DETOUR: A Large-Scale Non-blocking Optical Data Center Fabric

Jinzhen Bao[1,2], Dezun Dong[2(✉)], and Baokang Zhao[2]

[1] PLA Academy of Military Science, Beijing, China
[2] National University of Defense Technology, Changsha, China
{baojinzhen,dong,bkzhao}@nudt.edu.cn

Abstract. Optical data center networks (DCNs) are attracting growing interest due to the technical strength compared to traditional electrical switching networks, which effectively eliminates the potential hotspot caused by over-subscription. However, the evolving traffics with high fan-out and various patterns pose new challenges to optical DCNs. Prior solutions are either hard to support high fan-out communications in large-scale or suffer from limited connections with low performance.

In this paper we propose DETOUR, a large-scale non-blocking optical switching data center fabric. DETOUR composes of optical circuit switches (OCSes) and connects them in a 2D-Torus topology. It supports up to 729 racks and 69K+ ports with each OCS having 96 wavelengths. DETOUR utilizes a broadcast-and-select mechanism and enables signals optically forwarded to any dimension. Moreover, it realizes non-blocking by recursively adjusting conflict links between the diagonal forwarding OCSes. Our extensive evaluation results show that DETOUR delivers comparable high performance to a non-blocking optical switching fabric. It outperforms up to 2.14× higher throughput, and reduces 34% flow completion times (FCT) and 21% energy consumption compared with the state-of-the-art works.

1 Introduction

Data centers as the infrastructure of cloud computing, are rapidly expanded to meet the increasing demand of cloud services, big data and high performance applications. Many novel network architectures have been proposed to efficiently connect tens of thousands servers inside data centers. Pure electrical switching architectures, such as Fat-Tree [4], BCube [13] and Jellyfish [20], provide static and uniform interconnections among servers, without considering the dynamic traffic patterns. Due to the mismatch between the static interconnections and the dynamic network traffic, pure electrical switching networks must pay extremely high cost and complex wiring to deliver high bisection bandwidth.

Owning on the traffic characteristics of frequently concentrated and bursty [14], optical switching technologies are introduced to DCNs due to their reconfigurability, higher bit-rates and lower power [8,9,12,17,21,25]. Optical DCNs

R. Yokota and W. Wu (Eds.): SCFA 2018, LNCS 10776, pp. 30–50, 2018.
https://doi.org/10.1007/978-3-319-69953-0_3

support on-demand link connectivity and bandwidth allocation, which mitigating potential hotspots caused by the over-subscription. However, the increasing large-scale data-intensive applications have produced new traffic characteristics and pose challenges to existing optical DCNs:

(1) *High Fan-Out.* Traces from production clusters (e.g., Microsoft [16], Facebook [18], and Google [19]) show that source top-of-rack (ToR) electrical packet switches (EPSes) usually communicate with tens to hundreds of other EPSes simultaneously and have the stability across time periods from seconds to days. Constructing high fan-out EPSes connections in large-scale is significant to improve the network throughput and reduce flow completion times (FCT).
(2) *Various Communication Patterns.* The iterative computing frameworks (e.g., MapReduce, Spark, Hadoop) for large scale data analytics contain various communication patterns, such as unicast, multicast and broadcast (*-cast). Multicast and broadcast data dissemination are always the performance bottleneck for data analytics applications [11].

Along with the scale of DCN expanding, the ultimate goal is to provide non-blocking network services in large-scale with high flexibility. However, existing optical switching networks fail to meet all of the goals (as summarized in Table 1). Most designs are based on the techniques of Microelectromechanical system (MEMS) Optical Circuit Switch (OCS), Wavelength Division Multiplexing (WDM) and Wavelength Selective Switch (WSS).

(1) MEMS-based OCS is a $N \times N$ non-blocking switching matrix which mechanically rotates mirrors to direct any input signals to any one of the output ports. c-Through [21], Helios [12] and OSA [8] leverage a single MEMS-based OCS. However, the network scale is limited, since MEMS-based OCS is hard to scale and difficult to support high fan-out traffic patterns.
(2) WDM technology multiplexes multiple non-interfering wavelengths onto a single fiber, which supports up to 100 wavelengths by using Dense WDM (DWDM).

Table 1. Summary of prior optical DCNs and comparison to detour

Optical DCNs	Scalability(Ports)	Non-blocking	Flexibility
c-Through [21] Helios [12] OSA [8]	Low (∼2000)	Yes	Yes
Wavecube [9]	High (Unlimited)	No	No
Mordia [17], MegaSwitch [10]	Low (∼704) Low (∼6000)	Yes	Yes
OvS [25]	High (100K+)	No	Yes
DETOUR	**High (69K+)**	**Yes**	**Yes**

(3) WSS is reconfigurable to switch the input multiplexed wavelengths to desired output. WDM and WSS are usually used together to construct a distributed optical switching fabric (e.g., Wavecube [9], Mordia [17], MegaSwitch [10] and OvS [25]). Although Wavecube is scalable, it is blocking due to the optical links are reconfigurable only between neighbor switches. Mordia, MegaSwitch and OvS construct a broadcast-and-select optical switching fabric. They naturally supports unicast, multicast and broadcast with high flexibility. However, Mordia and MegaSwitch have low scalability. OvS is scalable based on the 2D-Torus topology, but it blocked without supporting arbitrary connections.

In this paper, we propose DETOUR, a large-scale and non-blocking optical switching data center fabric. DETOUR utilizes the DWDM technique and connects OCSes in a 2D-Torus topology. It can easily extend up to 729 racks and 69K+ ports when each OCS supports 96 wavelengths. DETOUR utilizes broadcast-and-select optical switching mechanism. The multiplexed DWDM signals are broadcasted along the same horizontal and vertical dimensions with the source OCS, and can also be optically forwarded to other orthogonal dimensions by the crosspoint OCSes. DETOUR recursively adjusts the conflicting broadcasting paths between diagonal forwarding OCSes to realize a non-blocking optical switching fabric.

We summarize the contributions of this paper as follows:

- We propose a novel optical switching data center network architecture, which is non-blocking in large-scale, named DETOUR. DETOUR delivers high scalability with up to 729 racks and 69K+ ports. By utilizing the broadcast-and-select mechanism and enabling optically forwarding signals to other orthogonal dimensions, DETOUR delivers high flexibility with establishing directly connected and dynamic bandwidth links between arbitrary EPS pairs.
- We develop control algorithms to optimize the network performance, including demand estimation, topology generation, wavelength assignment, reconfiguration and so on. Especially, we realize a non-blocking wavelength assignment algorithm by recursively adjusting the conflicting wavelengths, which exploiting the topology properties of DETOUR. And we also prove the non-blocking property in theory.
- We realize an event-based flow level simulator and conduct extensive simulations. Our simulation results show that DETOUR delivers comparable high performance to a non-blocking optical switching fabric. It outperforms up to 2.14× higher throughput and reduces 34% FCT and 57% energy consumption compared with the state-of-the-art works.

The rest of this paper is organized as follows. Section 2 describes the architecture of DETOUR. Section 3 details the algorithms realized in the controller. In Sect. 4, we implement a flow level simulator and evaluate the performance of DETOUR. Section 5 summarizes the related work. Finally, Sect. 6 concludes the paper.

2 DETOUR Architecture Overview

In this section, we firstly give an overview of DETOUR and describe the architecture of OCS, which is the key device to construct DETOUR. Then we give an example to explain the specific broadcast-and-select workflow. Lastly, we analyze the scalability, feasibility and cost of DETOUR.

DETOUR is a distributed optical switching fabric, in which the OCSes are physically connected in 2D-Torus topology and configured by a centralized controller, as shown in Fig. 1. DETOUR overlays above ToR EPSes to construct a flattened and hybrid packet/circuit switched DCN. Each OCS in DETOUR has m ports directly connected with the below EPS, whose up-link ports are equipped with m DWDM transceivers. Thus, the EPSes are logically connected as a m-regular random graph [20] in the optical DCN.

DETOUR is based on the broadcast-and-select optical switching mechanism and uses multi-fibers for space division multiplexing different broadcast signals. For each sender OCS, the signals are statically broadcasted along the same horizontal (west-east) and vertical (south-north) dimensions. For OCSes in different dimensions, the desired signals are selected from the broadcasted signal sets by the crosspoint OCS, and optically forwarded from one dimension to the orthogonal dimension without being relayed by the crosspoint EPS. DETOUR constructs a non-blocking optical switching fabric that supports establishing directly connected optical links between arbitrary EPS pairs. Thus, it reduces the hop counts of EPS and improves network throughput.

Figure 2 illustrates the detailed architecture of OCS. Each OCS comprises of commercial optical components, such as $N \times 1$ Wavelength Selective Switch (WSS), Multiplexer (MUX), Demultiplexer (DEMUX), Optical Splitter, Coupler, Erbium Doped Fiber Amplifier (EDFA).

We firstly give a description about these optical elements. (1) $N \times 1$ WSS takes N fibers with k wavelengths each and outputs a non-interfering subset

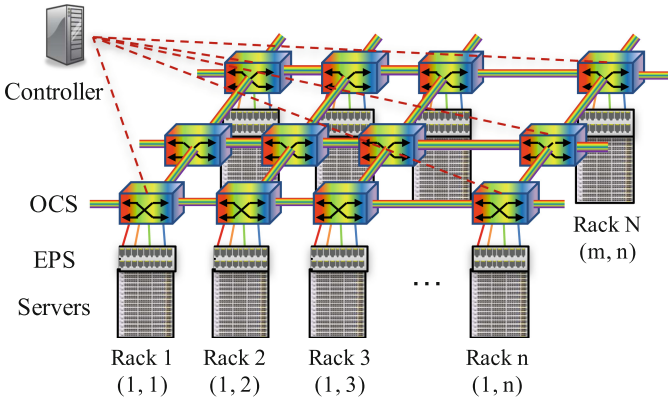

Fig. 1. The high-level architecture of DETOUR

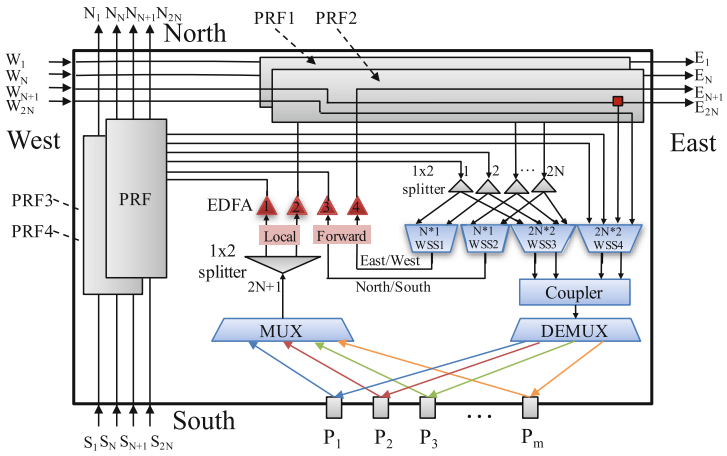

Fig. 2. The architecture of OCS

of the $k \times N$ wavelengths to an output fiber. The measured switching delay of $N \times 1$ WSS is less than 10 ms [25]. $2N \times 2$ WSS is composed of two $N \times 1$ WSSes and has N input ports for each horizontal and vertical dimension in the 2D-Torus topology. To construct a non-blocking optical switching fabric, there are at most $N + 1$ OCSes at each dimension. The option of N is the key factor to determine the number of other components. (2) MUX/DEMUX combines or separates optical signals at different wavelengths. (3) Splitter/Coupler are passive devices that combine or split optical power by a certain ratio. (4) EDFA is used to boost the signals and compensate losses.

In the following, we will detailedly explain the workflow of the specific broadcast-and-select optical switching mechanism.

– *From the Broadcasting View*

For each OCS, it has m ports to connect with the below EPS, and the corresponding EPS has m up-link ports equipped with DWDM transceivers, which have fixed unique wavelengths. The input DWDM signals are multiplexed into a single fiber by MUX. And the MUXed DWDM signals are then equally split into two fibers by a (5:5) optical splitter and unidirectionally transmitted to next neighbor OCSes. For broadcasting the signals, the source MUXed DWDM signals are statically split into two parts at each OCS along the broadcasting path. By using a passive drop-continue splitter, one part is dropped to the WSS for being passed to the local OCS or forwarded to the orthogonal dimension, the other part continues transmitting to the next OCS. As shown in Fig. 3, the multi-coloured line represents that the MUXed DWDM signals from EPS1 are statically and unidirectionally broadcasted to all the other OCSes along the west-east and south-north direction. The red line means that this signal is selected to be forwarded from west-east direction to south-north direction by WSS2. As optical splitter and connector insertion have signal losses, a single stage EDFA is

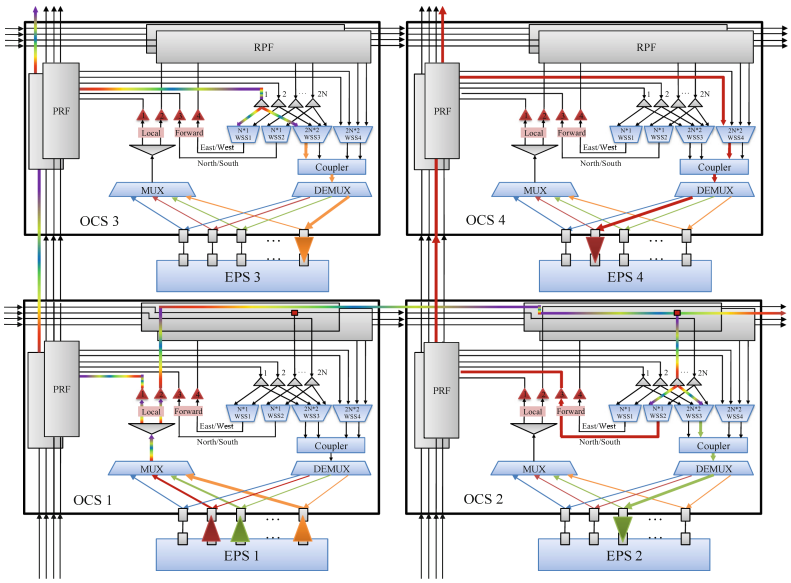

Fig. 3. An example of how DETOUR works (Color figure online)

used to boost the DWDM signals before being broadcasted out. This ensures all the dropped signals' intensity greater than the transceivers' receiver sensitivity, all signals can be recovered correctly.

To ensure the consistency of OCS architecture and simplify the connection between neighbor OCSes, OCS takes advantage of the uniform passive routing fabric (PRF) to reroute the multiple broadcasted signals. PRF also couples with passive drop-continue splitters to drop the broadcasted signals. The ratio of drop and continue is determined by the scale of DETOUR. As shown in Fig. 2, OCS contains 4 PRFs and each dimension has 2 PRFs. PRF1 and PRF3 are used for the signals that are broadcasted from the same dimension. PRF2 and PRF4 are used for the signals forwarded from other dimensions. Focusing on one dimension, for each OCS:

- The source DWDM signals are transmitted out from port E_1 and the source forwarded signals are transmitted out from port E_{N+1}.
- The signals from port W_i $(1 \leq i < N)$ are transmitted out from port E_{i+1} and dropped to the $(N+i)$-th (5:5) splitter.
- The signals from port W_i $(i = N)$ are only dropped to the $2N$-th (5:5) splitter.
- The signals from port W_i $(N+1 \leq i < 2N)$ are transmitted out from port E_{i+1} and dropped to the i-th port of WSS4.
- The signals from port W_i $(i = 2N)$ are only dropped to the $2N$-th port of WSS4.

Thus, OCS has consistency architecture and can be directly connected with neighbor OCSes to construct a 2D-Torus topology. And OCS uses $2N$-fiber optical ribbon to simplify the complexity of cabling, as shown in Fig. 3.

– *From the Selecting View*

As shown in Fig. 2, OCS contains 4 WSSes at the receiver end. $N \times 1$ WSS1 is used to forward the signals from south-north dimension to west-east dimension and $N \times 1$ WSS2 does the opposite. $2N \times 2$ WSS3 is used to select the desired non-conflict DWDM signals coming from other OCSes, which are at the same north-south and west-east dimension with the OCS. The input signals of $2N \times 2$ WSS4 come from OCSes which are not at the same south-north and west-east dimension with this OCS. The signals selected by WSS3 and WSS4 are coupled into one fiber and then de-multiplexed by DEMUX to the up-link ports of the associated EPS.

For each OCS, the input port represents the source OCS of the MUXed DWDM signals. Because the output port of the source DWDM signals gradually increases at each OCS along the broadcasting path, thus:

- The signals from port W_i $(1 \leq i \leq N)$ mean that the source OCS of these signals is the i-th OCS on the west of this OCS. The dropped signals are equally split by the $(N + i)$-th splitter, then transmitted to the i-th port of WSS2 and the $(N + i)$-th port of WSS3.
- The signals from port W_i $(N + 1 \leq i \leq 2N)$ mean that they are forwarded by the $(i - N)$-th OCS on the west of this OCS, and the source OCS of these signals is on the south-north dimension passing the forwarding OCS. Then the dropped signals are transmitted to the i-th port of WSS4.
- The signal from port S_i $(1 \leq i \leq N)$ means that the source OCS of these signals is the i-th OCS on the south of this OCS. The dropped signals are equally split by the i-th splitter, then transmitted to the i-th port of WSS1 and WSS3.
- The signals from port S_i $(N + 1 \leq i \leq 2N)$ mean that they are forwarded by the $(i - N)$-th OCS on the south of this OCS, and the source OCS of these signals is on the west-east dimension passing the forwarding OCS. Then the dropped signals are transmitted to the $(i - N)$-th port of WSS4.

In DETOUR, the controller performs wavelength assignment algorithm, which will be introduced in the next section. Given the wavelength configuration demand, WSSes are configured to pass the desired wavelengths, and block the others. For each destination OCS:

- If the source OCS is in the same south-north or west-east dimension, WSS3 selects the demand wavelengths from the input port associated with the source OCS. As shown in Fig. 3, the optical channel from EPS1 to EPS2 is assigned a green wavelength, and from EPS1 to EPS3 is assigned a orange wavelength. So WSS3 of OCS2 passes the green wavelength from the $(N + 1)$-th port and WSS3 of OCS3 passes the orange wavelength from the 1-th port.

– If the source OCS are not in the same dimension, it needs jointly configure WSSes of the forwarding OCS and destination OCS. WSS1 or WSS2 of the forwarding OCS selects the demand wavelengths from the input port associated with the source OCS and broadcasts it to the orthogonal dimension. Then, WSS4 of the destination OCS passes the demand wavelengths from the input port associated with the forwarding OCS. As shown in Fig. 3, the optical channel from EPS1 to EPS4 is assigned a red wavelength and forwarded by OCS2. So WSS2 of OCS2 passes the red wavelength from the 1-th port and forwards it to the south-north dimension, then WSS4 of OCS4 passes the red wavelength from the 1-th port.

In the following, we will analyze the feasibility, scalability and cost of DETOUR.

We will give a detailed analysis on the feasibility through theory. The key to show the feasibility of DETOUR mainly focuses on two parts: (1) demonstrating the feasibility of the optical components shown in Fig. 2, (2) guaranteeing the receiving signals being correctly identified by DWDM transceivers.

The OCS in DETOUR uses existing commodity optical components, such as $N \times 1$ WSS, EDFA, Splitter, Coupler. And its architecture is similar to that of OvS [25] without introducing any novel optical devices. Zhu et al. have implemented a prototype of OvS and built a small testbed. The key difference between DETOUR and OvS is that DETOUR optically forwards signals to orthogonal dimensions by using the same WSS component. Therefore, the implementation of OvS, as a side effect, has also demonstrated the feasibility of the optical components in DETOUR.

Next, we will explore the Optical Signal Noise Ratio (OSNR) performance and how many times the optical signals can be split while guaranteeing correctness. To support a large scale, OCS adopts a bidirectional design, as shown in Fig. 4. The source signals are broadcasted along each direction and decrease when crossing one OCS. The signal specifications of optical components are listed in Table 2. Let l denote the transmittance $n/(m+n)$, which means the fraction of signal passing and reflecting. For each source OCS:

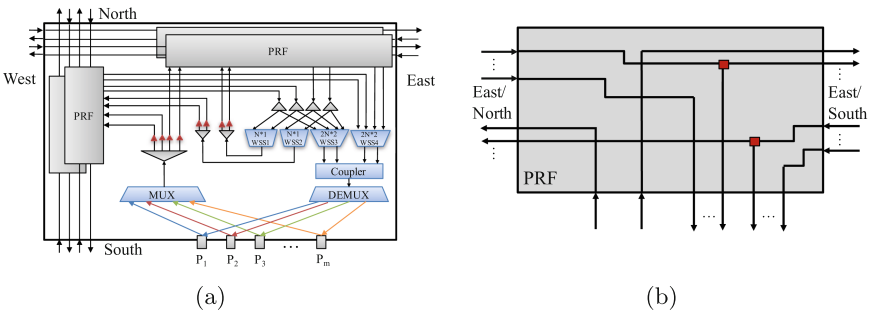

Fig. 4. Bidirectional design

Table 2. Optical component specifications

Contents		Specifications
Transceiver [1]	Output power	$-1 \sim 3$ dBm
	Receiver sensitivity	$-7 \sim -23$ dBm
EDFA [2]	Input power range	$-32 \sim -1$ dBm
	Saturated output power	17.3 ± 0.3 dBm
1×2 splitter (m:n)	Dropped (m) loss	$-10log(m/(m+n))$ dB
	Passed (n) loss	$-10log(n/(m+n))$ dB
Connector loss		1 dB
WSS loss		4 dB
Coupler loss		1 dB
DeMux loss		2.5 dB
1×2 splitter/1×4 splitter		3.5 dB/7 dB

The signal loss S^R_{Loss} for the receiving side of i-th switch is calculated as follows:

$$S^R_{Loss} = -10log(l) * (i-1) - 10log(1-l) + i + 11$$

The signal loss S^F_{Loss} for the forwarding side of i-th switch is calculated as follows:

$$S^F_{Loss} = -10log(l) * (i-1) - 10log(1-l) + i + 11$$

And the forwarding signal loss S^{FR}_{Loss} for the receiving side of i-th switch is calculated as follows:

$$S^{FR}_{Loss} = -10log(l) * (i-1) - 10log(1-l) + i + 7.5$$

The source signals are firstly enhanced by EDFA up to 17.3 dBm. To be correctly recovered by transceivers or enhanced by EDFA, the attenuated signals should satisfy the input constraints of EDFA and the receiver sensitivity of transceivers. Hence the signal loss should under the following constraints: S^R_{Loss}, $S^{RF}_{Loss} \leq 40.3$ dB and $S^F_{Loss} \leq 49.3$ dB. The signal loss is determined by the OCS hops i and the splitter transmittance l. From the related work [6], the number of optical splits increases with transmittance l. When the transmittance is up to 0.9, the number of i equals to 13. Thus, DETOUR can support up to 27 OCSes in one dimension.

Through the above analysis, there are at most $min(27, N+1)$ OCSes at each dimension to construct a non-blocking optical fabric. With state-of-the-art technologies, the option N of a $N \times 1$ WSS can be as high as 32 at reasonable cost [25]. So DETOUR is scalable to connect 27×27 OCSes. A 27×27 2D-Torus network is then achievable to connect up to 729 OCSes. As described in the ITU-T G.692 standard, the C-band can be divided up to 96 wavelengths at

50 GHz channel spacing. By leveraging the standard 50 GHz wavelength spacing of DWDM technology, DETOUR supports up to 729 racks and 69K+ ports with each OCS supporting 96 wavelengths.

DETOUR has the advantage of high performance compared with OvS, but with the cost of modest optical components, e.g. $N \times 1$ WSS, Splitter, EDFA. $N \times 1$ WSS is the most expensive component compared with Splitter, EDFA and Coupler. While, the digital Liquid Crystal (LC) based optical switching technology used in the WSS has been proven to be a reliable and cost-effective technology [25]. In the future, silicon photonics (e.g., matrix switch by ring resonators) can further improve the integration level and reduce the cost.

3 The Control Loop

Inspired by most prior DCN designs [8,9,12,21,25], the hybrid DCN based on DETOUR employs a centralized controller to manage EPSes and OCSes. The controller maintains network information. And it performs demand estimation, wavelength assignment, reconfiguration and so on.

3.1 Traffic Demand Estimation

There exists many traffic demand estimation solutions. For example, the controller can periodically capture snapshots of the overall traffic demand. Hedera [5] and Helios [12] allocate bandwidth of elephant flows by guaranteeing max-min fairness in an ideal non-blocking network. c-Through [21] increases the socket buffer and uses large buffer occupancy to indicate the optical link demand. Moreover, researchers have started to forecast traffic demands of scientific and data-intensive parallel applications from diverse layers (e.g., application layer [22], compiler layer [7,14]). And for clusters that are orchestrated by centralized schedulers (e.g., MPICH2 Hydra, Hadoop YARN), the schedulers orchestrate jobs to compute, storage nodes, and make traffic demand visible.

As the reconfiguration delay of DETOUR is about 10 ms, DETOUR is suitable for the stable or predictable traffic demand which tolerates with the reconfiguration overhead, or DETOUR is preallocated to specific jobs. Overall, DETOUR is proposed to realize a fully reconfigurable interconnection with high scalability, performance and flexibility. And it works as topology-on-demand network resources to match with the upper demand.

3.2 Wavelength Assignment

Given a traffic demand matrix, the controller converts it into wavelength assignments and pushes them into OCSes. The converting algorithm can be accomplished by using weighted b-matching [8], in which b represents the number of ports connected with DETOUR at each ToR EPS. Through the weighted

b-matching algorithm, we get a wavelength demand matrix $G_w(V_w, E_w, \phi_w)$, in which $\phi_w(u, v)$ denotes the number of wavelengths assigned on directed edge (u, v). We need to assign non-conflicting wavelengths to satisfy ϕ_w. Non-conflicting refers that the same wavelengths can not coexist in the same fiber. Due to the specific architecture of OCS shown in Fig. 2, a feasible assignment is that no same wavelengths simultaneously exist in the sending fiber of one OCS, the same as receiving fiber, forwarding fiber from x to y and forwarding fiber from y to x. This problem is equivalent to edge-coloring problem on a multi-graph with extra constraints of the forwarding nodes. And non-conflicting edge coloring of multigraph G_b means that there are no same colors in each source node, destination node, forwarding node f_{xy} and f_{yx}.

Figure 5 illustrates an example of wavelength assignment process. As shown in Fig. 5(a), the OCSes are physically connected in a 3 * 3 2D-Torus topology, and each OCS has 4 unique wavelengths. Figure 5(b) denotes a specified wavelength demand matrix G_w. We transform it to a bipartite multigraph G_b, as shown in Fig. 5(c). The number of edges between OCSes u and v equals to the entry $\phi_w(u, v)$. We proceed to compute a wavelength assignment using existing edge-coloring algorithms of bipartite multigraph [9], as shown in Fig. 5(d), (e). The label f_{xy} on edge (u, v) means that the wavelength is forwarded from x dimension

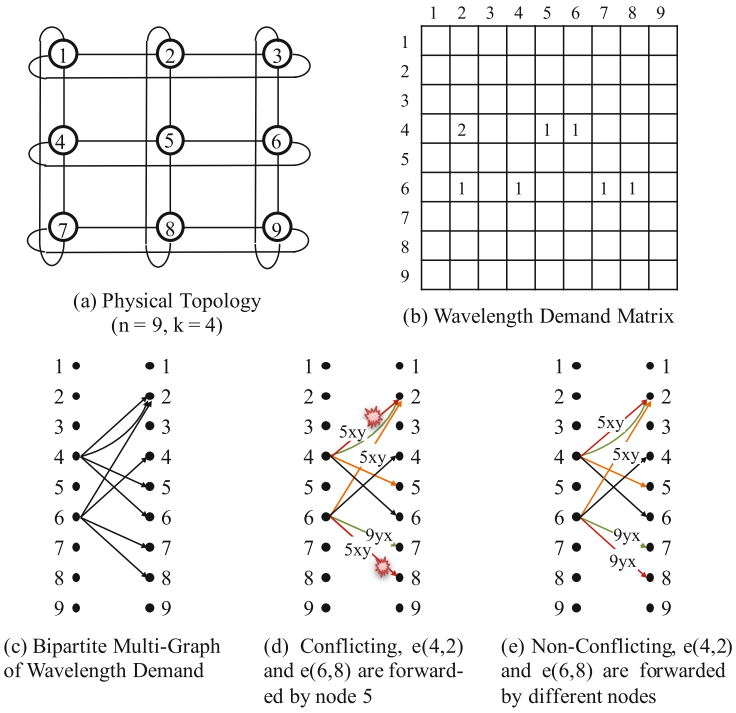

Fig. 5. Wavelength assignment

to y dimension by OCS f, and f_{yx} does the opposite. Edge (4, 2) and edge (6, 8) are conflicting as they are assigned the same color and both forwarded from x dimension to y dimension by OCS 5. To avoid the conflict, we use OCS 9 to forward edge (6, 8), as shown in Fig. 5(e).

Considering the constraint of forwarding OCSes, we cast the wavelength assignment problem into a constrained edge-coloring solution on a bipartite multigraph. König's theorem [23] states that any bipartite graph G has an edge-coloring solution with $\Delta(G)$ (maximal degree) colors. The challenge in our situation is that whether the bipartite multigraph G_b converted from the wavelength demand matrix G_w is $\Delta(G_b)$-colorable. Since $\Delta(G_b) \leq k$, that ϕ_w can always be satisfied if G_b is $\Delta(G_b)$-colorable. We solve this problem by designing a conflict avoiding algorithm which utilizes the properties of DETOUR.

Theorem 1: *Given a wavelength demand matrix $G_w(V_w, E_w, \phi_w)$, we can always satisfy ϕ_w (non-conflicting) using $\Delta(G_b)$ wavelengths.*

The proof is motivated by Fig. 5. For bipartite multigraph G_b, we can always get an edge-coloring solution with $\Delta(G_b)$ matchings, without considering the forwarding constraints. Each source and destination port in matching M are assigned the same wavelength. For each matching M, if we could reassign the conflicting edges' forwarding nodes (Fig. 5(e)) and get a non-conflicting matching, we will finally get $\Delta(G_b)$ matchings and Theorem 1 will be proven.

We prove that a non-conflicting matching M always exists by recursively adjusting the forwarding nodes. The proof procedure is shown in Algorithm 1. $dict_{xy}$ and $dict_{yx}$ record the assigned forwarding nodes and the corresponding edge, and each forwarding node can only be used once for non-conflicting (line 2). The algorithm only considers the edges (u, v) whose source node u and destination node v are not in the same dimension (line 3–4). Edge (u, v) has two forwarding nodes f_x, f_y and only uses one forwarding node at the same time. Initially, the forwarding nodes are not assigned, and the previous edges of M can choose one of the remaining forwarding nodes (line 6–9). When both forwarding nodes of edge (u, v) are assigned, the algorithm recursively (line 11, 13, 25, 32) adjusts forwarding nodes with former conflicting edges until there is an unassigned forwarding node (line 20–23, 28–30). Since the initial edges only use one forwarding node, another forwarding node is unassigned. The algorithm will always recursively find an unassigned forwarding node except one situation. If the algorithm recursively runs to adjust edge (u_1, v_1) and (u_2, v_2). Edge (u_1, v_1) is assigned forwarding node f_x^{12} and another forwarding node is f_y^{12}. By contrast edge (u_2, v_2) is assigned forwarding node f_y^{12} and another forwarding node is f_x^{12}. This will cause a deadlock and can not find a solution. However, this situation will not happen because there are no two edges having the same forwarding nodes in the 2D-Torus topology. Thus, the algorithm can always find an unassigned forwarding node to all the conflicting edges in matching M. Proof of Theorem 1 is completed.

Algorithm 1. Conflict Avoiding Algorithm

Input: Matching M
Output: Forwarding Set $dict_{xy}$, $dict_{yx}$

function CONFLICT_FREE(M)
  $dict_{xy} = \{\}$; $dict_{yx} = \{\}$ /*f: (u, v)*/
  for (u, v) **in** M **do**
    if not $Is\_Same\_Dimension(u, v)$ **then**
      $f_x, f_y = get\_forward(u, v)$
      if f_x **not in** $dict_{xy}$ **then**
        $dict_{xy}[f_x] = (u, v)$
      else if f_y **not in** $dict_{yx}$ **then**
        $dict_{yx}[f_y] = (u, v)$
      else /*conflicting*/
        $Adjust(dict_{xy}, dict_{yx}, u, v, f_x, `xy')$
        or
        $Adjust(dict_{xy}, dict_{yx}, u, v, f_y, `yx')$
  return $dict_{xy}, dict_{yx}$
function ADJUST($dict_{xy}, dict_{yx}, u, v, f, flag$)
  if $flag == `xy'$ **then**
    $(u_1, v_1) = dict_{xy}[f]$
  else $flag == `yx'$
    $(u_1, v_1) = dict_{yx}[f]$
  $f_x, f_y = get\_forward(u_1, v_1)$
  if $flag == `xy'$ **then**
    if f_y **not in** $dict_{yx}$ **then**
      $dict_{yx}[f_y] = (u_1, v_1)$; $dict_{xy}[f] = (u, v)$
      Return
    else
      Adjust$(dict_{xy}, dict_{yx}, u_1, v_1, f_y, `yx')$
      $dict_{yx}[f_y] = (u_1, v_1)$; $dict_{xy}[f] = (u, v)$
  else
    if f_x **not in** $dict_{xy}$ **then**
      $dict_{xy}[f_x] = (u_1, v_1)$; $dict_{yx}[f] = (u, v)$
      Return
    else
      Adjust$(dict_{xy}, dict_{yx}, u_1, v_1, f_x, `xy')$
      $dict_{xy}[f_x] = (u_1, v_1)$; $dict_{yx}[f] = (u, v)$

3.3 Reconfiguration

To instantiate the new topology, the controller needs to configure OCSes of DETOUR and update the flow tables of EPSes. It may lead to network instable during the reconfiguration. We adopt two strategies to minimize the influence on network performance during reconfiguration.

Minimizing Wavelength Shifting. During the process of calculating new wavelength matchings, we utilize current wavelengths distribution and the G_b matchings to calculate new k-perfect matchings. Then we assign colors to the

k-perfect matchings based on Hungarian algorithm, which minimizes the overlap of wavelength shifting to previous assignments. And for each colored matching, we use Algorithm 1 to get the non-conflicting forwarding sets.

Seamless Reconfiguration. From the above minimizing wavelength shifting algorithm, we will get a subset of wavelengths which does not need to adjust. And each EPS also reserves m-k static wavelengths to ensure the network connectivity. So there exists a stable subnetwork during the reconfiguration. To seamlessly reconfigure the network, each EPS maintains two flow tables: $table_{com}$, $table_{mid}$. $table_{com}$ is used for the complete network and $table_{mid}$ is used for the intermediate subnetwork. The controller maintains topology informations and active flow rules for each network.

Before adjusting the optical links, the controller updates flow rules of $table_{mid}$ based on the difference of new and old subnetwork. This way avoids forwarding packets to a dynamical link, which will cause packet loss. During reconfiguration, the controller enables $table_{mid}$ and calculates new flow rules based on the intermediate network. And it also deletes the flow rules affected by the adjusted links. When the configuration of OCSes finished, the controller enables $table_{com}$ and calculates flow rules based on the new network.

4 Evaluation

In this section, we evaluate the performance of DETOUR via flow-level simulation. We first introduce the simulation methodology and then analyze the performance of DETOUR by conducting extensive simulations.

4.1 Simulation Setting

(1) ***Simulator:*** Because existing packet-level simulators (e.g., NS2, NS3) are time consuming to simulate hundreds to thousands servers, and we are more interested in network throughput rather than packet-level behaviors. Therefore, we implemented a event-based flow-level simulator to perform simulations at large scale. The simulator takes flows with start time, size, source server and destination server as input. When the network status changes (e.g., flow arrival, flow departure, EPS and OCS reconfiguration), it updates the rate and remaining size of all active flows. The rate of each active flow is calculated by the progressive filling algorithm [3], which allocates bandwidth satisfying max-min fairness without considering the detailed transport layer protocol behaviors. A flow transmission is finished when the receiver receives all the data. In this simulator, we also realized a centralized controller, which maintains a global view of the network and manages all the EPSes and OCSes. It periodically (0.1 s in our simulation) predicts the traffic demand between ToRs and assigns optical wavelengths to meet the demand. The OCS reconfiguration and controller communication overhead is setted to 10 ms.

(2) ***Topology:*** We compare the performance of DETOUR against Jellyfish [20], OvS [25] and non-blocking optical switching network. Jellyfish is a pure electrical switching network and randomly connects ToR EPSes into a k-regular topology. It has higher bisection bandwidth and lower mean path length over other static network topologies. The typical optical solutions such as MegaSwitch [10], Mordia [17] and OSA [8] all construct non-blocking optical switching networks and support multiple wavelengths, but has limited scalability. We will compare the performance gap between DETOUR and these non-blocking solutions. OvS is a 2D-Torus optical switching fabric similar with DETOUR but it is blocked with limited connections. In this experiment, we simulate a 8 * 8 2D-Torus topology, each EPS has 18 ports with 10 ports connected with servers and the other 8 ports connected to the optical switching networks. We reserve 2 ports static in each EPS and connect the EPSes as base mesh, which ensures the network connectivity.

(3) ***Traffic Patterns:*** We synthetic the following traffic patterns used in [10]:
- *Server-Level Stride.* We index the servers from 1 to 640. In each round, we randomly select the stride k and each servers i talks with $(i + k)$ mod 640.
- *ToR-Level Stride.* We index the ToR from 1 to 64. In each round, we randomly select the stride k, all the servers in ToR i talk to all the servers in ToR $i + k$ mod 64.
- *Random.* In each round, each server in ToR i talks to servers in up to 4 randomly selected ToRs. Each server randomly communicates with other severs. In this pattern, many flows may select the same path, creating sparse bottleneck links.
- *MapReduce-Demand.* We use the Hive/MapReduce trace collected from a 3000-server, 150-rack cluster, which contains many shuffle processes. And we duplicate the traffic demands onto DETOUR using spatial replication.

(4) ***Metrics:*** We evaluate DETOUR from the following aspects. Firsty, we measure the network throughput under the above typical static traffic patterns. Second, we quantify the effect on reducing FCT and energy consumption under dynamic MapReduce traffic patterns. Third, we analyze the network performance on reconfiguration.

4.2 Network Practical Throughput

Figure 6 illustrates the average (max/min) network throughput under server-stride, ToR-stride and random workloads when running 10 instances. From the figure, we find that DETOUR achieves the same performance as non-blocking optical switching networks under all traffic patterns. And it increases the average throughput by 1.34–2.14× and 2.28–5.7× compared to OvS and Jellyfish respectively under all traffic patterns. The reason is that DETOUR can dynamically allocate directly connected links to perfectly match the traffic demand. While OvS needs multiple hops if the demand pair acrosses different dimensions. Forwarding high bandwidth traffic through multiple hops will consume bandwidth per link and incur load on each EPS it traverses.

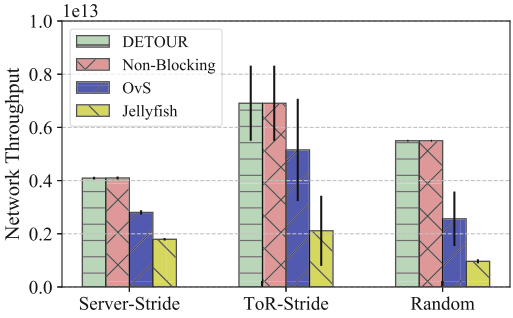

Fig. 6. Network practical throughput

4.3 Overall FCT and Energy Consumption

In this experiment, we evaluate the FCT and energy consumption performance under the dynamic MapReduce traffic pattern. Figure 7(a) and (b) shows the cumulative distribution function (CDF) of FCTs and the overall average FCT, respectively. From Fig. 7, we find that DETOUR achieves the same FCT performance as non-blocking switching networks. The FCTs for all the flows under DETOUR are less than the FCTs under OvS and Jellyfish. And DETOUR reduces the overall average FCT by ∼34% and ∼57% compared with OvS and Jellyfish, respectively. The reason is that the large flows in DETOUR are allocated directly connected links with demanded bandwidth, which reduces the FCT of large flows. Meanwhile, it also reduces the bandwitch preemption between small and large flows, which in trun reduces the FCT of small flows.

Compared with EPS, OCS delivers considerably less energy consumption and avoids unnecessary optical-electrical-optical conversions. The typical per port power values of commercially SFP+ transceivers, EPS switching and OCS switching are 1 W, 8.75 W and 0.14 W, respectively. The energy consumption of each flow is calculated based on per port power and transmission time. Figure 8 illustrates the average energy consumptions under the MapReduce workload.

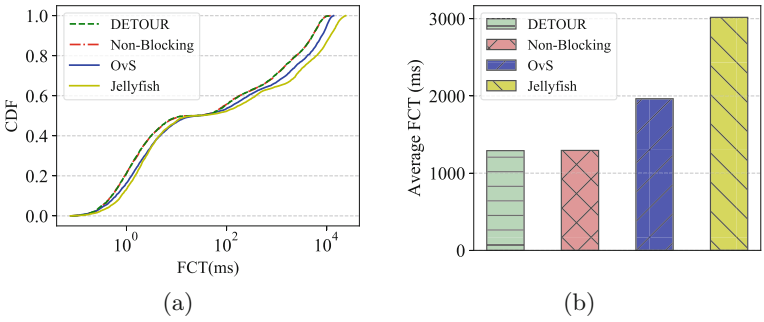

Fig. 7. (a) CDF distribution of FCT and (b) overall average FCT

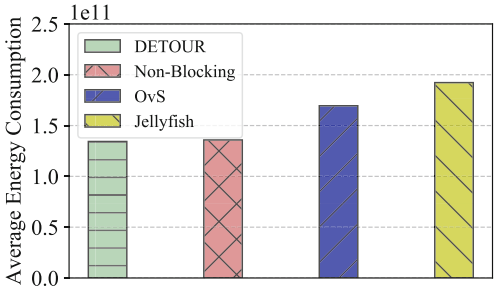

Fig. 8. Average energy consumption

From the figure, we find that DETOUR reduces the overall average energy consumption by ∼21% and ∼30% compared with OvS and Jellyfish, respectively. The reason is that flows in DETOUR traverse through less EPSes and OCSes compared with OvS and Jellyfish.

4.4 Network Performance on Reconfiguration

Figure 9 illustrates the 10 ms reconfiguration impact on network throughput. From the figure, we find that the throughput increases along time with the flows' injection. The throughput of DETOUR and non-blocking optical switching network increase to a relative higher value after each reconfiguration, which are greater than OvS. The reason is that the controller calculates new wavelength assignments to better match with traffic demands in each reconfiguration. During each reconfiguration, the being adjusted optical links are unable to use, and the network bisection bandwidth will temporarily degrades. So we adopt seamless reconfiguration strategies, which ensures the minimal network bisection bandwidth and minimize the reconfiguration impact on small flows.

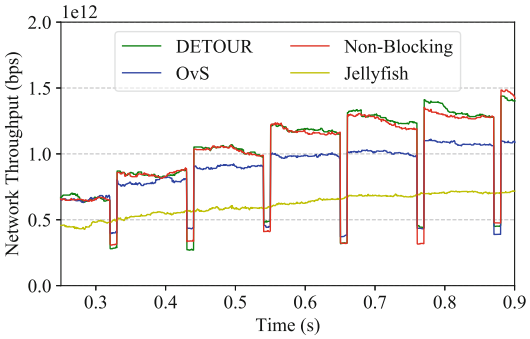

Fig. 9. Network performance on reconfiguration

4.5 Overhead of the Central Controller

The centralized controller maintains network status, estimates traffic demand and allocates wavelengths. In this experiment, we realized the simulator in python language, and we used the existed maximum weight matching and maximum matching algorithms in the networkx package. The time complexity of maximum weight matching and maximum matching are $O(n^3)$ and $O(n)$ respectively, in which n represents the number of racks. We run this simulator on Intel(R) Core(TM) i5-5257U CPU @ 2.70 GHz. We measure the time cost under 10 random seleted traffic patterns. The total time is about 46 ms under 64 switches and 8 wavelengths, in which demand estimation and max-weight matching algorithm consume the dominant. And when the network scales up to ∼700 switches, the runtime increases up to hundreds to thousands microseconds. In order to reduce the impact of the controller overhead to support large scale DCNs, the controller on the one hand can use traffic predicting mechanism, on the other hand can increase the demand estimation period. Moreover, our demand estimation and wavelength reconfiguration algorithms are adopted from Hedera [5] and WaveCube [9] respectively, which has been optimized to consume less than 100 ms for large data centers via parallelization. This means that there is a large room to speed up controller algorithms with advanced technologies to support large scale DCNs.

5 Related Work

Our work is mostly inspired by prior solutions on reconfigurable DCN. We summarize the existing reconfigurable techniques by three categories: Optical Circuit Switching (OCS), 60 GHz Wireless and Free-Space Optics (FSO).

Most solutions (e.g., c-Through [21], Helios [12], and OSA [8]) rely on MEMS-based optical switches, which has high reconfiguration latency and low port counts. They leverage a single MEMS-based optical switch to establish optical links between ToR EPSes. The DCN scalability is limited by the low port density of MEMS. Wavecube [9] removes the core MEMS and connects WSSes in mesh topology. Although Wavecube is scalable to support unlimited racks, the network diameter increases with the scalability as links are only reconfigured between neighbors. Mordia [17], MegaSwitch [10] and OvS [25] are based on the broadcast-and-select mechanism. Mordia takes microseconds switching technologies and establishes optical links with time-sharing. This is not efficient for high-out and stable traffic patterns. OvS uses multi-fiber multiplexing and connects OCSes into 2D-Torus topology. But it only supports establishing directly optical circuits between OCSes in the same dimension.

Flyway [14] was firstly proposed to augment the traditional data center with 60 GHz wireless devices, which relieves hotspot traffic. In Mirror [24], beams were bounded off the data center ceiling to eliminate the line-of-sight constraint. Unlike optical technology, 60 GHz wireless suffers from limited throughput and low distance, which is hard to use in large scale data centers.

Firefly [15] equips ToR EPSes with free-space optics and uses Galvo or switchable mirrors to dynamically establish optical links. ProjecToR [16] combines digital micromirror device (DMD) and mirror assembly to construct a high-fanout free-space topology. However, the beam of FSO is narrow and susceptible to interferences.

6 Conclusion

We presented DETOUR, a large-scale non-blocking optical data center fabric, which supports up to 700+ racks and 69K+ servers. We designed a recursive wavelength assignment algorithm based on the architecture of DETOUR. And We also implemented a flow-level simulator and realized the control algorithms. Extensive evaluation results show that DETOUR delivers high performance comparable to a non-blocking switching fabric. It outperforms up to 2.14× higher throughput, reduce 34% FCT and 21% energy consumption compared with the state-of-the-art works.

Acknowledgments. The work is supported by the project of National Key Research and Development Program of China under Grant No. 2016YFB0200400, and FANEDD under Grant No. 201450.

References

1. Cisco DWDM SFP+ module. http://www.cisco.com/c/en/us/products/collateral/interfaces-modules/dwdm-transceiver-modules/data_sheet_c78-711186.html
2. Cisco ONS15501 erbium doped fiber amplifier. http://www.cisco.com/en/US/products/hw/optical/ps2011/products_data_sheet09186a008008870d.html
3. Progressive filling algorithm. https://en.wikipedia.org/wiki/Max-min_fairness
4. Al-Fares, M., Loukissas, A., Vahdat, A.: A scalable, commodity data center network architecture. In: ACM SIGCOMM (2008)
5. Al-Fares, M., Radhakrishnan, S., Raghavan, B., Huang, N., Vahdat, A.: Hedera: dynamic flow scheduling for data center networks. In: NSDI (2010)
6. Bao, J., Dong, D., Zhao, B., Luo, Z., Wu, C., Gong, Z.: FlyCast: free-space optics accelerating multicast communications in physical layer. In: ACM SIGCOMM (2015)
7. Barker, K.J., Benner, A., Hoare, R., Hoisie, A., Jones, A.K., Kerbyson, D.K., Li, D., Melhem, R., Rajamony, R., Schenfeld, E., et al.: On the feasibility of optical circuit switching for high performance computing systems. In: IEEE SC (2005)
8. Chen, K., Singla, A., Singh, A., Ramachandran, K., Xu, L., Zhang, Y., Wen, X., Chen, Y.: OSA: an optical switching architecture for data center networks with unprecedented flexibility. In: NSDI (2012)
9. Chen, K., Wen, X., Ma, X., Chen, Y., Xia, Y., Hu, C., Dong, Q.: WaveCube: a scalable, fault-tolerant, high-performance optical data center architecture. In: IEEE INFOCOM (2015)
10. Chen, L., Chen, K., Zhu, Z., Yu, M., Porter, G., Qiao, C., Zhong, S.: Enabling wide-spread communications on optical fabric with megaswitch. In: NSDI 2017, Boston, MA, pp. 577–593 (2017)

11. Chowdhury, M., Stoica, I.: Coflow: a networking abstraction for cluster applications. In: ACM HotNets (2012)
12. Farrington, N., Porter, G., Radhakrishnan, S., Bazzaz, H.H., Subramanya, V., Fainman, Y., Papen, G., Vahdat, A.: Helios: a hybrid electrical/optical switch architecture for modular data centers. In: ACM SIGCOMM (2010)
13. Guo, C., Lu, G., Li, D., Wu, H., Zhang, X., Shi, Y., Tian, C., Zhang, Y., Lu, S.: BCube: a high performance, server-centric network architecture for modular data centers. In: ACM SIGCOMM (2009)
14. Halperin, D., Kandula, S., Padhye, J., Bahl, P., Wetherall, D.: Augmenting data center networks with multi-gigabit wireless links. In: ACM SIGCOMM (2011)
15. Hamedazimi, N., Qazi, Z., Gupta, H., Sekar, V., Das, S.R., Longtin, J.P., Shah, H., Tanwer, A.: Firefly: a reconfigurable wireless data center fabric using free-space optics. In: ACM SIGCOMM (2015)
16. Monia (Manya), G., Ratul, M., Amar, P., Nikhil, R., Gireeja, R., Jana, K.: Projector: agile reconfigurable data center interconnect. In: ACM SIGCOMM (2016)
17. Porter, G., Strong, R., Farrington, N., Forencich, A., Chen-Sun, P., Rosing, T., Fainman, Y., Papen, G., Vahdat, A.: Integrating microsecond circuit switching into the data center. In: ACM SIGCOMM (2013)
18. Roy, A., Zeng, H., Bagga, J., Porter, G., Snoeren, A.C.: Inside the social network's (datacenter) network. In: ACM SIGCOMM (2015)
19. Singh, A., Ong, J., Agarwal, A., Anderson, G., Armistead, A., Bannon, R., Boving, S., Desai, G., Felderman, B., Germano, P., et al.: Jupiter rising: a decade of clos topologies and centralized control in Google's datacenter network. In: ACM SIGCOMM (2015)
20. Singla, A., Hong, C.Y., Popa, L., Godfrey, P.B.: Jellyfish: networking data centers randomly. In: NSDI (2012)
21. Wang, G., Andersen, D.G., Kaminsky, M., Papagiannaki, K., Ng, T., Kozuch, M., Ryan, M.: c-Through: part-time optics in data centers. In: ACM SIGCOMM (2010)
22. Wang, H., Chen, L., Chen, K., Li, Z., Zhang, Y., Guan, H., Qi, Z., Li, D., Geng, Y.: Flowprophet: generic and accurate traffic prediction for data-parallel cluster computing. In: IEEE ICDCS (2015)
23. Wikipedia: König's theorem (graph theory) – wikipedia, the free encyclopedia (2015)
24. Zhou, X., Zhang, Z., Zhu, Y., Li, Y., Kumar, S., Vahdat, A., Zhao, B.Y., Zheng, H.: Mirror mirror on the ceiling: flexible wireless links for data centers. In: ACM SIGCOMM (2012)
25. Zhu, Z., Zhong, S., Chen, L., Chen, K.: Fully programmable and scalable optical switching fabric for petabyte data center. Opt. express **23**(3), 3563–3580 (2015)