this set of simulations, we assume that by default, the average file size is 10 Mbits, and each of them is stored at an average of three other pods (besides the requesting pod itself). In addition, we will omit the performance of In-Profile and Out-of-Profile traffic, and simply assume that each file piece will be sent out as an Out-of-Profile burst when its own HC is not available (this is equivalent to setting the queue length threshold to zero). The successful ratio of fetching files (in their entity), and the end-to-end (ETE) delay in fetching these files which consists of the queuing delay, transmission delay and propagation delay are evaluated as functions of the offered load.

From the results shown in Figs. 15 and 16, we can see that both the successful ratio and ETE delay under the geometric distribution are better than those under the normal and uniform distributions. This is similar to the results from the Push model. However, note that in the normal case, the successful ratio decreases significantly as the offered load increases even though, as expected from the results from the Push model, the absolute number of successfully fetched file pieces does not decrease that fast with the traffic load. The main reason is that as long as the requesting pod has not yet received one piece of the file from a far-away storing pod, the file is considered not successfully fetched, and under the normal distribution, the chance that a file piece is stored in a far-away storing pod is much higher than the other two distributions.

The effect of having different file sizes and different number of pieces per file on the traffic performance of LOBS-HC based iDCN have also been investigated. Figs. 17 and 18 show the results when the average file sizes (FSs) are 10 Mbits and 20 Mbits, respectively, and each file is stored at an average of 3, 5, and 10 other pods (besides the requesting pod itself). These results, obtained under the geometric distribution only, show that whether $\mathrm{FS} = 10$ Mbits or 20 Mbits, as long as the average number of pieces (APs) is the same, the successful ratios will be close. However, the successful ratio will decrease as the number of pieces increases since it becomes more difficult to fetch all pieces from all storing pods in a given simulation time. In other words, the successful ratio is not affected by the file size but rather by the number of pieces each file is divided into. On the other hand, the ETE delay will increase either with the file size or the number of pieces per file as shown in Fig. 15, which is expected since it will take a long time to completely fetch a larger file or more pieces per file.

Finally, we compare the traffic performance in iDCN using LOBS-HC and OBS respectively as shown in Figs. 19 and 20 under the geometric and uniform distributions. As can be seen, under the geometric distribution, the successful ratio of LOBS-HC and OBS are quite similar as most of the pieces of a file are stored at pods that are only one hop away from the pod requesting for the file. However, under the normal distribution,

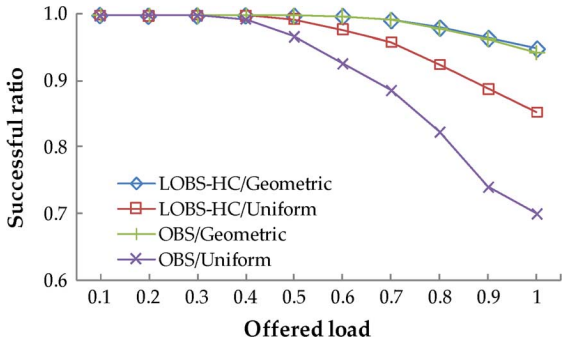

Fig. 19. Successful ratio of fetched files for iDCN using LOBS-HC and OBS, respectively.

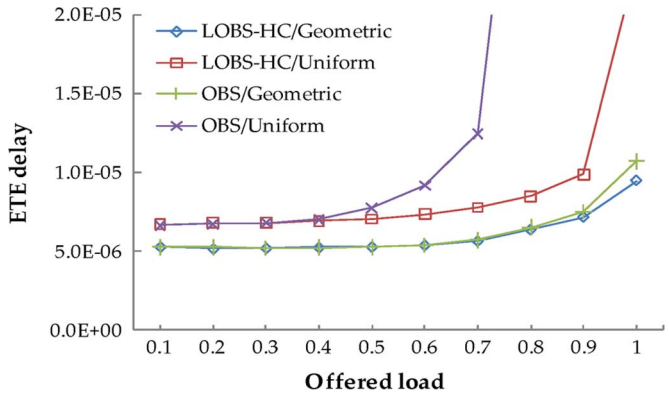

Fig. 20. ETE delay of fetched files for iDCN using LOBS-HC and OBS, respectively.

LOBS-HC performs much better than OBS when the traffic load is high. This is because in LOBS-HC, a requesting pod has a much better chance to receive the pieces that are stored far away from it than in OBS, thanks to the HC between the storing pod and the requesting pod which provides some guaranteed bandwidth (whereas in OBS, no such bandwidth guarantee exist between a far-way storing pod and the requesting pod).

Note that, although not shown, we have also compared LOBS-HC with OBS under the Push model. Our simulation results have revealed that LOBS-HC performs slightly better than OBS under a high load under both the geometric and uniform distribution. The reason for LOBS-HC to be only slightly better than OBS under the Push model is that the goodput and ETE delay do not distinguish one data frame from another as long as both are delivered, and in OBS, although it cannot deliver as many data frames between requesting pod and far-away storing pods as LOBS-HC, it can make up for some loss of goodput and ETE delay performance by delivering a few more data frames between requesting and storing pods that are close to each other.

We have not compared the traffic performance of LOBS-HC with that of electrical switching or hybrid switching since a fair comparison is extremely difficult, if not impossible. This is mainly due to the fact that these approaches use different amount of resources (e.g., number of wavelengths, transceivers) as discussed earlier. We refer interested readers to related works in [23], [24] that compared wide-area network traffic performance of OBS (and its variation including LOBS-HC) and OCS and discussed the challenges in achieving a fair comparison. In addition, although LOBS-HC is applicable to any topology, we have not applied LOBS-HC to Fat-tree or BCube and plan to explore these and other topologies as a future work.

C. Performance Comparison Between Fat-Tree and 5-Cube

In this subsection, we compare the performance between an electronic Fat-Tree and a LOBS-HC based cube in terms of the queuing delay and throughput.

Note that there is no direct quantitative performance comparison between different iDCNs, let alone between an electronic iDCN and an optical iDCN since it is difficult, if not impossible to do so due to the lack of consensus and data on how to characterize the workloads and traffic patterns etc.

Here, we try to make a simulation-based comparison as fair as feasible Specifically, we consider an electronic Fat-Tree with 32 pods that can interconnect a total of 8192 servers ($= 32^3/4$ in [1]). Note that with a full interconnection, the number of links between core switches and the aggregated switches reaches up to 8192 ($= 256 \times 32$ in [1]), which is too time consuming to simulate. Accordingly we assume an oversubscription ratio of 8 in the Fat-Tree so there is only a total of 1024 ($= 32 \times 32$) links between core switches and aggregate switches. For a fair comparison, we also consider a 5-cube LOBS-HC datacenter with 32 pods. Each pod has 5 bidirectional output fiber links, each consisting of 16 100 Gbps wavelengths [according to (6)].

We will limit the comparison study to the Push model using the same set of default parameters about the traffic arrival/generation model and burst assembly algorithms, and routing used in the LOBS-HC cube. For the Fat-tree, we assume that the queue size at each port of a switch is 1000 packet. In addition, to route in the electronic fat-tree, a source pod simply selects a core switch, and then go from the same core switch to the destination pod.

We have measured the throughput (or goodput) and queuing delay as a function of the offered load under the two traffic distributions, and the results are shown in Figs. 21 and 22. From Fig. 21, we can observe that: 1) the queuing delays for electronic Fat-Tree under both uniform and geometric distributions are very similar to (almost overlap with) each other; and 2) the queuing delays of Fat-Tree under both distributions are much higher than that of LOBS-HC based 5-cube. The first observation is due to the fact that every two pods are two hops away in the Fat-Tree architecture, i.e., each packet traverse two hops no matter how the destination pods are distributed. The second observation is due to the store-and-forwarding packet processing in electronic Fat-Tree results in packet buffering time at intermediate nodes, which is absent from the all-optical LOBS-HC 5-cube.

From Fig. 22, we can observe two similar facts that: 1) the throughput of Fat-Tree under both distributions are similar to each other except when the offered load gets high (near to 1); and 2) the throughput of the 5-cube under geometric distribution is the highest, followed by that under the two distributions in Fat-Tree, and that under the uniform distribution in the 5-cube. To explain the second observation, we note that the throughput decreases in the LOBS-HC 5-cube mainly due to burst loss/contention, whereas throughput decreases in Fat-Tree mainly due to

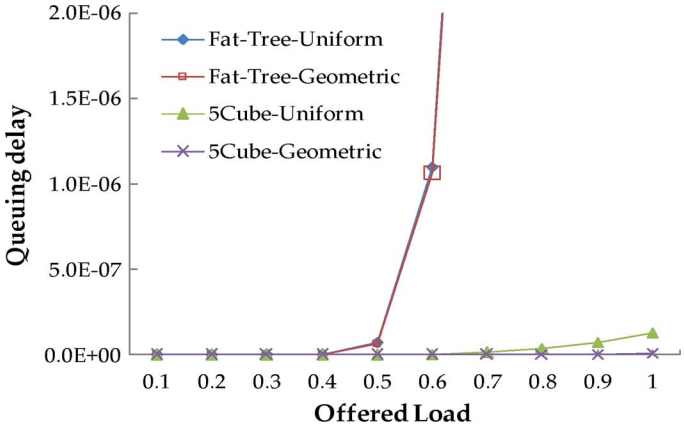

Fig. 21. Queuing delay under different distributions for Fat-Tree and 5-Cube.

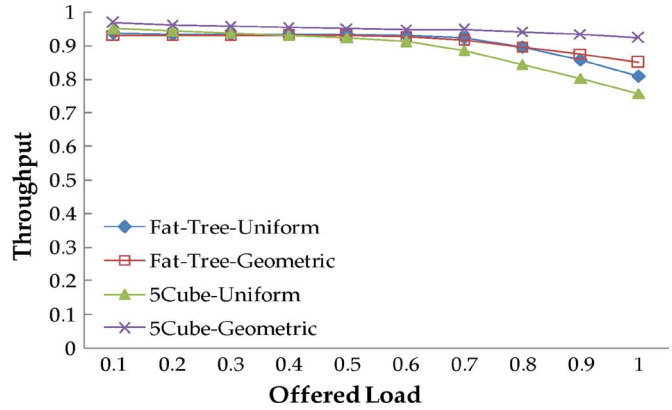

Fig. 22. Throughput under different distributions for Fat-Tree and 5-Cube.

increased queuing delay at intermediate nodes. The geometric distribution in LOBS-HC performs the best since more than 50% of the packets (bursts) are transmitted to storing pods that are only 1-hop away (with no contention), and only a small percentage of packets will go through multiple hops and thus encounter very little contention and loss, plus they do not suffer from any queuing delay at the intermediate nodes. However, the uniform distribution in LOBS-HC performs the worst because about 60% of packets are transmitted to storing pods that are more than 3 hops away in which case, the damage caused by contention and loss in LOBS-HC out weights the benefits of avoiding the queuing delay in the Fat-Tree.

VI. Concluding Remarks

In this paper, we have proposed a novel design of iDCN with optical switching based on LOBS-HC. We have developed efficient HC routing, grouping and wavelength assignment algorithms for hypercube-like interconnection topologies which are used to interconnect the pods, and discussed several key unique features of the design including the ability to guarantee bandwidth for In-Profile traffic among all pods using HCs, and support opportunistic transmissions of Out-of-Profile traffic. We have evaluated the complexity and traffic performance of iDCN using LOBS-HC numerically and experimentally (via simulations). The numerical results have shown that the proposed iDCN using LOBS-HC requires much fewer wavelengths than their optical circuit switching counterpart, and requires much fewer wires and transceivers for providing the full bisection bandwidth than their electronic packet switching counterpart using Fat-tree or BCube. The experimental results have shown acceptable traffic performance under various communication patterns and traffic model of iDCN using LOBS-HC.

References

[1] M. A. Fares, A. Loukissas, and A. Vahdat, "A scalable, commodity data center network architecture," presented at the SIGCOMM, Seattle, WA, Aug. 2008.
[2] C. X. Guo *et al.*, "BCube: A high performance, server-centric network architecture for modular data centers," presented at the SIGCOMM, Barcelona, Spain, Aug. 2009.
[3] C. X. Guo *et al.*, "DCell: A scalable and fault-tolerant network structure for data centers," presented at the SIGCOMM, Seattle, WA, Aug. 2008.
[4] N. Farrington *et al.*, "Helios: A hybrid electrical/optical switch architecture for modular data centers," presented at the SIGCOMM, New Delhi, India, Sep. 2010.
[5] G. Wang *et al.*, "c-Through: Part time optics in data centers," presented at the SIGCOMM, New Delhi, India, Sep. 2010.
[6] X. H. Ye *et al.*, "DOS-A scalable optical switch for datacenters," presented at the ANCS, La Jolla, CA, Oct. 2010.
[7] K. Xi, Y. H. Kao, M. Yang, and H. J. Chao, Petabit Optical Switch for Data Center Networks [Online]. Available: http://eeweb.poly.edu/chao/publications/TechReports.html
[8] A. Singla, A. Singh, K. Ramachandran, L. Xu, and Y. Zhang, "Proteus: A topology malleable data center networks," presented at the HotNets Monterey, CA, Oct. 2010.
[9] C. Minkenberg *et al.*, "Designing a crossbar scheduler for HPC applications," *IEEE Micro*, vol. 26, pp. 58–71, 2006.
[10] R. Hemenway, R. R. Grzybowski, C. Minkenberg, and R. Luijten, "Optical-packet-switched interconnect for supercomputer applications," *J. Optical Netw.*, vol. 3, no. 12, pp. 900–913, 2004.
[11] O. Liboiron-Ladouceur *et al.*, "The data vortex optical packet switched interconnection network," *J. Lightw. Technol.*, vol. 26, no. 13, Jul. 2008.
[12] T. Tanemura, A. A. Amin, and Y. Nakano, "Multihop field trial of optical burst switching testbed with PLZT optical matrix switch," *IEEE Photon. Technol. Lett.*, vol. 21, no. 1, pp. 42–44, Jan. 2009.
[13] A. A. Amin *et al.*, "Optical burst switching with burst collision resolution using a fast 4 × 4 PLZT switch," *IEICE Electron. Exp.*, vol. 3, no. 23, pp. 504–508, 2006.
[14] D. Banerjee and B. Mukherjee, "Wavelength-routed optical networks: Linear formulation, resource budgeting tradeoffs, and a reconfiguration study," *IEEE/ACM Trans. Netw.*, vol. 8, pp. 598–607, 2000.
[15] H. Zang, J. P. Jue, and B. Mukherjee, "A review of routing and wavelength assignment approaches for wavelength-routed optical WDM networks," *Optical Netw. Mag.*, vol. 1, no. 1, pp. 47–60, 2000.
[16] F. Xue, S. J. B. Yoo, H. Yokoyama, and Y. Horiuchi, "Performance comparison of optical burst and circuit switched networks," presented at the OFC/NFOEC, Anaheim, CA, Mar. 2005.
[17] X. Liu, C. Qiao, X. Yu, and W. Gong, "A fair packet-level performance comparison of OBS and OCS," presented at the OFC/NFOEC'2006, Anaheim, CA, Mar. 2006.
[18] M. G. Ortega, C. Qiao, A. S. Gonalez, X. Liu, and J. L. Ardao, "LOBS-H: An enhanced OBS with wavelength sharable home circuits," presented at the ICC, Capetown, South Africa, May 2010.
[19] M. G. Ortega *et al.*, "Evaluation of labeled OBS with home circuits," presented at the ICCCN, Zurich, Swtizerland, Aug. 2010.
[20] X. Zhang and C. Qiao, "On scheduling all-to-all personalized connections and cost-effective designs in WDM rings," *IEEE/ACM Trans. Network.*, vol. 7, no. 3, pp. 435–445, 1999.
[21] C. T. Ho and S. L. Johnson, "Spanning balanced trees in boolean cubes," *J. Sci. Statist.Comput.*, vol. 10, no. 4, pp. 607–630, 1989.
[22] S. G. Ziavras and S. Krishnamurthy, "Evaluating the communications capabilities of the generalized hypercube interconnection networks," *Concurrency: Pract. Exp.*, vol. 11, no. 6, pp. 281–300, 1999.
[23] X. Liu, C. Qiao, X. Yu, and W. Gong, "A fair packet-level performance comparison of OBS and OCS," presented at the OFC/NFOEC, Anaheim, CA, Mar. 2006.
[24] C. Qiao, M. G. Ortega, A. S. Gonzalez, X. Liu, and J. L. Ardao, "On the benefit of fast switching in optical networks," presented at the OFC/NFOEC, San Diego, CA, Mar. 2010.

[25] A. Greenberg, J. Hamilton, D. A. Maltz, and P. Patel, "The cost of a cloud: Research problems in data center networks," *Comput. Commun. Rev.*, vol. 39, no. 1, 2009.
[26] S. Kandula, S. Sengupta, A. Greenberg, P. Patel, and R. Chaiken, "The nature of data center traffic: Measurements& analysis," presented at the IMC, IL, Nov. 2009.
[27] T. A. Benson, A. Anand, A. Akella, and M. Zhang, "Understanding data center traffic characteristics," presented at the WREN2009, Barcelona, Spain, Aug. 2009.
[28] S. James and P. Crowley, "Fast content distribution on datacenter networks," presented at the ANCS, Brooklyn, NY, Oct. 2011.
[29] J. Kim, W. J. Dally, and D. Abts, "Flattened butterfly: A cost-efficient topology for high-radix networks," presented at the ISCA, San Diego, CA, Jun. 2007.
[30] L. M. Peng *et al.*, "A novel approach to optically switching inter-pod traffic in datacenters," presented at the CC'2011, Shanghai, China, Apr. 2011.
[31] L. M. Peng, C. Qiao, W. Tang, and C. H. Youn, "Cube-based intra-datacenter networks with LOBS-HC," presented at the ICC, Kyoto, Japan, Jun. 2011.

Limei Peng received the B.Sc. degree from the South Central University for Nationalities, Wuhan, China, in 2004, and the M.Sc. and Ph.D. degrees from the Chonbuk National University in Jeonju, Chonbuk, South Korea, in 2006 and 2010, respectively.

She has worked as a Postdoctorate Fellow in Grid Middleware Research Center, Korea Advanced Institute of Science and Technology (KAIST), South Korea. She is currently an Associate Professor at the School of Electronic and Information Engineering, SooChow University, China. Her research interests include optical communication networks and protocols, datacenter networks, optical fiber sensor networks, and Cloud computing networks.

Chan-Hyun Youn (S'84–M'87) received the B.Sc. and M.Sc. degrees in electronics engineering from Kyungpook National University, Taegu, Korea, in 1981 and 1985, respectively. He also received the Ph.D. in electrical and communications engineering from Tohoku University, Japan, in 1994.

He served at Korean Army as a communications officer, First Lieutenant, from 1981 to 1983. Before joining the University, from 1986 to 1997, he was the leader of high-speed networking team at Korea Telecom (KT) Telecommunications Network Research Laboratories, where he had been involved in the research and developments of Centralized Switching Maintenance System (CSMS), Maintenance and Operation system for Various ESS's (MOVE) system, high-speed networking, and HAN/B-ISDN network testbed. Especially, he was a principal investigator of high-speed networking projects including ATM technical trial between KT and KDD, Japan, Asia-Pacific Information Infrastructure (APII) testbed, Korea Research and Education Network (KOREN) and Asia-Pacific Advanced Network (APAN), respectively. Since 2009, he has been a Professor at KAIST, Daejeon, Korea. He also was a Dean of Office of Planning Affairs and a Director of Research and Industrial Cooperation Group at former Information and Communications University, in 2006 and 2007. He was a Visiting Professor at MIT in 2003 and has been engaged in the development of Physio-Grid system with Prof. R. G. Mark's group in LCP (Laboratory for Computational Physiology) of MIT since 2002. He also is a Director of Grid Middleware Research Center at KAIST. Where, he is developing core technologies that are in areas of advanced mobile cloud system, cloud collaboration system, workflow management system, genome workflow management and others.

Dr. Youn is serving as Editor-in-Chief in KIPS (Korea Information Processing Society), an Editor of *Journal of Healthcare Engineering* (U.K.), and served as a Head of Korea branch (computer section) of IEICE, Japan (2009, 2010).

Wan Tang received the B.Sc. and M.Sc. degrees in computer application technology from South Central University for Nationalities, Wuhan, China, in 1995 and 2001, respectively, and received the Ph.D. degree in communication and information system from Wuhan University, China in 2009.

She is currently an Associate Professor in the College of Computer Science of South-Central University for Nationalities. Also, from 2001 to 2002, she worked as a Visiting Researcher at the Advanced Communications and Networks Laboratory at Chonbuk National University, Jeonju, South Korea. Her research interests include protocols for optical/wireless communication networks and computational intelligence.

Chunming Qiao (S'89–M'92–SM'04–F'10) received the B.Sc. degree from University of Science and Technology, China, in 1985, and the Ph.D. degree from University of Pittsburgh, Pittsburgh, PA, in 1993.

He is currently the Director for the Lab for Advanced Network Design, Analysis, and Research (LANDER), which conducts cutting-edge research with current foci on optical networking and survivability issues in cloud computing, human-factors and mobility in wireless networks, low-cost and low-power sensors, and mobile sensor networking. He has published more than 95 and 150 papers in leading technical journals and conference proceedings, respectively. His pioneering research on Optical Internet in mid 1990, in particular, the optical burst switching (OBS) paradigm has produced some of the highest cited works. In addition, his work on integrated cellular and ad hoc relaying systems (iCAR), started in 1999, is recognized as the harbinger for today's push towards the convergence between heterogeneous wireless technologies, and has been featured in Business Week and Wireless Europe, as well as at the websites of New Scientists and CBC. His Research has been funded by nine NSF grants as a PI including two ITR awards, and by seven major telecommunications companies, as well as Industrial Technology Research Institute (in Taiwan).

Dr. Qiao has given a dozen of keynotes, and numerous invited talks on the above research topics. He has chaired and cochaired a dozen of international conferences and workshops. He has been an Editor of a couple of IEEE TRANSACTIONS and a Guest Editor for several IEEE JOURNAL ON SELECTED AREAS IN COMMUNICATIONS (JSAC) issues. He was the Chair of the IEEE Technical Committee on High Speed Networks (HSN) and currently chairs the IEEE Subcommittee on Integrated Fiber and Wireless Technologies (FiWi), which he founded.