NORTHWESTERN
UNIVERSITY

Electrical Engineering and Computer Science Department

Technical Report
NWU-EECS-12-01
March, 2012

OSA: An Optical Switching Architecture for Data Center Networks with Unprecedented Flexibility

Kai Chen, Ankit Singla, Atul Singh, Kishore Ramachandran,
Lei Xu, Yueping Zhang, Xitao Wen, Yan Chen

Abstract

Data center networks (DCNs) form the backbone infrastructure of many large-scale enterprise applications as well as emerging cloud computing providers. This paper describes the design, implementation and evaluation of OSA, a novel Optical Switching Architecture for DCNs. Leveraging runtime reconfigurable optical devices, OSA dynamically changes its topology and link capacities, thereby achieving unprecedented flexibility to adapt to dynamic traffic patterns. Extensive analytical simulations using both real and synthetic traffic patterns demonstrate that OSA can deliver high bisection bandwidth (60%-100% of the non-blocking architecture). Implementation and evaluation of a small-scale functional prototype further demonstrate the feasibility of OSA.

Keywords: Data center networks, optical networking technology, switching architecture

OSA: An Optical Switching Architecture for Data Center Networks with Unprecedented Flexibility

Kai Chen*, Ankit Singla†, Atul Singh‡, Kishore Ramachandran‡, Lei Xu‡,
Yueping Zhang‡, Xitao Wen*, Yan Chen*
*Northwestern University, †University of Illinois at Urbana-Champaign, ‡NEC Labs America, Inc.
*{kch670,xwe334}@eecs.northwestern.edu, †singla2@illinois.edu,
‡{atuls,kishore,leixu,yueping}@nec-labs.com, *ychen@northwestern.edu

Abstract—**Data center networks (DCNs) form the backbone infrastructure of many large-scale enterprise applications as well as emerging cloud computing providers. This paper describes the design, implementation and evaluation of OSA, a novel Optical Switching Architecture for DCNs. Leveraging runtime reconfigurable optical devices, OSA dynamically changes its topology and link capacities, thereby achieving unprecedented flexibility to adapt to dynamic traffic patterns. Extensive analytical simulations using both real and synthetic traffic patterns demonstrate that OSA can deliver high bisection bandwidth (60%-100% of the non-blocking architecture). Implementation and evaluation of a small-scale functional prototype further demonstrate the feasibility of OSA.**

Index Terms—**Data center networks, Optical networking technology, Switching architecture**

I. Introduction

Many on-line services, such as those offered by Amazon, Google, FaceBook, and eBay, are powered by massive data centers hosting hundreds of thousands of servers. The network interconnect of the data center plays a key role in the performance and scalability of these services. As the number of hosted applications and the amount of traffic grow, the industry is looking for larger server-pools, higher bit-rate network interconnects, and smarter workload placement approaches to satisfy the demand. To meet these goals, a careful examination of traffic characteristics, operator requirements, and network technology trends is critical.

Traffic characteristics. Several recent DCN proposals attempt to provide uniformly high capacity between all servers [1]–[4]. Given that it is not known *a priori* which servers will require high speed connectivity, for a static, electrical network, this appears to be the only way to prevent localized bottlenecks. However, for many real scenarios, such a network may not be fully utilized at all times. For instance, measurement on a 1500-server Microsoft production DCN reveals that only a few ToRs are hot and most of their traffic goes to a few other ToRs [5]. Likewise, an analysis of high-performance computing applications shows that the bulk of interprocessor traffic is degree-bounded and slowly-changing [6]. Thus, even for a few thousand servers, uniformly high capacity networks appear to be an overkill. As the size of the network grows, this weighs on the cost, power consumption and complexity of such networks.

Dealing with the oversubscribed networks. Achieving high performance for data center services is challenging in the oversubscribed networks. One approach is to use intelligent workload placement algorithms to allocate network-bound service components to physical hosts with high bandwidth connectivity [7], *e.g.*, placing these components on the same rack. Such workloads exist in practice: dynamic creation and deletion of VM instances in Amazon's EC2 or periodic backup services running between an EC2 (compute) instance and an S3 (storage) bucket. An alternate approach is to flexibly allocate more network bandwidth to service components with heavy communications. If the network could "shape-shift" in such fashion, this could considerably simplify the workload placement problem.

Higher bit-rates. There is an increasing trend towards deploying 10 GigE NICs at the end hosts. In fact, Google already has 10 GigE deployments and is pushing the industry for 40/100 GigE [8]–[10]. Deploying servers with 10 GigE naturally requires much higher capacity at the aggregation layers of the network. Unfortunately, traditional copper-wire 10 GigE links are not viable for distances over 10 meters [11] due to their high power budget and larger cable size, necessitating the need to look for alternative technologies.

Optical networking technology is well suited to meet the above challenges. Optical network elements support on-demand provisioning of connectivity and capacity where required in the network, thus permitting the construction of thin, but flexible interconnects for large server pools. Optical links can support higher bit-rates over longer distances using less power than copper cables. Moreover, optical switches run cooler than electrical ones [12], resulting in lower heat dissipation and cheaper cooling cost. The long-term advantage of optics in DCNs has been noted in the industry [12, 13].

Recent efforts in c-Through [14] and Helios [11] provide a promising direction to exploit optical networking technology (*e.g.*, one-hop high-capacity optical circuits) for building DCNs. Following this trailblazing research, we present OSA, a novel Optical Switching Architecture for DCNs. OSA achieves high flexibility by leveraging and extending the techniques devised by previous work, and further combining them with novel techniques of its own. Similar to the previous work, OSA leverages reconfigurability of optical devices to dynamically set up one-hop optical circuits. Then, OSA employs the novel hop-by-hop stitching of multiple optical links to provide all-to-all connectivity for mice flows and bursty communications, and also to handle workloads involving high fan-in/out hotspots [15] that the existing one-hop electrical/optical architectures cannot address efficiently via their optical interconnects. Further, OSA dynamically adjusts the capacities on the optical links to satisfy changing traffic demand at a finer granularity. Additionally, to make efficient use of expensive optical ports, OSA introduces Circulator (Sec. II-B), a bi-directionality-enabling component for simultaneous transmission in both directions over the same circuit, which potentially doubles the usage of MEMS ports.

Overall, the highlights of this paper are as follows.

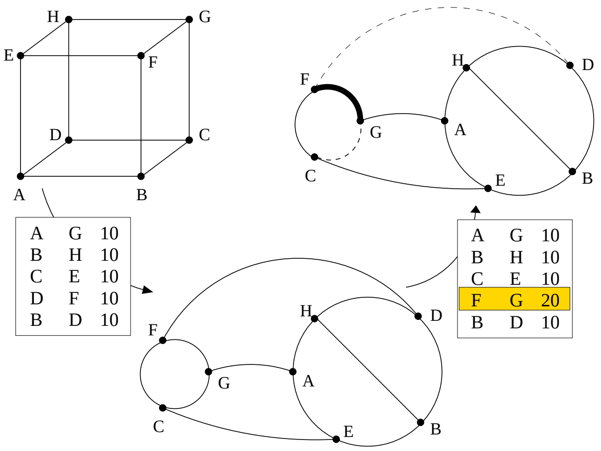
A flexible DCN architecture. Given a number N of Top-of-Rack (ToR) switches and a design-time-fixed parameter k, OSA can assume *any* k-regular topology over the N ToRs. To illustrate how many options this gives us, consider that for just N=20, there are over 12 billion (non-isomorphic) connected 4-regular graphs [16]. In addition, OSA allows the capacity of each edge in this k-regular topology to be varied from a few Gb/s to a few hundred Gb/s on-demand. Evaluation results in Sec. V-B2 suggest an up to 150% and 50% performance improvement brought by flexible topology and flexible link capacity, respectively.

An analysis of OSA-2560. We evaluate a particular instance of container-size OSA architecture, OSA-2560 (N=80, k=4), with 2560 servers via extensive simulations and analysis. Our evaluation results (Sec. V-B) suggest that OSA-2560 can deliver high bisection bandwidth that is 60%-100% of the non-blocking network and outperform the hybrid structures by 80%-250% for both real and synthetic traffic patterns. Our analysis (Sec.III-C) shows that OSA incurs lower cost ($\sim$38%), lower ($\sim$37%) power consumption, and one order of magnitude simpler cabling complexity compared to a non-blocking Fattree [1] connecting a similar number of servers. Furthermore, compared with the hybrid structures, OSA has similar cost but consumes slightly less power. We believe that for data centers that expect skewed traffic demands, OSA provides a compelling tradeoff between cost, complexity and performance.

An implementation of OSA prototype. We build a small-scale 8-rack OSA prototype with real optical devices. Through this testbed, we evaluate the performance of OSA with all software and hardware overheads. Our results show that OSA can quickly adapt the topology and link capacities to meet the changing traffic patterns, and that it achieves nearly 60% of non-blocking bandwidth in the all-to-all communication. We further examine the impact of OSA design on bulk data transfer and mice flows, and find that the overhead introduced by hop-by-hop routing on mice flows is small: a 2 ms additional latency for a 7-hop routing with full background traffic. We also measure the device characteristics of the optical equipment, evaluate the impact of multi-hop optical-electrical-optical (O-E-O) conversion, and discuss our experience building and evaluating the OSA prototype.

Limitations. OSA, in its current form, has limitations. Small flows, especially those latency-sensitive ones, may incur non-trivial penalty due to reconfiguration delays ($\sim$10 ms). While the fraction of such affected flows is small (Sec. VII), we propose multiple avenues to solve this challenge. The second challenge is how to scale OSA from a container-size to a larger date center consisting of tens to hundreds of thousands of servers. This requires non-trivial efforts in both architecture and management design, and is left as part of our ongoing investigation. In this paper, we describe OSA that is designed to interconnect a few thousands of servers in a container.

Roadmap. In Sec. II, we discuss the idea of OSA's unprecedented flexibility, followed by background on



A	G	10
B	H	10
C	E	10
D	F	10
B	D	10

A	G	10
B	H	10
C	E	10
F	G	20
B	D	10

Fig. 1. OSA adapts topology and link capacities to the changing traffic demands.

optical technologies for OSA. Then we describe OSA architecture (Sec. III) and its algorithm design (Sec. IV) in response to traffic patterns. In Sec. V and Sec. VI, we evaluate OSA via extensive simulations and implementation respectively. We discuss some design issues and related work to OSA in Sec. VII before concluding in Sec. VIII.

II. Motivation and Background

We first use a motivating example to show what kind of flexibility OSA delivers. Then, we introduce the optical technologies that make OSA possible.

A. A Motivating Example

We discuss the utility of a flexible network using a simple hypothetical example in Fig. 1. On the left is a hypercube connecting 8 ToRs using 10G links. The traffic demand is shown in the bottom-left of Fig. 1. For this demand, no matter what routing paths are used on this hypercube, at least one link will be congested. One way to tackle this congestion is to reconnect the ToRs using a different topology (Fig. 1, bottom-center). In the new topology, all the communicating ToR pairs are directly connected and their demand can be perfectly satisfied.

Now, suppose the traffic demand changes (Fig. 1, bottom-right) with a new (highlighted) entry replacing an old one. If no adjustment is made, at least one link will face congestion. With the shortest path routing, $F \leftrightarrow G$ will be that link. In this scenario, one solution to avoid congestion is to increase the capacity of the $F \leftrightarrow G$ to 20G at the expense of decreasing capacity of link $F \leftrightarrow D$ and link $G \leftrightarrow C$ to 0. Critically, note that in all three topologies, the degree and the capacity of nodes remain the same, *i.e.*, 3 and 30G respectively.

As above, OSA's flexibility lies in its flexible topology and link capacity. In the absence of such flexibility, the above example would require additional links and capacities to handle both traffic patterns. More generally, a large variety of traffic patterns would necessitate 1:1 over-subscription (*i.e.*, non-blocking) network construction. OSA, with its high flexibility, can avoid such non-blocking construction, while still providing equivalent performance for various traffic patterns.

B. Optical Networking Technologies

We now discuss the optical networking technologies that enable the above flexibility.

1. Wavelength Division Multiplexing (WDM): Depending on the channel spacing, using WDM, typically 40 or up to 100 channels or wavelengths can be transmitted over a single piece of fiber in the conventional or C-band. For the purposes of our architecture, each wavelength is rate-limited by the electrical port it connects to.

2. Wavelength Selective Switch (WSS): A WSS is typically a $1 \times N$ switch, consisting of one *common* port and N *wavelength* ports. It partitions (runtime-configurable within a few ms) the set of wavelengths coming in through the common port among the N wavelength ports. E.g., if the common port receives 80 wavelengths then it can route wavelengths 1–20 on port 1, wavelengths 30–40 and 77 on port 2, etc.

3. Optical Switching Matrix (OSM): Most OSM modules are bipartite $N \times N$ matrix where any input port can be connected to any one of the output ports. Micro-Electro-Mechanical Switch (MEMS) is the most popular OSM technology and achieves reconfigurable (at 10ms [17]) one-to-one circuit by mechanically adjusting micro mirrors. A few hundred ports are common for commercial products, and >1000 for research prototypes [18]. The current commercially available OSM modules are typically oblivious to the wavelengths carried across it. We use MEMS and OSM interchangeably.

4. Optical Circulators: Circulators enable bidirectional optical transmission over a fiber, allowing more efficient use of the ports of optical switches. An optical circulator is a three-port device: one port is a shared fiber or switching port, and the other two ports serve as send and receive ports.

5. Optical Transceivers: Optical transceivers can be of two types: coarse WDM (CWDM) and dense WDM

(DWDM). We use DWDM-based transceivers, which support higher bit-rates and more wavelength channels in a single piece of fiber compared to CWDM.

III. OSA Architecture

In this section, we introduce how OSA architecture is built from the above described optical technologies. Our current design is targeted for container-size DCNs.

A. Building Blocks

Flexible topology. OSA achieves flexible topology via exploiting the reconfigurability of MEMS. Say we start by connecting each of N ToRs to one port on an N-port MEMS. Given the MEMS bipartite port-matching, this implies that every ToR can only communicate with one other ToR at any instant, leaving the ToR level graph disconnected. If we connect N/k ToRs to k ports each at the MEMS, each ToR can communicate with k ToRs simultaneously. Here, $k > 1$ is the degree of a ToR, not its port count, in the ToR graph. The configuration of the MEMS determines which set of ToRs are connected; and OSA must ensure that the ToR graph is connected when configuring the MEMS.

Given a ToR graph connected by optical circuits through the MEMS, we use *hop-by-hop* stitching of such circuits to achieve network-wide connectivity. To reach remote ToRs that are not directly connected, a ToR uses one of its k connections. This first-hop ToR receives the transmission over fiber, converts it to electrical signals, reads the packet header, and routes it towards the destination. At each hop, every packet experiences conversion from optics to electronics and then back to optics (O-E-O) and switching at the ToR. Pure O-E-O conversion can be done in sub-nanoseconds [19]. Note that at any port, the aggregate transit, incoming and outgoing traffic cannot exceed the port's capacity in each direction. So, high-volume connections must use a minimal number of hops. OSA should manage the topology to adhere to this requirement. Evaluation in Sec. VI quantifies the overhead (both O-E-O and switching) of hop-by-hop routing.

Flexible link capacity. Every ToR has degree k. If each edge had fixed capacity, multiple edges may need to be utilized for this ToR to communicate with another ToR at a rate higher than a single edge supports. To overcome this problem, OSA combines the capability of optical fibers to carry multiple wavelengths at the same time (WDM) with the dynamic reconfigurability of the WSS. Consequently, a ToR is connected to the MEMS through a multiplexer and a WSS unit.

Specifically, suppose ToR A wants to communicate with ToR B using w times the line speed of a single port. The ToR will use w ports, each associated with a (unique) wavelength, to serve this request. WDM enables these w wavelengths, together with the rest from this ToR, to be multiplexed into one optical fiber that feeds the WSS. The WSS splits these w wavelengths to the appropriate MEMS port which has a circuit to ToR B (doing likewise for $k-1$ other wavelengths). Thus, a $w\times$ (*line-speed*) capacity circuit is set up from A to B, at runtime. By varying the value of w for every MEMS circuit, OSA achieves dynamic capacity for every edge.

We note that a fiber cannot carry two channels over the same wavelength in the same direction. Moreover, to enable a ToR pair to communicate using all available wavelengths, we require that each ToR port (facing the optical interconnect) is assigned a wavelength unique across ports at this ToR. The same wavelength is used to receive traffic as well: each port thus sends and receives traffic at one fixed wavelength. The same set of wavelengths is recycled across ToRs. This allows all wavelengths at one ToR to be multiplexed and delivered after demultiplexing to individual ports at the destination ToR. This wavelength-port association is a design time decision.

Efficient port usage. To make full use of the MEMS ports, we desire that each circuit over the MEMS be bidirectional. For this, we use optical circulators between the ToR and the MEMS ports. A circulator connects the send channel of the transceiver from a ToR to the MEMS (after the channel has passed through the WSS). It simultaneously delivers the traffic incoming towards a ToR from the MEMS, to this ToR. Note that even though the MEMS edges are bidirectional, the capacities of the two directions are independent of each other.

B. Putting it All Together: OSA-2560

Fig. 2 illustrates the general OSA architecture. We now discuss one specific instantiation, OSA-2560, with $N = 80$ ToRs, $W = 32$ wavelengths and ToR degree $k = 4$ using a 320-port MEMS to support 2560 servers.

Each ToR is a commodity electrical switch with 64 10-GigE ports [20]. 32 of these ports are connected to servers, while the remaining face the optical interconnect. Each port facing the optical interconnect has a transceiver associated with a fixed and unique wavelength for sending and receiving data. The transceiver

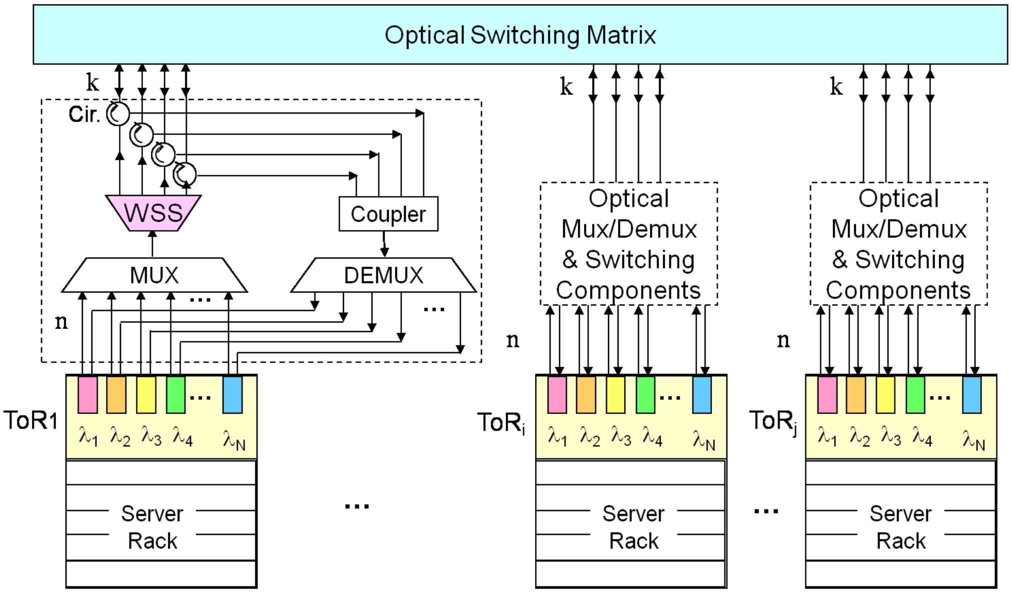

Optical Switching Matrix
k
Cir.
WSS
Coupler
MUX
DEMUX
n
ToR1
λ_1 λ_2 λ_3 λ_4 λ_N
Server Rack
k
Optical Mux/Demux & Switching Components
n
ToR$_i$
λ_1 λ_2 λ_3 λ_4 λ_N
Server Rack
k
Optical Mux/Demux & Switching Components
n
ToR$_j$
λ_1 λ_2 λ_3 λ_4 λ_N
Server Rack

Fig. 2. The overall OSA architecture; detailed structure is shown only for ToR$_1$ for clarity.

uses separate fibers to connect to the send and receive infrastructures.

The send fiber from the transceivers from each of the 32 ports at a ToR is connected to an optical multiplexer. The multiplexer feeds a 1×4 WSS. The WSS splits the set of 32 wavelengths it sees into 4 groups, each group being transmitted on its own fiber. These fibers are connected to the MEMS via circulators to enable bidirectional communications. The 4 receive fibers from 4 circulators are connected to a power coupler (similar to a multiplexer, but simpler), which combines their wavelengths onto one fiber. This fiber feeds a demultiplexer, which splits each incoming wavelength to its associated port on the ToR.

We point out two key properties of the above interconnect. First, each ToR can communicate simultaneously with any 4 other ToRs. This implies that the MEMS configuration allows us to construct all possible 4-regular graphs among ToRs. Second, through WSS configuration, the capacity of each of these 4 links can be varied in $\{0, 10, 20, \ldots, 320\}$ Gbps. The MEMS and WSS configurations are decided by a central OSA manager. The manager estimates the traffic demand, calculates appropriate configurations, and pushes them to the MEMS, WSS units and ToRs. This requires direct, out-of-band connections between the manager and these components. Note that our employment of such a central OSA manager is inspired by many recent works [2, 3, 11, 14, 21] in the context of DCNs given the fact that a DCN is usually owned and operated by a single organization.

Furthermore, we choose $k = 4$ for container-sized DCNs because it is a tradeoff between the network size and performance. A larger k value can enable one ToR to connect to more other ToRs simultaneously, thus achieving higher performance. However, given the fixed 320-port MEMS, it also means that fewer ToRs ($320/k$) can be supported. Our experiments with $k = 1, 2, 4, 8$ indicate that $k = 4$ can deliver considerable bisection bandwidth between thousands of servers.

C. Analysis

Table I lists the cost and power usage of different network elements. Table II is the comparison between the traditional network, hybrid structure, OSA and Fattree.

Traditional over-subscribed network. For connecting 2560 servers using a two-tiered 2:1 oversubscribed architecture, we use 80 48×10G port ToR switches that have 32 ports connected to servers. The remaining 16 ports at each ToR are connected to aggregation switches. We use a total of 80 aggregation switches each with 16×10G ports. The choice of 32 server-facing port and 16 aggregation-switch-facing ports results in 2:1 over-subscription. Note that we picked 2:1 over-subscription ratio because, for all traffic patterns we studied, OSA delivers network bisection bandwidth that is at least 60% of the non-blocking network (Sec. V). Thus, a 2:1 oversubscribed traditional network (50% of non-blocking) is a conservative comparison point. This architecture costs USD 4.6M and consumes

Element	\$	W	Element	\$	W
ToR (10G port)	.5K[†]	12.5[†]	(DE)MUX	3K	0
MEMS	.5K[†]	0.24[†]	Coupler	.1K	0
WSS	1K[†]	1[†]	Circulator	.2K	0
Transceiver	.8K	3.5	-	-	-

TABLE I
[†]Cost (USD) and power (Watt) per port for different elements, we refer the values from Helios [11].

Architecture	\$	KW	% of non-blocking
Traditional	4.6M	73	50%
Hybrid	5.6M	78	20%–50%[‡]
OSA	5.6M	73	60%–100%[‡]
Fattree	14.6M	196	100%

TABLE II
Cost, power and performance for different network architectures to support 2560 servers with 10GigE ports. ([‡]For traffic patterns we evaluate in Sec. V.)

72.96KW. The number of cross-ToR fibers required is 1280. The bisection bandwidth provided is 50% of the non-blocking network. However, for skewed traffic demands, it is desirable to allocate high fraction of this capacity to more demanding flows and achieve better cost/performance tradeoff.

Simplified model of the hybrid structure. Helios [11] and c-Through [14] are two well-known hybrid electrical/optical structures. The hybrid structure model we used here and in Sec. V is an abstract model that captures key aspects of both. In this model, each ToR has connections to an electrical network and an optical network. The electrical network is a two or three tiered tree with a certain over-subscription ratio (8:1 for Table II). In the optical part, each ToR has only one optical link connecting to one other ToR, but this link is of unrestricted capacity. This hybrid structure costs USD 5.6M, consumes 78KW and has 480 long fibers – 160 above the MUX in optical part and 320 above the ToRs in electrical part.

OSA. The total cost is approximately USD 5.6M, with a power consumption of 73KW. ToRs and transceivers are responsible for a large portion of the cost and power budget. Compared to the traditional architecture, the additional cost is mainly due to (DE)MUX and WSS units. The number of long fibers required by OSA is small – 320 fibers above the circulator layer. The ToR to circulator connection is very short and can be packaged with the ToR. OSA's cost is similar as the hybrid structure but is $\sim$20% more expensive than the traditional structure, however, it can dynamically adjust the bandwidth allocated to demanding flows. For all traffic demands we evaluated in Sec. V, this enables OSA to achieve 60%-100% of the non-blocking bisection bandwidth. The power consumption is nearly identical to that of the traditional oversubscribed network; this is because the total number of electrical ports used in both architectures are identical, and optical components add negligible power. We note that the cost of optics is expected to fall significantly with commoditization and production volume. Much of these benefits have already been reaped for electrical technology. There is also scope for packaging multiple components on a chip - the 32 transceivers and the MUX could be packaged into one chip. This will reduce power consumption, cost, as well as the number of fibers.

Fattree. The cost and power of Fattree depends solely on the number of ports needed: a Fattree topology with p port Ethernet switches can connect $p^3/4$ hosts with a total of $5{*}p^3/4$ ports. Note that for 10G port electrical switches, optical transceiver for remote connection is a necessity. To connect 2560 servers, Fattree costs 14.6M USD. The power consumption is 196KW. The number of fibers required above the ToR layer is 5120. Fattree is more expensive and consumes more power, because it is designed to provide non-blocking connectivity and is also highly fault-tolerant. Our intention is not to perform a head-to-head comparison with Fattree, but to illustrate the cost/power/performance tradeoff of building a non-blocking network architecture.

Summary. For data center deployments where skewed traffic demands are expected, we believe that OSA is a better alternative than both Fattree and traditional oversubscribed networks: Fattree suffers from significantly higher cost and cabling complexity, and traditional architectures are inflexible and cannot assign spare bandwidth to demanding flows on the fly. Compared with the hybrid structure, OSA can achieve better performance with similar cost and power consumption.

IV. Design

In this section, we present OSA network optimization in details. Our goal is to compute the optimal topology and link capacities such that the network bisection bandwidth maximized for a given traffic demand. Estimating traffic demand is not our main focus of this paper and we assume this can be readily achieved in a fashion similar to Helios [11], c-Through [14] or

Flyways [15]. For optimization, we need to find: 1) a MEMS configuration to adjust the topology to localize high traffic volumes, 2) routes between ToRs to achieve high throughput, low latency or avoid congestion, and 3) a configuration for each WSS to provision the capacities of its outgoing links.

In the following, we first present a mathematical formulation for optimization. Considering its complexity, we then introduce an approximation solution.

A. *Problem Formulation*

Given: A traffic demand D between ToRs – D_{ij} is the desired bandwidth from ToR$_i$ to ToR$_j$.

Variables: We use four sets of variables: $l_{ij} = 1$ if ToR$_i$ is connected to ToR$_j$ through MEMS and 0 otherwise; $w_{ijk} = 1$ if l_{ij} carries wavelength λ_k in the $i \rightarrow j$ direction and 0 otherwise; v_{ijk} is the traffic volume carried by wavelength λ_k along $i \rightarrow j$; a traffic-served matrix S – S_{ij} is the bandwidth achieved from ToR$_i$ to ToR$_j$. For the last two sets of variables, S_{ij} have end-to-end meaning, while v_{ijk} have hop-to-hop significance. For all variables, $k \in \{1, 2, \ldots, \lambda_{Total}\}$; $i, j \in \{1, 2, \ldots, \#\text{ToRs}\}$, $i \neq j$; l_{ij} are the only variables for which $l_{ij} = l_{ji}$, and all other variables are directional.

Objective: To achieve optimal network bisection bandwidth, we maximize the traffic served:

$$Maximize \sum_{i,j} S_{ij}. \qquad (1)$$

Constraints: If the number of outgoing ports of the WSS is W, then ToR$_i$ is connected to exactly W other ToRs:

$$\forall i : \sum_{j} l_{ij} = W. \qquad (2)$$

A wavelength λ_k can only be used between two ToRs if they are directly connected via MEMS:

$$\forall i, j, k : w_{ijk} \leq l_{ij}. \qquad (3)$$

To avoid wavelength contention, ToR$_i$ can only receive/send λ_k from/to at most one ToR:

$$\forall i, k : \sum_{j} w_{jik} \leq 1; \sum_{j} w_{ijk} \leq 1. \qquad (4)$$

Traffic carried by λ_k between two ToRs is limited by ToR port capacity (C_{port}) and wavelength capacity (C_λ):

$$\forall i, j, k : v_{ijk} \leq min\{C_{port}, C_\lambda \times w_{ijk}\}. \qquad (5)$$

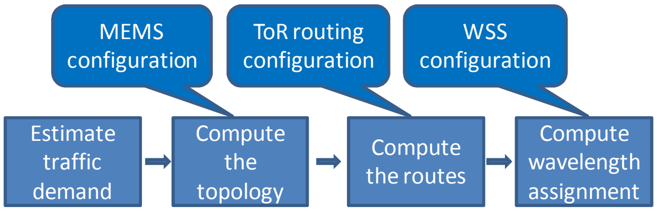


Fig. 3. The steps in OSA control algorithm.

The outgoing transit traffic is equal to the incoming transit traffic at ToR$_i$:

$$\forall i : \sum_{j,k} v_{ijk} - \sum_{j} S_{ij} = \sum_{j,k} v_{jik} - \sum_{j} S_{ji}. \qquad (6)$$

Finally, the traffic served is bounded by the demand:

$$\forall i, j : S_{ij} \leq D_{ij}. \qquad (7)$$

The above mixed-integer linear program (MILP) can be viewed as a maximum multi-commodity flow problem with degree bounds, further generalized to allow constrained choices in link capacities. While several variants of the degree-bounded subgraph and maximum flow problems have known polynomial time algorithms, the trivial combination of two is NP-hard [22].

B. *Solution*

As introduced above, in our approximation solution, we decompose the problem into three sequential subparts as shown in Fig. 3, *i.e.*, computing the topology, computing the routing and computing the wavelength assignment. Similar as Helios [11], in this paper, we adopt the traffic demand estimation method introduced by Hedera [23], which is based on the max-min fair bandwidth allocation for TCP flows in an ideal non-blocking network.

1. Compute the topology: We localize high-volume communicating ToR pairs over direct MEMS circuit links. This is accomplished by using a weighted b-matching [24], where b represents the number of ToRs that a ToR connects to via MEMS ($b = 4$ in OSA-2560). In the ToR graph, we assign the edge-weight between two ToRs as the estimated demand between them, and then cast the problem of localizing high-volume ToR connections to b-matching. Weighted b-matching is a graph theoretic problem for which polynomial-time algorithm exists [24]. We implement it using multiple perfect matchings, for which public library is available [25].

The b-matching graph above is not necessarily a connected graph. Fortunately, connectivity is easy to achieve via the edge-exchange operation [26]. First, we find all the connected components. If the graph is not connected, we select two edges $a \rightarrow b$ and $c \rightarrow d$ with lowest weights in different connected components, and connect them via replacing links $a \rightarrow b$ and $c \rightarrow d$ with links $a \rightarrow c$ and $b \rightarrow d$. We make sure that the links removed are not themselves cuts in the graph. The output of step 2 is used to tell MEMS on how to configure the new topology.

2. Compute the routes: Once we have connectivity, the MEMS configuration is known. We proceed to compute routes using any of the standard routing schemes such as the shortest path routing or low congestion routing. Note that some of the routes are single-hop MEMS connection while others are multi-hop ones. For simplicity, we use the shortest path routing in this paper. However, our framework can be readily applied to other routing schemes. The output of step 3 is used to tell ToRs on how to configure their routing tables.

3. Compute the wavelength assignment: Given the traffic demand and routes between any pair of ToRs, we can easily compute the capacity desired on each ToR link in order to serve the traffic demand on this link.

With the desired capacity demand on each link, we need to provision a corresponding amount of wavelengths to serve the demand. However, wavelength assignment is not arbitrary: due to the contention, a wavelength can only be assigned to a ToR at most once. Given this constraint, we reduce the problem to an edge-coloring problem on a multigraph. We represent our ToR level graph as a multigraph. Multiple edges correspond to the number of wavelengths between two nodes, and we assume each wavelength has a unique color. Thus, a feasible wavelength assignment is equivalent to an assignment of colors to the edges of the multigraph so that no two adjacent edges have the same color – exactly the edge-coloring problem [27]. Edge-coloring is a known problem and fast heuristics are known [28]. Libraries implementing this are publicly available.

We also require at least one wavelength to be assigned to each edge on the physical topology. This guarantees an available path between any ToR-pair, which may be required for mice/bursty flows. The output of step 4 is used to tell WSS on how to assign wavelengths.

All the above steps are handled by OSA manager. Specifically, OSA manager interacts with MEMS, WSS units and ToRs to control the topology, link capacities and routing respectively. We note that our decomposition heuristic is not optimal and there is room to improve. However, it provides satisfactory gains as we will see.

V. Simulation

In this section, we evaluate OSA-2560 via analytical simulations. We start with the simulation methodology, and then present the results.

A. *Simulation Methodology*

Simulation goals: Since our testbed only has 8 ToRs (Sec. VI), to evaluate OSA's capabilities at its intended scale, we conduct analytical estimation of network bisection bandwidth of OSA-2560 under various traffic patterns. Our results in this section are essentially computations of the expected bisection bandwidth in the steady state, ignoring software and hardware overheads which are considered in our testbed experiments in Sec. VI. We compare OSA with a non-blocking network, a hybrid network with varied over-subscription ratios in the electrical part and a 2:1 oversubscribed traditional network.

Communication patterns: We use the following real measurement traces and synthetic traffic data to evaluate the performance of OSA in the presence of changing communication patterns and traffic demands.

1. MapReduce-demand: We collected real traffic matrices in a production data center with around 400 servers, which mainly runs MapReduce applications[1]. We compute network demands by averaging the traffic over 30-second periods. For each demand, we identify the communication pattern by filtering out mice flows and focusing on the elephant ones. We map these communication patterns onto OSA-2560 using spatial replication.

2. Measurement-based: Recent measurements [15, 29] reveal several data center traffic characteristics. One important feature is that hotspot ToR links are often associated with a high fan-in (or fan-out), and most of the traffic (80%) are within the rack, resulting in highly skewed distribution. We synthesize this kind of traffic patterns by randomly choosing 12 hotspots out of 80 racks, with each one connecting to 6-10 other randomly chosen ToRs respectively. We intentionally assume all traffic exit the rack in order to create intensive communications.

[1] The name of the production data center company is anonymized.

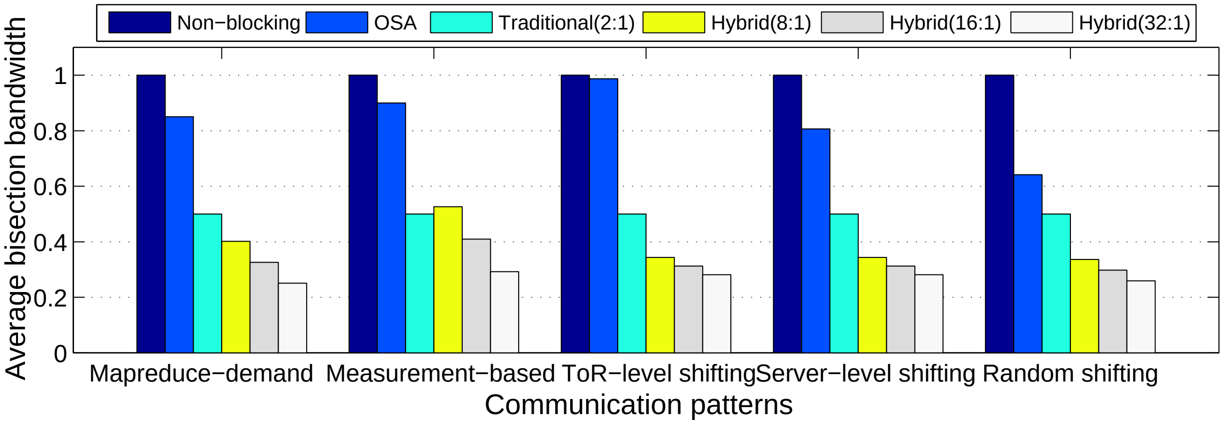


Fig. 4. The average network bisection bandwidth (normalized) achieved for different communication patterns.

3. ToR Level Shifting: We index the ToR switches from 0 to 79 and shift traffic round-by-round. Initially, all servers in ToR i talk to all servers in ToRs $(i \pm 1) \bmod 80$ and $(i \pm 2) \bmod 80$. Then we shift these communications to servers in the next ToR after each round.

4. Server Level Shifting: We index the servers from 0 to 2559. We start with server i talking to 4 other servers: $(i \pm 32) \bmod 2560$ and $(i \pm 64) \bmod 2560$. With 32 servers in a rack, initially, this implies that each rack communicates with 4 other racks. In successive rounds, server i talks to $(i \pm (32+s)) \bmod 2560$ and $(i \pm (64+s)) \bmod 2560$ $(s = 4, 8, 12, \cdots)$. This implies that each rack communicates with 6 racks in most rounds, with traffic spread across these 6 connections increasing and decreasing periodically.

5. Random Shifting: In each round, each server in ToR i talks to servers in up to 10 randomly selected ToRs. In this pattern, many ToRs may simultaneously talk to one ToR, creating hotspots and communication bottlenecks.

6. Increasing Destinations: We gradually increase the number of destinations for each ToR from 4 through 79 (*i.e.*, all-to-all communications) to further investigate the impact of traffic spread on OSA performance.

Evaluation metrics: First, we evaluate the network bisection bandwidth provided by OSA for each communication pattern. Then, we quantify the impact of flexible topology and flexible link capacity within OSA architecture respectively. Finally, we measure time cost of the control algorithm described in Sec IV-B. The experiments were conducted on a Dell Optiplex machine with Intel 2.33 GHz dual-core CPU and 4 GB Memory.

The hybrid structure: We simulate the hybrid structure model introduced in Sec. III-C which captures the key features of c-Through and Helios. To optimize the network to traffic, we run maximum weighted matching to determine which optical circuits to establish. Then we calculate how much of the remaining demand can be satisfied by the electrical network at best.

Traditional 2:1 oversubscribed network: We also simulate a 2:1 over-subscribed electrical network whose details were described earlier in Sec. III-C.

B. Evaluation Results

1) Performance of OSA: In this experiment, the topology and link capacities are adaptively adjusted to the current traffic pattern. As soon as traffic pattern changes, the network reconfigures its topology instantaneously. In practice, the performance of OSA would be also impacted by the time taken to estimate the traffic demand, the time taken by the algorithms to identify the appropriate topology, and reconfiguration time of optical devices. Experimental results from our prototype will encompass these overheads (see Sec. VI).

Fig. 4 shows the average network bisection bandwidth over 100 instances of each traffic pattern obtained by different DCN structures. Note that all the results are normalized by the bisection bandwidth of the non-blocking scenario. We make following observations.

First, we find that OSA delivers high bisection bandwidth (60%-100% of non-blocking) for both real and synthetic traffic patterns. Under the Mapreduce-demand, OSA can provide over 80% of the non-blocking bandwidth. This is because OSA adaptively changes its topology and link capacities according to the present traffic pattern. In our simulation setting, we choose 4-regular graph for OSA, that is why we are still 20%

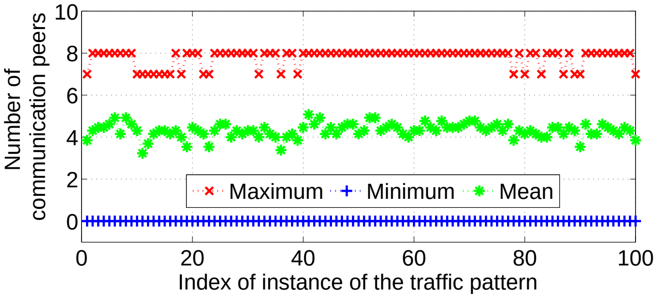


Fig. 5. Number of ToRs each ToR communicates with in every instance of the Mapreduce-demand pattern.

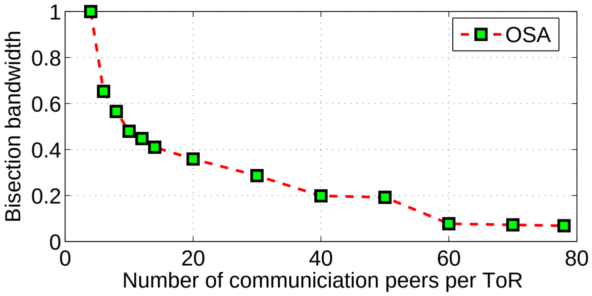


Fig. 6. Network bisection bandwidth with an increasing number of peers each ToR communicates with.

from non-blocking given the communication distribution shown in Fig. 5. Because some ToRs talk to more than 4 (up to 8) other ToRs, OSA cannot assign direct circuits to feed all these communications. The multi-hop routing possibly causes congestion on the intermediate switches, leading to performance degradation. From the figure, we find that OSA delivers higher bandwidth (90% of non-blocking) for the measurement-based pattern, because it has relatively less hotspots compared to the previous one.

Second, when each ToR communicates with 4 other ToRs (in the ToR-level shifting pattern), OSA achieves bisection bandwidth nearly identical to that of the non-blocking network. This result is not surprising given that OSA allows a 4-regular graph and hence provides 4 optical circuits at each ToR to perfectly support the demand. Note that the traditional 2:1 oversubscribed network delivers 50% of non-blocking for all traffic patterns.

Third, in our results (not shown here due to lack of space), we observe that the bisection bandwidth achieved by OSA oscillates periodically from approximately 60% to 100% (with average at 80%) of non-blocking for the server-level shifting pattern. This is because each ToR would periodically communicate with 4 and 6 other ToRs in such traffic pattern. We further observe that the bisection bandwidth obtained by OSA in the random shifting pattern is the worst – 60% of non-blocking. This is expected since the number of peers each ToR communicates with is larger than the other two shifting patterns. Specifically, for the ToR-level shifting, a ToR talks to 4 other peers; For the server-level shifting, a ToR communicates with 4-6 peers; While for the random shifting pattern, a ToR communicates with 5-20 peers. As discussed above, when the number of communication peers for a ToR is larger than 4, some flows will necessarily use multi-hop paths causing performance degradation. Concretely, for the ToR-level shifting most paths are direct, for the server-level shifting most paths are direct or 2 hops, and for the random shifting most paths are increased to 2-6 hops. The multi-hop paths taken by some flows contend for the available bandwidth at intermediate switches, thus limiting the peak achievable bandwidth.

Next, we present the bisection bandwidth achieved by OSA with an increasing number of inter-ToR communications. As it moves gradually to all-to-all communication (Fig. 6), as expected, the network bisection bandwidth drops due to extensive bandwidth contention at the ToRs. Note that the traditional 2:1 oversubscribed network would continue to perform at 50% of non-blocking. This result is presented only for comparison purposes since OSA is not designed for all-to-all communication.

Furthermore, we note that OSA outperforms the hybrid model by 80%-250% in our evaluation. This is not a surprising result because the hybrid model only has a perfect matching between ToRs in the optical part. This means that one ToR is able to talk to one other ToR at a time. We increase over-subscription ratios in the electrical part from 32:1 to 8:1 and see only incremental improvement due to the oversubscribed network. In contrast, in OSA-2560, we have a 4-regular graph meaning one ToR can directly communicate with 4 other ToRs simultaneously. Further, OSA also dynamically adapts its link capacities to the traffic demand. The higher flexibility of OSA leads to its better performance.

In Fig. 7, we inspect the performance delivered by OSA with varied k values (left) and the number of hops traversed by the traffic (right) using the Mapreduce-demand. We assume that there are always 80 ToRs. It is evident from the left figure that with a larger k value, the network bisection bandwidth delivered is higher. However, the larger k value also necessitates

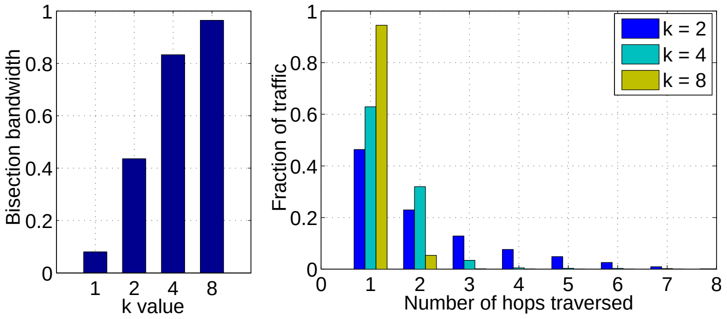


Fig. 7. The performance of OSA with varied k values (left) and the number of hops traversed by traffic (right).

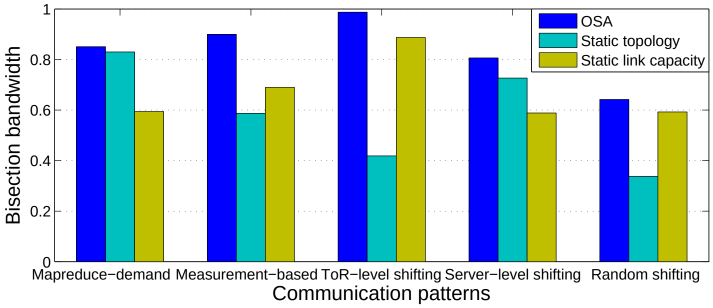


Fig. 8. Effect of flexible topology and link capacity.

Control algorithm	Time (ms)
Estimate traffic demand	161
Compute the topology	48
Compute the routes	41
Compute the wavelength assignment	40
Total	290

TABLE III
TIME CONSUMPTION OF THE CONTROL ALGORITHM.

more MEMS ports in order to support the same number of ToRs and servers. Note that $k = 2$, where we see low performance, is exactly equivalent to the optical part of the hybrid structure. From the right figure, we find that, for our case of OSA-2560 (*i.e.*, $k = 4$), the vast majority of traffic only traverses less than 3 hops - over 60% of traffic goes one hop and over 30% of traffic goes two hops. We also find that with a small k value, a considerable portion of traffic needs to traverse multiple hops to reach the destinations. When k increases, more traffic will go fewer hops, indicating better network performance. Though not shown, the similar trends hold for the rest traffic patterns.

2) Effect of Flexible Topology & Link Capacity: We quantify the effect of flexible topology and flexible link capacity respectively. For this purpose, in the first experiment we randomly select a fixed topology (*e.g.*, the one generated by the first instance of a traffic pattern), and only adjust the link capacity according to the current traffic pattern. In the second experiment, we hypothetically assume each link has 8 fixed wavelengths assigned (thus static link capacity), and only adjust the topology based on the current traffic pattern. Fig. 8 shows the bisection bandwidth of both scenarios and the original OSA. Comparing the static topology scenario with OSA, we observe up to $\frac{100\%-40\%}{40\%} = 150\%$ improvement due to the effect of flexible topology in case of the ToR-level shifting pattern. Comparing the static link capacity scenario with OSA, we observe up to $\frac{90\%-60\%}{60\%} = 50\%$ improvement because of the effect of flexible link capacity in case of the measurement-based traffic pattern. These results suggest that the flexible topology and link capacity are essential to improve the performance of OSA.

3) Time Cost of Control Algorithm: We measure the time cost of the OSA control algorithm as described in Sec IV-B. We run our current software implementation with 50 randomly selected traffic patterns that we used above and compute the average values for each step. As shown in Table III, the total time is 290 ms. We observe that out of the 4 steps, traffic demand estimation is dominant (161 ms). The reason is that the algorithm for this step is based on the number of servers, while the rest are based on the number of ToRs. Note that our demand estimation algorithm is adopted directly from Hedera [23], which has recently been shown to be less than 100 ms for large data centers via parallelization over multiple cores or machines. This means there is a large room to speed up with advanced technologies.

Though most of the remaining steps take only tens of milliseconds, we still believe optimizations are possible throughout the control software to make it more responsive even for larger networks. For instance, b-matchings for 1,024 nodes could be computed in as few as 250 ms in the year 2000 with contemporary hardware [24]. It is also likely that better-performing, faster heuristics can be built based on more accurate models of the traffic.

VI. IMPLEMENTATION

We have built a small-scale OSA prototype with real optical devices (Fig. 9). We first introduce our testbed setup, and then present our experiments over it.

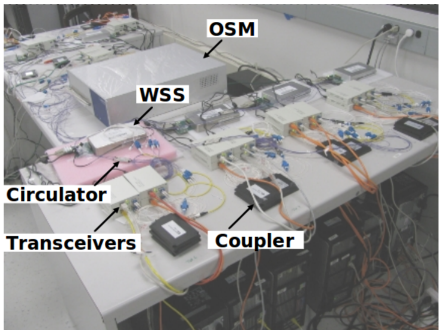


Fig. 9. OSA testbed.

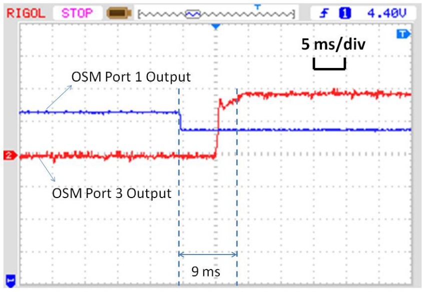


Fig. 10. Switching time of our OSM.

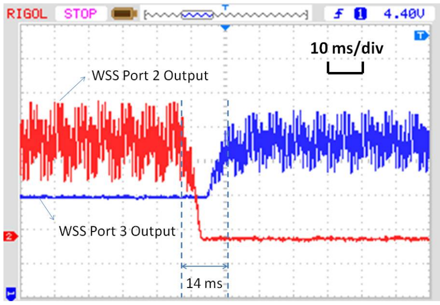


Fig. 11. Switching time of our WSS.

A. Testbed Setup

Our testbed connects 32 end-hosts, uniformly distributed in 8 racks. To reduce the cost, we configure 8 Dell Optiplex servers to emulate 32 end-hosts. Each server acts as a virtual rack of end-hosts (V-Rack), running 4 virtual-machines (VMs) to emulate 4 end-hosts.

We now do not have programmable ToR switches, so we use high-end servers to emulate ToRs. We have 4 Dell PowerEdge servers, each equipped with an Intel 2.4GHz quad-core CPU, 8GB DRAM and 12GigE NICs. On each such server, we deploy 2 VMs, giving us a total of 8 virtual ToRs (V-ToRs). Each V-ToR binds to 6 NICs: one is connected to one V-Rack, one is used for a control connection to OSA manager, and remaining 4 are used as uplinks to reach other V-ToRs via optical elements.

On top of each V-ToR is a 1×4 CoAdna WSS, a coupler, a circulator, a 1×4 MUX and DEMUX pair, and 4 transceivers (which are packaged into a media converted (MC) unit). As in Fig 2, each ToR uplink is connected to a transceiver, with the send-fiber of the transceiver connected through the MUX, the WSS and the circulator to the OSM; and the receive-fiber connected to the same circulator through the coupler and the DEMUX. We use a 1 Polatis series-1000 OSM with 32 ports which allows a 16×16 bipartite interconnect. (Each V-ToR has 2 uplinks connected to each of these two sets of 16 ports.) We use 4 wavelengths: 1545.32, 1544.53, 1543.73 and 1542.94 nm, corresponding to channel 40, 41, 42 and 43 of ITU grid with 100 GHz channel spacing.

Further, in our testbed, OSA manager is a separate Linux server and talks to the OSM and ToRs via Ethernet ports, and to the WSS units via RS-232 serial ports.

Optical signal reflection issue: We observed that some media converters did not work after reconfiguration. During debugging, we noticed that they received a signal even when there was no external optical signal, and the measured optical power was about -22∼-24 dB. Further exploration revealed that the received signal was caused by reflection of the output signal from the same SFP module through the optical connector at the end of the common port of the circulator. This reflected signal caused the media converter to validate the optical connection. When the intended external signal was received by the SFP module, the media converter did not automatically initiate a re-validation process, and therefore the intended new optical link could not be setup. One solution is to unplug and plug back the fiber connector to the SFP receiver port to re-initiate the optical link, but it is a manual process. Fortunately, our results are not affected by this current limitation since we ensure that the throughput of non-blocking network is calculated after obtaining the throughput of OSA using an identical set of non-reflecting SFP transceivers. We are investigating alternate solutions with help from the device vendors.

B. Understanding the Optical Devices

Two critical optical devices in OSA are OSM and WSS. A common concern for them is the reconfiguration overhead. To measure the overhead, Fig. 10 shows the output power level on two ports of the OSM over time, during a reconfiguration event. We see a clear transition period, *i.e.*, from the high→low output power level shift on one port, to the low→high output power level shift on the other port. We observe that the switching delay of our OSM is 9 ms, consistent with [11, 14].

Next, we measure the reconfiguration time of WSS by switching a wavelength channel between two output

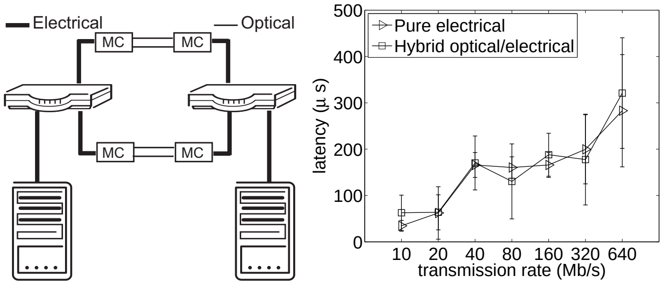


Fig. 15. O-E-O conversion.

ports. As shown in Fig. 11, this transition period is around 14 ms. However, OSA manager can performs the reconfiguration of the OSM and WSS in parallel to reduce the total penalty of reconfiguration.

C. Understanding the O-E-O Conversion

To measure the impact of O-E-O conversion, we specially connect 4 servers as in Fig. 15 (left). Two servers in the middle are configured as routers and equipped with optical media converters. We create a routing loop by configuring the IP forwarding tables of the routers. In each router, we deploy a `netfilter` kernel module and utilize the `NF_IP_PRE_ROUTING` hook to intercept all IP packets. We record the time lag between the instant when the packets first arrive in the network and when their TTL expires. This way, we are able to measure the multi-hop latency for O-E-O conversion and compare it with the baseline where all servers are directly connected using only electrical devices. Results in Fig. 15 (right) compare the average one-hop switching latency for both the hybrid optical/electrical and pure electrical architectures under different traffic load. It is evident from the figure, that O-E-O conversion does not incur noticeable (the maximum deviation in the absolute value and standard deviation is 38 and 58 μs, respectively), if any, additional switching latency. demonstrating feasibility of O-E-O employed by OSA.

D. OSA System Performance

We conduct two sets of experiments: one is for original OSA and the other is OSA with static topology. We use synthetic traffic patterns similar to Sec V-A. More specifically, traffic is described by parameters (t, r): servers in ToR i ($i = 0 \cdots 7$) send traffic to servers in t ToRs, *i.e.*, $[i+r, i+r+1, ..., i+r+(t-1)]$. We change t from 1 to 7 to generate different traffic loads (t=7 means all-to-all communication). For each t, we vary r from 1 to 7.

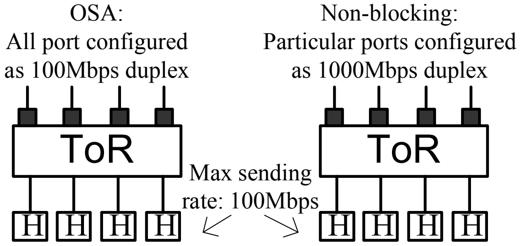


Fig. 16. Make a non-blocking network from OSA.

Our goal is to compare the achieved bisection bandwidth of OSA against that of a non-blocking network as the traffic spread out (with increasing t), and to measure the effect of topology reconfiguration. Note that varying r with a fixed t does not produce fundamentally different traffic distributions, it merely permutes which ToRs talk with which other ToRs, thus necessitating a change of topology without a change in traffic load or spread.

In our testbed, the NICs of servers support 10, 100, and 1000 Mbps full-duplex modes. In all our experiments, we limit the maximum sending rate of each flow to be 100 Mbps. This enables us to emulate a non-blocking network for comparison (Fig. 16): for OSA, all the uplink ports of ToRs are set at 100 Mbps, while for non-blocking, we increase particular uplink ports to 1000 Mbps to satisfy the traffic demands we simulate.

Results of OSA: Fig. 12 shows the average bisection bandwidth of OSA with changing traffic (t=$1 \cdots 7$). For each t, r steps 1 through 7 every 20 seconds. The network topology is dynamically reconfigured according to the current traffic demand. The results are along expected lines. We observe that the achieved bisection bandwidth of OSA is within 95% of the non-blocking network when t is 1 or 2. This is because when $t = 1$ each ToR talks with 2 other ToRs and when $t = 2$ each ToR talks with 4 other ToRs. Given that our topology is a 4-regular graph, OSA assigns direct links to each pair of communicating ToRs for efficient communication. For $t > 2$, the performance of OSA decreases, along similar lines as in the simulation (Sec. V). A careful reader will notice that the performance of our testbed under all-to-all communication is 58% of non-blocking, much higher than that in our simulation results. The reason is simple: our testbed has 8 ToRs each with degree 4, while our simulations used a sparse graph with 80 ToRs each having degree 4. Our intention with the testbed results is to demonstrate the feasibility of OSA rather than to show the performance achieved in a real deployment.

Next, Fig. 13 shows the impact of optical device

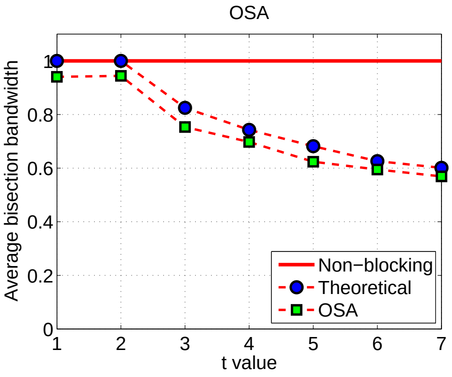


Fig. 12. Average bisection bandwidth of OSA.

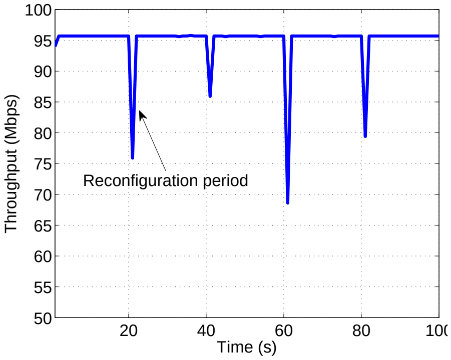


Fig. 13. Throughput of a flow in the presence of reconfigurations.

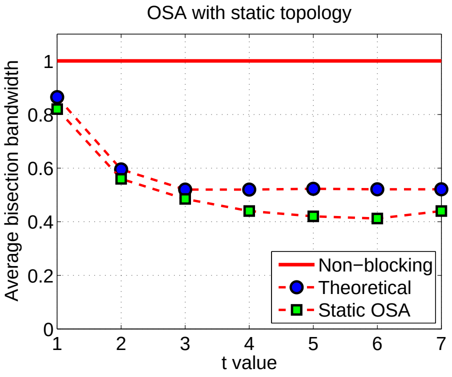


Fig. 14. Average bisection bandwidth of OSA with a static topology.

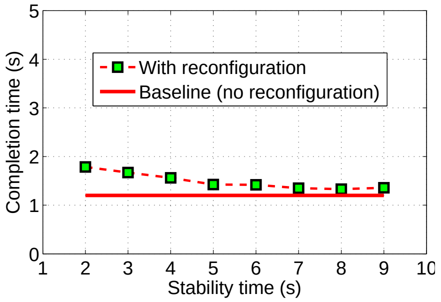


Fig. 17. Impact of topology reconfiguration on bulk data transfer.

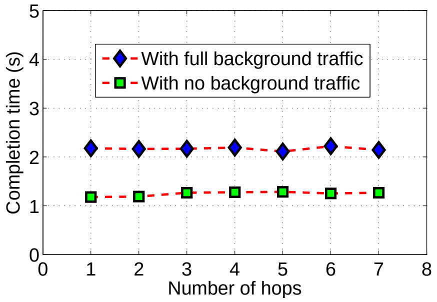


Fig. 18. Impact of multi-hop routing on bulk data transfer.

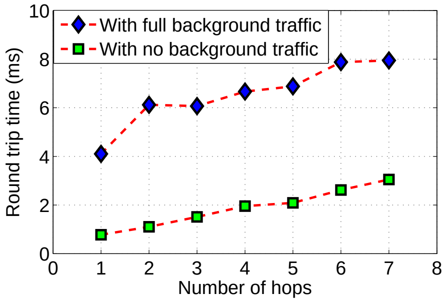


Fig. 19. Impact of multi-hop routing on simulated mice flows.

reconfigurability on the end-to-end throughput between two hosts. We observe that the performance drops during reconfiguration but quickly resumes after it finishes.

Finally, we also present the theoretical bisection bandwidth achievable in our testbed that ignores the overhead of reconfiguration, software routing, and TCP/IP protocol, etc. We observe that the gap between theoretically achievable values and OSA is within 5-7%, suggesting that our prototype is efficient.

Results of OSA with a static topology: We randomly select a topology and run the same experiments as above and present results in Fig. 14. Given the small diameter of our topology, the static topology OSA still achieves satisfactory performance. For example, in the worst case of all-to-all traffic (*i.e.*, $t = 7$), static OSA achieves more than 40% of the non-blocking network's bisection bandwidth. Since all the paths are 1 or 2-hop long, even the randomly selected topology performs satisfactorily.

For different t values, we find that the performance of OSA on the static topology is lower than that on dynamic topology by 10%-40%. This is because the topology is not optimized for the current traffic patterns. We expect that on a larger network where OSA topology is sparse (*e.g.*, the one we used in Sec. V), this performance gap will become more pronounced, highlighting the need for a dynamically optimized network for better performance.

E. Bulk Data Transfer

We study how the network reconfiguration and multi-hop routing affect the bulk data transfer, *i.e.*, elephant flows.

Impact of network reconfiguration: We periodically reconfigure the network and observe the completion time of transferring a chunk of data (100 MB file transferred using `scp`) during the reconfiguration events. We present mean value of 100 trials. Fig. 17 shows our results and the baseline performance where no reconfiguration takes place. The stability time is defined as the lifetime for a single static topology, after which the network is reconfigured. We notice that the completion time increases in the presence of reconfigurations. After analyzing the network trace using `tcpdump`, we observed that the round trip time (RTT) and accordingly the initial retransmission time out

(RTO) value in data centers are very small (sub-ms level), while network reconfiguration requires tens of milliseconds. As a consequence, each reconfiguration almost always triggers RTO events, after which TCP waits for 200 ms (Linux default RTO value) before the next retransmission, thereby degrading throughput and increasing latency. Recent work [30]–[32] has pointed out TCP's RTO issues in data centers, and proposed to reduce it to the μs level by employing fine-grained timers. We expect TCP's performance in OSA under network reconfiguration to significantly improve once these changes are adopted. We also notice from the figure that the completion time decreases as the stability time increases – larger stability period results in fewer network state changes and therefore fewer RTO events during the course of data transfer.

Impact of multi-hop routing: Our prototype topology is a low-diameter network due to a 8-node 4-regular graph. In order to evaluate the impact of multi-hop routing on bulk data transfer, we intentionally rearrange our 8 ToR switches in a line to form a linear topology with larger diameter. In Fig. 18, we measure the completion time of data transfer (100 MB file transferred using `scp`) in terms of the number of hops they pass through. Specifically, we consider two scenarios. In the first case, the network is free of background traffic. In the second case, all the links in the network are saturated by other elephant TCP flows. From the figure, we find that in both cases the completion time is relatively consistent regardless of the hops. This gives us confidence that multi-hop routing does not affect the performance of bulk data transfer seriously. We can further notice from the figure that the influence of multi-hop O-E-O conversion during data transfer is negligible. We also observe a nearly constant gap between the two curves, which is due to different link utilizations in the two experiments.

F. Mice Flow Transfer

After inspecting the performance of bulk data transfer, we further check the impact of multi-hop routing on transferring mice flows. For this purpose, we use `ping` to emulate latency sensitive flows and evaluate its performance with/without background traffic as above. Fig. 19 shows the average round trip time (RTT) of 100 `ping` packets with varying path lengths. As expected, the RTT increases with more hops. However, we find that the absolute increment is small: 1ms (without background traffic) and 2ms (with full background traffic), respectively, after 7 hops. These results suggest that the hop-by-hop stitching of optical paths is a feasible approach to provide overall connectivity. We note that network reconfiguration may have non-trivial impact on the latency-sensitive flows transfer, since it happens on the order of 10ms. We further discuss options to handle such issue in Sec. VII.

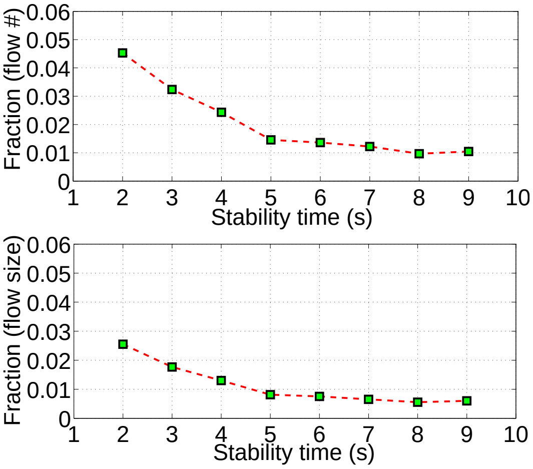


Fig. 20. The potentially affected mice flows.

VII. Discussion and Related Work

A. Mice Flow during Reconfiguration

OSA ensures that all ToRs are in a connected graph and uses hop-by-hop stitching of existing circuits to provide overall network connectivity. However, during network reconfiguration, a pair of ToRs may be temporarily disconnected for around 10 ms. While this can be largely tolerated by latency-insensitive applications such as Mapreduce or Dryad, it would affect those operating with latency-sensitive mice flows like Dynamo [33].

In Fig. 20, we estimate, in the worst case, how many mice flows (in terms of flow count and size) can be potentially affected due to the reconfiguration. We used the production data center traffic from Sec. V-A and used 10 MB to differentiate elephant flows from small ones. We find that for this particular dataset, when the stability time varies from 9 to 2 seconds, there are 1% to 4.5% of the mice flows that can be affected during the reconfigurations. This implies that as the network experiences more frequent reconfigurations, a larger fraction of mice flows may get affected. We next discuss two possible options to handle this issue.

Our basic idea is to reserve a static, connected channel in the OSA network. To do so, we can reserve a small number of wavelengths and MEMS/WSS ports that are never reconfigured and mice flows are always sent over them. Such a channel can be simply a spanning tree or other connected topologies. Given the topology of the channel is controlled by MEMS, we can arrange it in a low-diameter manner so that the transmission of mice flows is optimized. However, this approach consumes expensive MEMS/WSS ports, which otherwise can be better utilized for other applications or at stable time.

An alternative approach to building the channel without using MEMS/WSS ports is directly connecting all the ToRs together to form a ring or a star network. For the ring, we can reserve 2 ports on each ToR and directly connect them iteratively. In case of OSA-2560 with 80 ToRs, the diameter is 40 hops. To reduce the path length, it is possible to reserve more ports on each ToR and connect them structurally using DHT techniques [34], *e.g.*, the diameter is expected to be 3-4 hops with high probability for 80 ToRs if we reserve 4 ports on each ToR. Another option is to employ one additional central electrical switch – each ToR uses 1 port to connect to the central switch. Note that, in Helios or c-Through, the electrical switches (usually forming tree or even multi-root tree) are used for overall connectivity among all Pods/ToRs. In OSA, the all-to-all connectivity is maintained by optical components. A comprehensive evaluation and comparison of these solutions is part of our ongoing work.

B. OSA Applicability vs Traffic Properties

For all-to-all traffic, the non-oversubscribed network is indeed more appreciated. However, such workloads are neither reflected in our dataset nor in measurements elsewhere [2, 15, 29]. Our flexible OSA architecture would work best when traffic pattern is skewed and stable on the order of seconds. It has been noted in [5] over measurements of a 1500-server production DCN that "Only a few ToRs are hot and most of their traffic goes to a few other ToRs." Another study [2], also on a 1500-server production DCN, shows that more than 90% of bytes flow in elephant flows. Regarding traffic stability, a similarly sized study [35] shows that 60% of ToR-pairs see less than 20% change in traffic demand for between 1.6 to 2.2 seconds on average. Despite these, we expect that OSA may exhibit undesirable performance degradation if the traffic pattern is highly dynamic, in which case any topology adaptation mechanism may be unsuitable as the situation changes instantaneously. In practice, the infrastructure manager should choose the proper sensitivity of OSA according to the operational considerations.

C. Scalability

The current OSA design focuses on container-size DCNs. To scale, we may confront several challenges. The first one is the MEMS's port density. While the 1000-port MEMS is theoretically feasible, the largest MEMS as of today has 320 ports. One natural way to increase the port density is via interconnecting multiple small MEMS switches. However, this poses additional requirement for fast coordinated circuit switching. Secondly, larger network size necessitates more control and management. In our OSA-2560 with 80 ToRs, all the intelligences, *e.g.*, network optimization and routing, are handled by OSA manager. How to handle such tasks in a larger DCN with thousands of ToRs is an open question especially when the network environment is dynamic. Further, circuit visit delay [14] is another issue to notice when scaling. We are considering all these challenges in our continuous effort designing a scalable optical DCN.

D. Closely Related Work

OSA's design goals are closely related to those of c-Through [14] and Helios [11]. In both approaches, flows requiring high bandwidth are dynamically provisioned on optical circuits while a parallel electrical network is used to provide overall connectivity. OSA differs from these prior proposals in its degree of flexibility and its architecture. Both Helios and c-Through achieve some topology flexibility via a limited number of single-hop optical links. In their optical part, one ToR only connects to one other ToR at a time. While it can connect to different ToRs at different times, the switching latency would be around 10 ms. On the contrary, in OSA, one ToR can connect to multiple ToRs simultaneously at a time and multi-hop connection exists between any pair of remote ToRs via hop-by-hop circuit stitching. Furthermore, OSA allows the link capacities to be adjusted on the fly. Unlike these existing hybrid architectures, OSA avoids using electrical components other than the ToR switches.

OSA is more comparable to c-Through than Helios, because its current target is inter-rack DCNs with a few thousand servers unlike Helios' inter-container mega-DCN scale. Qualitatively, OSA provides more flexibility than either Helios or c-Through and is able to serve a larger space of skewed traffic demands with performance similar to that of non-blocking interconnects. We present a coarse quantitative comparison with an abstract hybrid architecture model in Sec. V, showing that OSA achieves significantly higher bisection bandwidth.

Recently, Kandula et al. [5, 15] proposed dynamically configuring 60GHz short-distance multi-Gigabit wireless links between ToRs to provide additional bandwidth for hotspots. Optical and wireless interconnects provide different trade-offs. For example, wired optical interconnects can deliver much more bandwidth at lower power consumption over long-distance, while wireless has lower costs and is easier to deploy though management and interference are challenging issues to deal with.

VIII. Conclusion

In this paper, we have presented OSA, a novel Optical Switching Architecture for DCNs. OSA is highly flexible because it can adapt its topology as well as link capacities to different traffic patterns. We have evaluated OSA via extensive simulations and prototype implementation. Our results suggest that OSA can deliver high bisection bandwidth (60%-100% of non-blocking) for a series of traffic patterns. Our implementation and evaluation with the OSA prototype further demonstrate its feasibility.

Acknowledgements

We thank CoAdna Photonics and Polatis for the equipment support of WSS and OSM respectively, Guangxun Liao for helpful discussions on WSS properties, and Don DePaolo for help during the testbed setup.

References

[1] M. Al-Fares, A. Loukissas, and A. Vahdat, "A scalable, commodity data center network architecture," in *SIGCOMM*, 2008.

[2] A. Greenberg, J. R. Hamilton, N. Jain, S. Kandula, C. Kim, P. Lahiri, D. A. Maltz, P. Patel, and S. Sengupta, "VL2: A scalable and flexible data center network," in *ACM SIGCOMM*, 2009.

[3] R. N. Mysore, A. Pamboris, N. Farrington, N. Huang, P. Miri, S. Radhakrishnan, V. Subramanya, and A. Vahdat, "Portland: A scalable fault-tolerant layer 2 data center network fabric," in *ACM SIGCOMM*, 2009.

[4] C. Guo, G. Lu, D. Li, H. Wu, X. Zhang, Y. Shi, C. Tian, Y. Zhang, and S. Lu, "BCube: A high performance, server-centric network architecture for modular data centers," in *ACM SIGCOMM*, 2009.

[5] S. Kandula, J. Padhye, and P. Bahl, "Flyways to de-congest data center networks," in *ACM HotNets*, 2009.

[6] K. Barker and etal., "On the feasibility of optical circuit switching for high performance computing systems," in *SC*, 2005.

[7] J. Hamilton, "Data center networksare in my way," "http://mvdirona.com/jrh/TalksAndPapers/JamesHamilton_CleanSlateCTO2009.pdf".

[8] H. Liu, C. F. Lam, and C. Johnson, "Scaling optical interconnects in datacenter networks opportunities and challenges for wdm," in *IEEE Symposium on High Performance Interconnects*, 2010.

[9] C. Lam, H. Liu, B. Koley, X. Zhao, V. Kamalov, and V. Gill, "Fiber optic communication technologies: What's needed for datacenter network operations," 2010.

[10] J. Rath, "Google eyes "optical express" for its network," "http://www.datacenterknowledge.com/archives/2010/05/24/google-eyes-optical-express-for-its-network/".

[11] N. Farrington, G. Porter, S. Radhakrishnan, H. H. Bazzaz, V. Subramanya, Y. Fainman, G. Papen, and A. Vahdat, "Helios: A hybrid electrical/optical switch architecture for modular data centers," in *ACM SIGCOMM*, 2010.

[12] CIR, "40g ethernet c- closer than ever to an all-optical network," "http://cir-inc.com/resources/40-100GigE.pdf".

[13] ADC, "40 & 100 gigabit ethernet: An imminent reality," "http://www.adc.com/Attachment/1270718303886/108956AE,0.pdf".

[14] G. Wang, D. G. Andersen, M. Kaminsky, K. Papagiannaki, T. S. E. Ng, M. Kozuch, and M. Ryan, "c-Through: Part-time optics in data centers," in *ACM SIGCOMM*, 2010.

[15] D. Halperin, S. Kandula, J. Padhye, P. Bahl, and D. Wetherall, "Augmenting data center networks with multi-gigabit wireless links," in *SIGCOMM*, 2011.

[16] M. Meringer, "Regular graphs," http://www.mathe2.uni-bayreuth.de/markus/reggraphs.html.

[17] T. Truex, A. A. Bent, and N. W. Hagood, "Beam steering optical switch fabric utilizing piezoelectric actuation technology," in *NFOEC*, 2003.

[18] J. K. et.al, "1100×1100 port mems-based optical crossconnect with 4-db maximum loss," *IEEE Photonics Technology Letters*, vol. 15, no. 11, pp. 1537 –1539, 2003.

[19] G. Keeler, D. Agarwal, C. Debaes, B. Nelson, N. Helman, H. Thienpont, and D. Miller, "Optical pump-probe measurements of the latency of silicon cmos optical interconnects," *IEEE Photonics Technology Letters*, vol. 14, no. 8, pp. 1214 – 1216, 2002.

[20] Broadcom, "Bcm56840 series enables mass deployment of 10gbe in the data center," http://www.broadcom.com/products/features/BCM56840.php.

[21] K. Chen, C. Guo, H. Wu, J. Yuan, Z. Feng, Y. Chen, S. Lu, and W. Wu, "Generic and automatic address configuration for data center networks," in *SIGCOMM*, 2010.

[22] E. Akcali and A. Ungor, "Approximation algorithms for degree-constrained bipartite network flow," in *ISCIS*, 2003.

[23] M. Al-Fares, S. Radhakrishnan, B. Raghavan, N. Huang, and A. Vahdat, "Hedera: Dynamic flow scheduling for data center networks," in *NSDI*, 2010.

[24] M. Müller-Hannemann and A. Schwartz, "Implementing weighted b-matching algorithms: insights from a computational study," *J. Exp. Algorithmics*, vol. 5, p. 8, 2000.

[25] L. Library.

[26] K. Obraczka and P. Danzig, "Finding low-diameter, low edge-cost, networks," USC, Tech. Rep., 1997.

[27] E. coloring.

[28] J. Misra and D. Gries, "A constructive proof of vizing's theorem," *Inf. Process. Lett.*, vol. 41, no. 3, pp. 131–133, 1992.

[29] T. Benson, A. Akella, and D. Maltz, "Network traffic characteristics of data centers in the wild," in *IMC*, 2010.

[30] Y. Chen, R. Griffith, J. Liu, R. H. Katz, and A. D. Joseph, "Understanding TCP Incast Throughput Collapse in Datacenter Networks," in *ACM WREN*, 2009.

[31] V. Vasudevan and etal., “Safe and Effective Fine-grained TCP Retransmissions for Datacenter Communication,” in *ACM SIGCOMM*, 2009.

[32] H. Wu, Z. Feng, C. Guo, and Y. Zhang, “ICTCP: Incast Congestion Control for TCP,” in *ACM CoNext*, 2010.

[33] G. DeCandia, D. Hastorun, M. Jampani, G. Kakulapati, A. Lakshman, A. Pilchin, S. Sivasubramanian, and P. Vosshall, “Dynamo: Amazon’s highly available key-value store,” in *SOSP*, 2007.

[34] A. Rowstron1 and P. Druschel, “Pastry: Scalable, decentralized object location and routing for large-scale peer-to-peer systems,” in *Middleware*, 2001.

[35] T. Benson, A. Anand, A. Akella, and M. Zhang, “The case for fine-grained traffic engineering in data-centers,” in *USENIX INM/WREN*, 2010.