# HFOS $_L$

Telecommunication Systems (2022) 80:397–411
https://doi.org/10.1007/s11235-022-00905-2

# HFOS$_L$: hyper scale fast optical switch-based data center network with $L$-level sub-network

Elham Khani[1] · Shaahin Hessabi[1] · Somayyeh Koohi[1] · Fulong Yan[2] · Nicola Calabretta[2]

Accepted: 3 April 2022 / Published online: 17 May 2022


**Abstract**
The ever-expanding growth of internet traffic enforces deployment of massive Data Center Networks (DCNs) supporting high performance communications. Optical switching is being studied as a promising approach to fulfill the surging requirements of large scale data centers. The tree-based optical topology limits the scalability of the interconnected network due to the limitations in the port count of optical switches and the lack of optical buffers. Alternatively, buffer-less Fast Optical Switch (FOS) was proposed to realize the nanosecond switching of optical DCNs. Although FOSs provide nanosecond optical switching, they still suffer from port count limitations to scale the DCN. To address the issue of scaling DCNs to more than two million servers, we propose the hyper scale FOS-based $L$-level DCNs (HFOS$_L$) which is capable of building large networks with small radix switches. The numerical analysis shows $L$ of 4 is the optimal level for HFOS$_L$ to obtain the lowest cost and power consumption. Specifically, under a network size of 160,000 servers, HFOS$_4$ saves 36.2% in cost compared with the 2-level FOS-based DCN, while achieves 60% improvement for cost and 26.7% improvement for power consumption compared with Fat tree. Moreover, a wide range of simulations and analyses demonstrate that HFOS$_4$ outperforms state-of-art FOS-based DCNs by up to 40% end-to-end latency under DCN size of 81920 servers.

**Keywords** Fast optical switch · Optical data center network · Low latency interconnection · Scalable data center network

## 1 Introduction

Enforced by emerging cloud computing services, web-based applications, and Internet of Things (IoT), DCNs experience tremendous growth of global IP traffic. Statistically, it is estimated that the Data Center (DC) IP traffic will reach to around 20 zettabyte by the end of 2021 [1].

Underlying interconnection network plays a key role in scaling the DCs to guarantee performance of traffic-boosting applications [2]. Therefore, it is of crucial importance to design an interconnection network which is highly scalable, power consumption and cost efficient. Moreover, interconnection network must meet the stringent requirements of current DC traffic in terms of high bandwidth and low latency.

The traditional multi-tier DCNs utilizing commodity electrical switches consume vast amount of power and cost to support the huge bandwidth requirements (>10Gbps) of the hundreds of thousands of communicating servers. One alternative is to deploy high-radix electrical switches to flatten the architecture. However, the implementation of high-radix electrical switches at high data rates is restricted by the limited I/O bandwidth of the Application Specific Integrated Circuit (ASIC) caused by limited Ball Grid Array (BGA) density [3]. Moreover, data center traffic roughly doubles every year, while Moore's law is approaching the physical limits [4]. Consequently, electrical switching will not be capable of supporting efficient bandwidth for modern cloud applications.

To overcome the scaling limitations of DCNs, optical technology has been introduced as a promising approach [5,6]. Compared with its electrical counterpart, optical switching technology is capable of providing data rate/format transparency and high capacity [7]. Optical architectures eliminating power hungry optical-electrical-optical conversion have been proposed to mitigate the high power consumption and large latency of electrical architectures [8]. Wavelength Division Multiplexing (WDM) technology leverages parallelism through transmitting more than a hundred of wavelength

✉ Shaahin Hessabi
hessabi@sharif.edu

[1] Sharif University of Technology, Tehran, Iran

[2] Eindhoven University of Technology, Eindhoven, The Netherlands

channels at the data rate of 100Gbps per each wavelength channel [7].

Micro Electro-Optical Systems (MEMS) switches are reconfigurable devices which employ small mirrors. The reconfiguration delay of 3D-MEMS is on the order of milliseconds which is the time to turning the small mirrors to reflect light to the desired output [9]. MEMS are mainly employed as a core switch in Optical Circuit Switching (OCS) architectures [10]. Arrayed Waveguide Grating (AWG) are passive optical components which are made of multiple waveguide tapes [11]. AWG can be employed in conjunction with tunable lasers to implement low-power and high speed DCNs [12]. This component can also be used in core switches of optical packet switches. Semiconductor Optical Amplifiers (SOA) are optical amplifiers with fast switching time on the order of nanoseconds [13]. Wavelength Selective Switches (WSS) are 1–N optical switches which distribute incoming wavelengths among N output ports [13]. Wavecube [14] is an optical DCN that utilizes WSS to build a dynamic topology. Tunable transceivers (TRX) are employed to transmit and receive data. Commercially available TRXs are capable of tuning over C-band wavelength on the order of tens of milliseconds [15].

Opera [16] is an optical network with expander graph based topology designed to serve latency-sensitive portion of the traffic. It focuses on finding the shortest path for delay sensitive flows at the cost of the reconfiguration time on the order of millisecond. Opera can provide direct path for bulk traffic and immediate path for latency-sensitive path. However, the throughput of the network decreases under skewed and permutation workloads. In [17], the authors proposed a flexible topology optical switching solution for DCN to decrease the reconfiguration time of the optical circuits. The authors also evaluated the parameters that influence network performance such as reconfiguration period and controller delay. However, the performance of the proposed solution degrades with the nodes' distance and the number of circuit chains in the network. FlexNet [18] proposes an optical switch architecture for DCN to improve optical links utilization. FlexNet utilizes multiple optical MEMS switches in one layer to interconnect TORs together. This architecture lacks scalability since network scale is limited by the port counts of MEMS switches. Optical Packet Switching (OPS) is capable of switching data in nanosecond scales [12,19,20]. Sirius [4] employs optical passive components to implement an optical DCN. It establishes a connection through AWGs and transceivers with tunable lasers to realize fast optical switching. The proposed architecture has a single-layer gratings with very low power consumption. Modulated wavelengths carry the data, and determine the destination address through a static schedule. The authors designed and fabricated a custom tunable laser chip to implement picosecond tunable lasers. Although the cost and power consumption of Sirius is lower than the electrical DCN, it is still expensive for large network sizes due to the number of arrayed waveguide gratings and the tunable lasers. HyFabric [21] is a hybrid DCN that utilizes electrical packet switching and optical switching to interconnect TORs. Simulation analysis of HyFabric shows less cost and power consumption than similar hybrid DCNs

Table 1 shows the summary of comparative analysis of optical DCNs which are recently proposed. In this table, the previously proposed architectures are compared in terms of scalability, cost, and power consumption. The latency of the mentioned architectures represents the latency of their optical plane. The connection of the networks can be either packet switching or circuit switching, which are mentioned in Table 1.

Opera can be realized using commodity optical components with medium cost and low power consumption. On the other hand, Opera functions better for small network sizes since network scalability is limited by large number of required routing state (expander graphs). In [17], authors employ commodity optical circuit switches which have low cost and power, while they have limited scalability and high delay. On the other hand, FlexNet has faster switching of millisecond at higher cost and limited scalability. Sirius achieves low latency interconnection. However, the cost of its architecture is high since it is based on the tunable lasers and high-radix gratings. Scalability of Sirius is limited by the radix of its single AWG switch. HyFabric has a lower implementation cost for its optical plane while its microsecond delay is against high fan-out of emerging applications of data center traffic. It is worth noting that all the architectures in Table 1, except HyFabric, uses all-optical technology. HyFabric employs hybrid of electrical and optical technology. Consequently, the requirement of optical and electrical plane imposes high cost and power consumption for the network.

One of the main challenges of the aforementioned DCN architectures is their scalability. Consequently, several optical switching architectures have been proposed to support large size DCNs [7,22], [23–25]. However, a major challenge when scaling out the optical DCN is the limited port counts of FOSs. High-radix FOSs are prevented from the practical implementation, due to quadratic increase of components with respect to the FOS radix. Employing more switching levels allows the architecture to scale out with low-radix FOSs. Meanwhile, adding extra levels results in more number of optical switching components. Hence, there is a trade off between the number of optical switching levels and the optical switch radix.

In this paper, we address the DCN scalability issue by utilizing multiple levels of fast optical switches by extending the FOS-based architecture of [26] which was built upon FOS switch. The target architecture is comprised of

**Table 1** Comparative analysis of the recent architectures for optical DCNs. The column Commodity Switch shows whether the architecture uses the off-the-shelf components or proof of concept components.

| Architecture | Scalability | Technology | | Power | Cost | Switching time | Topology | Commodity switches |
|---|---|---|---|---|---|---|---|---|
| | | Hybrid E/O | All-optical | | | | | |
| Opera [16] | Low | | Yes | Low | Medium | $\mu s$ | Two-tier leaf-spine | Yes |
| Sirius [4] | Limited | | Yes | Low | High | $ns$ | Single layer | No |
| HyFabric [21] | Low | Yes | | Low | High | $ms$ | Crossbar switch | No |
| Johannes et al. [17] | Low | | Yes | Low | Low | $ms$ to s | Single optical switch | Yes |
| FlexNet [18] | Limited | | Yes | Low | High | $ms$ | Single layer | Yes |

multiple-levels of FOS switches which work in parallel. The parallel multiple-level enables the realization of hyper-scale DCNs. The hyper-scale FOS-based $_L$-level DCN (we named it HFOS$_L$) exploits the fast optical packet switches with port count independent switching time of nanoseconds scale ($20ns$). In our previous work [27], we numerically analyzed the cost and power consumption of HFOS$_L$ and compared the DCNs with different levels. As verified by the numerical analysis, there is a slight difference on the cost of HFOS$_L$ for various scales, while the cost of HFOS$_4$ is marginally less than that of HFOS$_2$ (OPSquare [26]) and HFOS$_3$ (FOS-cube [28]) for large DCN sizes (>100K servers). Besides, the cost of HFOS$_4$ is less than that of HFOS$_5$ for any DCN sizes. The power consumption analysis of different levels showed that power consumption of HFOS$_{j+1}$ is larger than HFOS$_j$ for $j$>3.The four-level architecture allows the realization of highly scalable DCN while achieves low cost and power consumption. We investigate the performance of HFOS$_4$ building with low-radix FOS. HFOS$_4$ achieves $8\mu$s and $4\mu$s server-to-server and network latency, respectively, at the offered load of 0.4 under the network size of 81920 servers. Furthermore, the end-to-end latency of HFOS$_4$ increases at a slower pace than that of similar architectures with same size. It must be noted that the large path diversity of the proposed DCN facilitates the implementation of load balancing algorithm [29] in the DCN.

The rest of the paper is organized as follows. In Sect. 2, the building blocks of HFOS$_4$ are described. Section 3 presents HFOS$_4$ architecture and related algorithms. Section 4 explains the simulation environment and parameters. Section 5 reports and discusses network performance of HFOS$_L$. The numerical analysis of cost and power consumption of HFOS$_4$ are presented and compared with Fat-tree in Sect. 6. Finally, Sect. 7 concludes the paper.

## 2 Preliminaries

In this section, the building blocks of HFOS$_L$ and their functionalities are explained. Two main building blocks of HFOS$_L$ are Top-of-Rack (TOR) and FOS switches which are described in this section.

### 2.1 TOR building blocks

TOR is an Ethernet switch which receives packets from the servers inside the rack and send them to the destination servers. In this work, TOR connects to $K$ servers with $K$ 10Gbps link and connects to FOS switches through optical interfaces. Figure 1 shows the structure and internal components of a TOR. The traffic arriving from servers is directed to the TOR's head processor to check the packet header (destination). Based on the destination server, TOR forwards the

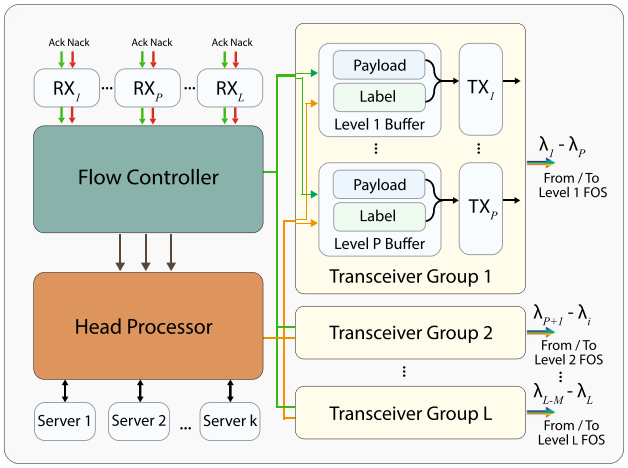


**Fig. 1** TOR building blocks

intra-TOR packets into intra-TOR buffer queues while inter-TOR packets are directed to optical interfaces. In this paper, we focus on the inter-TOR connections.

TOR assigns different wavelengths to packets and aggregates the packets based on the destination rack before sending them out of the rack. The optical interfaces of TOR contain WDM transceivers with the dedicated electrical buffers. In HFOS$_L$, each optical interface is connected to one of the $L$ level FOSs through $t_i$ optical transceivers. The number of dedicated transceivers to each FOS level is determined based on the volume of exchanged traffic among those levels. The number of transceivers can be flexibly allocated to achieve the required oversubscription ratio.

In the routing table of TOR, destination TORs are divided into groups of size $N/t_i$. Packets are modulated on different wavelengths based on their destinations. Accordingly, each transmitter is responsible for sending optical packets to a specific group of TORs.

The head processor distributes the inter-TOR packets among the $L$ optical interfaces (Transceiver Group $i$), based on the destination of the packets. Transceiver Group $i$ connects TOR to FOS level $i$ ($1 \leq i \leq L$). Then, packets are stored in corresponding transceiver electronic buffers to be sent.

As depicted in Fig. 1 TOR is also connected to FOSs through optical links, to receive ACK/NACK signals from FOSs. TOR sends the optical packets to corresponding FOS switches (based on the destination TOR) and keeps the copy of sent packets. When the packet is successfully sent by the FOS switch, an ACK signal is issued by the FOS and is sent back to the sending TOR. Upon receiving ACK signal from FOS, TOR releases that copy of packet from its buffers. In the case that packet is blocked in FOS (due to contention), the FOS's flow control generates a NACK signal and sends it to the TOR to retransmit the packets. The flow control of TOR handles the received packet to the TOR. It forwards the

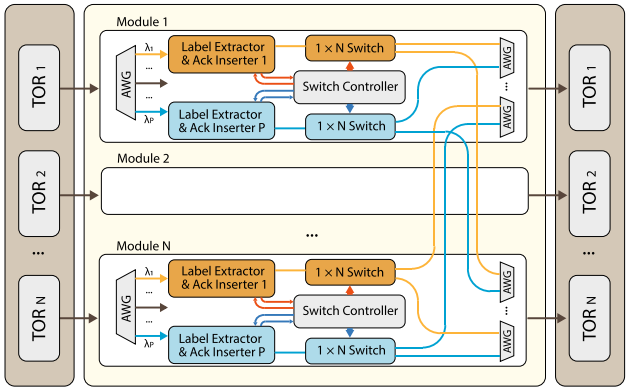


**Fig. 2** The schematic of FOS radix-$N$ [26]

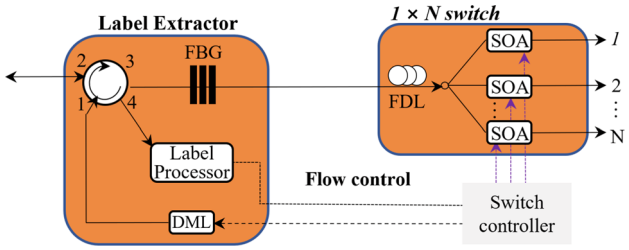


**Fig. 3** The schematic of label extractor and photonic switch [26]

received packets to the head processor unit to determine if they are arrived to the destination or required to be forwarded to the next TOR.

## 2.2 Schematic of FOS switch

Figure 2 shows the schematic diagram of FOS radix-$N$ architecture. FOS radix-$N$ has $N$ input/output ports to receive/send WDM packets from $N$ connecting TORs in parallel.

FOS is a buffer-less broadcast and select optical switch consisting of $N$ identical parallel modules. The AWG, at input/output of module, multiplex/de-multiplex wavelength channels receiving/sending from TORs. The received optical packets are forwarded to the label extractor. The payload is directed to SOA based autonomously controlled $1 \times N$ photonic switch unit. The SOA gates amplify the optical signals and compensate the losses caused by splitting of the signals. Besides, nanoseconds switching time of SOA enables nanosecond realization of optical switching in FOS switches. The label extractor extracts the optical label from packet. The structure of label extractors and $1 \times N$ photonic switch is depicted in Fig. 3.

The label extractor separates the in-band RF tone label and payload by fiber Bragg grating (FBG). Then the label processor processes the label and send it to switch controller. Based on the label information, switch controller checks the packet's destination and the corresponding output port. In this

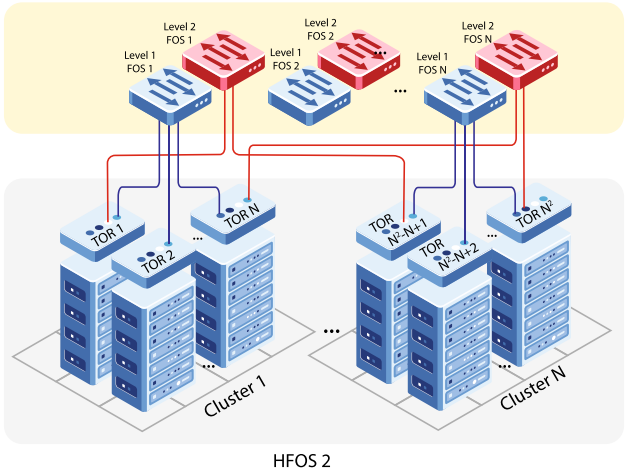


**Fig. 4** The HFOS$_2$ (OPSquare) architecture

stage, contention is detected by switch controller and packets with lower priority are blocked. Switch controller also configures $1 \times N$ photonic switch unit to forward packets with higher priority to the outputs. The priority in this paper is based on FIFO approach. The ACK/NACK flow control scheme is adopted in FOS to handle the contention of optical packets [30]. The ACK/NACK signals are generated by switch controller and are sent back to corresponding TORs to update it about blocking/sending of packet respectively. Finally, the optical payload is directed to the output port. As shown in Fig. 2, the contention resolution among $N$ input ports is performed in a distributed manner. The switching time of FOS is radix-independent with respect to the high latency caused by the central controlling mechanism. It is worth mentioning that FOSs of 64, 48, 32, 24, 16, 8 ports are feasible to implement. Details on the FOS architecture can be found in [26].

## 3 HFOS$_L$ DCN system operation

In this section the high level view of the HFOS$_L$ DCN architecture is described. Next, the pattern of the interconnection of TORs and FOSs are presented. Finally, the routing algorithm of HFOS$_L$ is explained.

### 3.1 HFOS$_L$ DCN architecture

HFOS$_L$ is a recursive architecture that is constructed by parallel levels of FOS switches to interconnect TORs. Each TOR is connected to one specific FOS in each level via WDM links. Hence, for $L$ level FOSs (HFOS$_L$) every TOR is connected to $L$ FOSs in $L$ different levels. Assuming identical radix-$N$ FOSs in all levels, HFOS$_L$ DCN accommodates $N^L$ TORs. More generally, the total number of interconnected TORs is computed as $\prod_{i=1}^{L} N_i$, where $N_i$ is the FOS radix of the $i$-th level. Figure 4 illustrates the architecture of HFOS$_2$ (OPSquare) as the first building block of HFOS$_L$ topology.

Given $N^2$ TORs, TORs are grouped in clusters of size $N$ by the level 1 FOSs, so $N$ clusters of size $N$ TOR are created. The level 2 FOSs interconnects these clusters together. The details and algorithm of interconnecting TORs to FOSs are presented in Sect. 3.2.

Figs. 5 and 6 show the abstract and in-depth schematic of 4-level FOS-based architecture (HFOS$_4$) containing $N^4$ TORs.

In the first level, TORs are partitioned to clusters of size $N$ TORs and are directly interconnected to each other via $N$-radix FOSs of level 1. As a result, $N^3$ clusters of size $N$ are created. The algorithm of interconnecting and grouping of TORs in all levels is presented in Sect. 3.2. In the second level, clusters are grouped in super clusters of size $N^2$, i.e. groups of $N$ clusters create super clusters of size $N^2$. That are connected via $N$-radix FOSs of level 2. Now, DCN is partitioned to $N^2$ super clusters of size $N^2$. In the third level, super clusters are divided to hyper clusters of size $N^3$ connecting through the $N$-radix FOS of level 3. In this way, $N$ hyper clusters of size $N^3$ are created. At the level 4, hyper clusters are interconnected through $N$-radix FOSs level 4. To this end, 4 levels of FOSs provide all-to-all connection among $N^4$ TORs (Sect. 3.2 explains the algorithm of interconnecting TORs to FOSs in details). In general, $N$-radix FOSs of level $i$, interconnect sub-clusters of size $N^{i-1}$ which were created in level $i$-1. As mentioned before, for

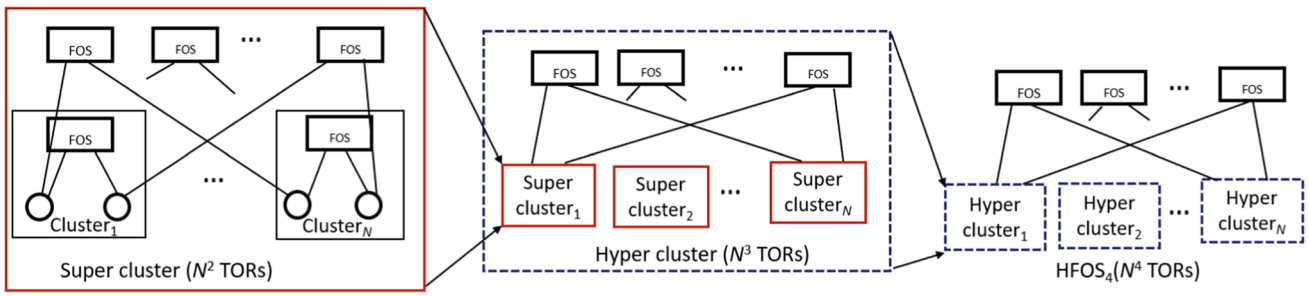


**Fig. 5** The general view of HFOS$_4$ architecture

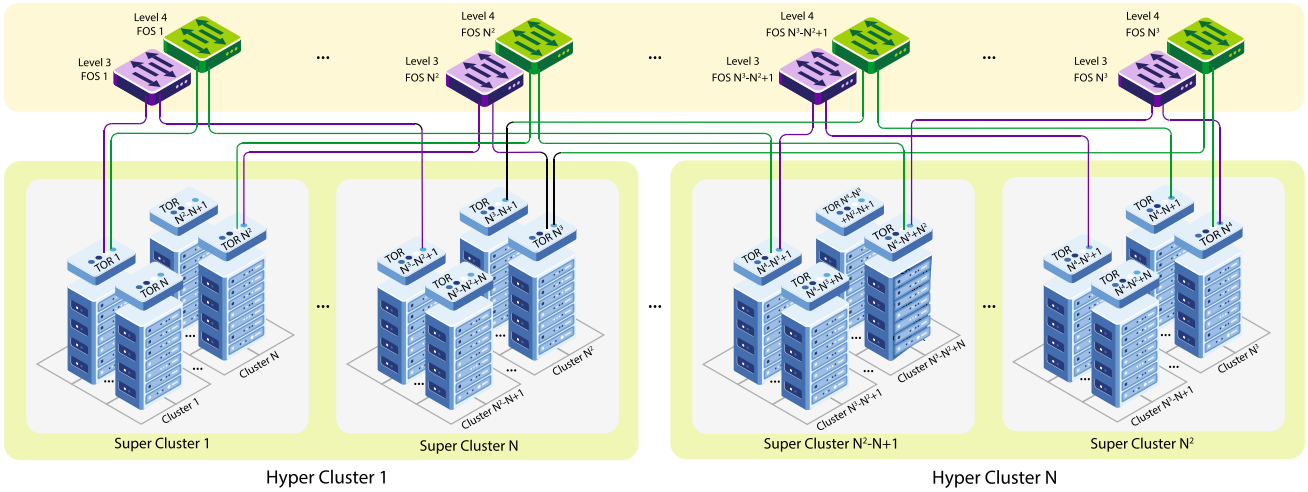


**Fig. 6** The $HFOS_4$ in-depth view of the architecture

FOSs of the same radix, the $HFOS_L$ contains $N^L$ TORs. Given the same number of TORs, $HFOS_L$ with more levels (hence more number of FOSs) requires fewer-radix FOSs, while fewer-levels $HFOS_L$ (hence fewer number of FOSs) requires higher-radix FOSs. Although fewer-radix FOSs are less expensive and power consuming, the more number of FOSs on the other hand may result in more total amount of cost and power consumption. In this paper, we investigated $HFOS_L$ with various $L$ to find the $L$ which is the most efficient in terms of cost, power consumption while maintains the network performance.

### 3.2 $HFOS_L$ interconnection rule

The interconnection rule is shown as follows: the $i$-th FOS in level-2 interconnects the $i$-th TOR of each cluster while $1 \leq i \leq N$. Similar to the interconnection pattern of level-2, the $j$-th FOS in each level-4 connects the $j$-th TORs of each level-3. In this way, level-3 clusters are interconnected to each other through the third level network. The $k$-th FOS of the level-4 network connects the $k$-th TORs of each level-4 cluster. In general, the $m$-th FOS of the $n$-th level network interconnects the $m$-th TOR in $HFOS_{n-1}$ while $2 \leq n \leq L$ and $L$ is the number of network level which is 4 in the case of $HFOS_4$. In $HFOS_L$, the total number of shortest paths between a pair of communication TORs located in different $HFOS_{L-1}$ could be $L!$ at most. The path diversity of $HFOS_4$ allows TORs to be interconnected through diverse routed paths, which implies that in the case of a path failure there will be alternative paths to connect a source TOR to the destination TOR. Moreover, path diversity of the network facilitates traffic engineering techniques and improves network performance. It is worth mentioning that in $HFOS_L$, the communication hops between nodes are independent of the number of servers inside DCN. The small hop count of the network routing implies low propagation delay. Considering the mentioned interconnection pattern, utilizing $N$-radix FOSs and four-level network, $HFOS_4$ sizes to $N^4$ TORs. As an illustration, 8-radix FOSs can build a DCN of 4096 TORs, while 8-radix FOSs can interconnect only 512 TORs and 64 TORs in FOScube and OPSquare, respectively. Considering that $HFOS_4$ can be built with practical low-radix FOSs which have low power consumption and cost, $HFOS_4$ outperforms $HFOS_2$ and HFOS3 in terms of scalability, cost, and power consumption efficiency.

### 3.3 Routing mechanism of $HFOS_4$ network

Assuming identical FOSs of radix-$N$ in all levels, $HFOS_L$ DCN accommodates $N^L$ TORs. In general, the total number of interconnected TORs is $\prod_{i=1}^{L} N_i$ , where $N_i$ is the FOS radix of the $i$-th level. Considering each rack contains $K$ servers, DCN supports $K \times N^L$ servers. Given a number of $K$=40 servers per rack, 16-radix FOS, and $L$=4 levels, $HFOS_L$ supports 2,621,440 servers. Given interconnection rules in Sect. 3.2, each TOR can be indexed from 1 to $N^4$, and servers are indexed from 1 to $K \times N^4$, where $K$ denotes the number of servers inside one rack. Take the index of the TOR, where the server is located, as $i$, and denote the index of the cluster comprising the TOR as $c$. Similarly, take the index of the super cluster and hyper cluster, where the server is located, as $s$ and $h$, respectively. Then the TOR index $T$ can be calculated using a polynomial function as shown in (1).

$$T(N) = h \times N^3 + s \times N^2 + c \times N + i \tag{1}$$

On the contrary, given the TOR index $T$, we can immediately obtain the value of $h$, $s$, $c$, and $i$ with the following

equations (2) to (5).

$$h = \lfloor T/N^3 \rfloor \tag{2}$$
$$s = \lfloor (T \bmod N^3)/N^2 \rfloor \tag{3}$$
$$c = \lfloor (T \bmod N^2)/N \rfloor \tag{4}$$
$$i = T \bmod N \tag{5}$$

Considering a pair of source and destination server with indices $S_{src}$ and $S_{dst}$, the mathematical model of the routing algorithm is shown in Algorithm 1.

**Algorithm 1** Routing mechanism of HFOS$_4$ network

Initialization: $T_{src} = \lceil S_{src}/K \rceil$ and $T_{dst} = \lceil S_{dst}/K \rceil$. Then we can obtain the hyper cluster index $h_s(h_d)$, super cluster index $s_s(s_d)$, cluster index $c_s(c_d)$ and ToR index inside cluster $i_s(i_d)$ of $T_{src}(T_{dst})$ using (1) to (5)

1: **if** $T_{src} = T_{dst}$ **then**
2: Intra-TOR connection
3: **else**
4: **if** $h_s == h_d$ **then** ▷ $S_{src}$ & $S_{dst}$ at the same hyper clusters
5: **if** $s_s == s_d$ **then** ▷ $S_{src}$ & $S_{dst}$ at the same super clusters
6: **if** $c_s == c_d$ **then**
7: $T_{src}$ sent to *level 1 FOS* send to $T_{dst}$ ▷ $S_{src}$ & $S_{dst}$ at the same clusters
8: **else** ▷ $S_{src}$ & $S_{dst}$ at different clusters
9: $T_{src}$ send to *level 2 FOS (level 1 FOS)* send to *Mid TOR* send to *level 1 FOS(level 2 FOS)* send to $T_{dst}$
10: **end if**
11: **else** ▷ $S_{src}$&$S_{dst}$ at different super clusters
12: $T_{src}$ send to *level 3 FOS* send to *Mid TOR*
13: $T_{src}$ = *Mid TOR*
14: go to 5
15: **end if**
16: **else** ▷ $S_{src}$&$S_{dst}$ at different hyper clusters
17: $T_{src}$ send to *level 4 FOS* send to *Mid TOR*
18: $T_{src}$ = *Mid TOR*
19: go to 4
20: **end if**
21: **end if**

In Algorithm 1, when the source and destination TORs are located in the same cluster, they are directly connected through level-1 FOS. For inter-cluster transmissions, path diversity of HFOS$_4$ allows packets destined to outside of cluster to have multiple paths to destinations. As an example, when the source and destination TORs are located at different hyper clusters and their relative locations at each hyper cluster are different, there will be 4! = 24 possible paths between source and destination TORs. Assuming the following indices for source and destination TORs:

$$T_{src} = h_s \times N^3 + s_s \times N^2 + c_s \times N + i_s \tag{6}$$
$$T_{dst} = h_d \times N^3 + s_d \times N^2 + c_d \times N + i_d \tag{7}$$

two of the possible paths are:

Path1: $T_{src}$ → Level 4 FOS→Mid TOR→Level 3 FOS→Mid TOR2→Level 2 FOS→Mid TOR→Level 1 FOS→$T_{dst}$

Path2: $T_{src}$ →Level 3 FOS→Mid TOR1→Level 2 FOS →Mid TOR2→Level 1 FOS→Mid TOR3→Level 4 FOS→ $T_{dst}$

## 4 Simulation environment

OMNeT++ simulation framework is utilized to develop the proposed HFOS$_4$ DCN architecture. We conduct various simulations to completely investigate the network performance of HFOS$_4$ in terms of average end-to-end delay and packet loss. To validate the advantages of HFOS$_4$ DCN architecture, we compare the network performance of HFOS$_4$ with that of FOS-based DCNs OPSquare and FOScube.

### 4.1 Traffic generation

The traffic flowing through DCNs cannot be generalized since the traffic characteristics highly depend on the applications hosted inside the DCN. However, some studies [31–33] report key parameters to be used to synthesize the traffic pattern of the DCNs. In this manner, to evaluate the performance of HFOS$_4$, the realistic traffic pattern with ON/OFF inter-arrival times is applied [32].

During the ON periods, servers generate and send packets towards a destination server, while no packet is generated during the OFF periods. The duration of ON/OFF periods are modeled with a Pareto distribution since it is characterized by heavy-tailed random distribution. In our simulation set, the cumulative distribution function (CDF) of ON periods in accordance with length distribution and the CDF distribution of OFF periods. The ON period lengths are independent of load value, while OFF time lengths are proportional to the traffic load value. In our simulation data sets, all servers can send data packets during ON periods and the destination server is chosen under the uniform distribution.

Considering the locality of DC traffic, the majority of the traffic originating from servers remains within the rack (intra-TOR traffic) to reduce the communication costs [34]. The rest of the traffic is exchanged between servers of different racks. Specifically, we set the traffic ratios to 50% intra-TOR and 50% inter-TOR. The inter-TOR traffic distribution is categorized into intra-cluster and inter-cluster traffic as follows: intra-cluster traffic is exchanged between the randomly chosen destinations located in the same cluster (HFOS$_1$) while the inter-cluster traffic is destined to randomly chosen destinations located in the rest on the DCN (HFOS$_i$, $2 \leq i \leq 4$). Table2 shows three types of traffic patterns employed to verify the scalability performance of the HFOS$_4$.

**Table 2** The traffic pattern ratio

| Traffic pattern | P1 (%) | P2 (%) | P3 (%) |
|---|---|---|---|
| Intra-TOR | 50 | 50 | 50 |
| Intra-cluster | 35 | 37.5 | 40 |
| Inter-cluster | 15 | 12.5 | 10 |

**Table 3** The simulation parameters

| Parameter | Value | |
|---|---|---|
| RTT | 560 ns | |
| Propagation delay(2*50m) | 500 ns | |
| Head processor delay | 80 ns | |
| Label processing delay | 20 ns | |
| Buffering time of the cells | 51.2 ns | |
| Link length | 50 m | |
| Number of TRX | Level 1 TRX | 4 |
| | Level 2 TRX | 1 |
| | Level 3 TRX | 1 |
| | Level 4 TRX | 1 |

In our traffic model, packet size is a bimodal distribution with packets between 64 bytes (small control packet) and 1518 bytes (data payloads). This bimodal distribution stems from the dominant TCP-based applications inside DCNs [31, 32].

### 4.2 Simulation parameters

Each rack groups 40 servers connecting to a TOR switch via 10Gbps links. TORs are connected to 4 levels of FOSs utilizing 50Gbps WDM links. Packets leaving the TORs are stored in WDM transceiver buffers of size 50 KB. A generated packet of size *Packet Length* occupies $N_{cell}$ 64-byte buffer cells where $N_{cell} = Packet\ Length/64$. Hence, 25 cells with the same destination are grouped to create an optical packet of size 1600 bytes. The number of transceivers interfacing optical switches does not change during the simulation. Table 3 lists the simulation parameters.

## 5 Performance assessment of $HFOS_4$ DCN

The simulation of end-to-end latency and packet loss of $HFOS_4$ under different network configurations and traffic loads in the range of [0.1, 1] is conducted in this section to evaluate the performance of $HFOS_4$. It is worth noting that packet loss occurs in the scenario that the TOR buffers are full so the packet retransmission is not possible. Packet loss ratio is computed as the ratio of the number of discarded packets to the total number of sent packets.

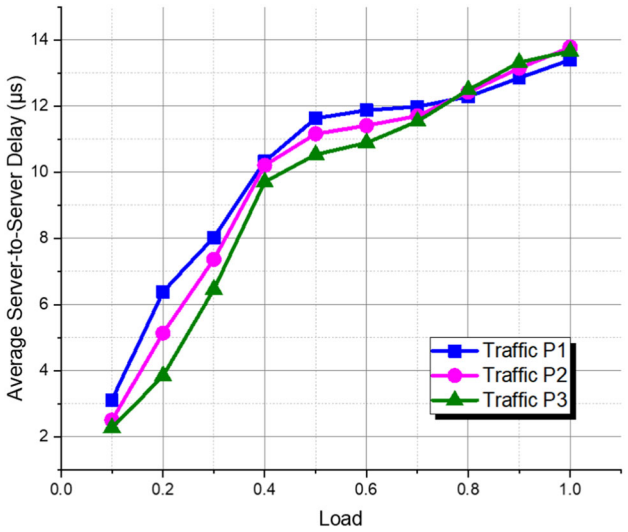


**Fig. 7** The average server-to-server delay of HFOS4 under different traffic patterns

### 5.1 Performance assessment under varying traffic patterns

In this section, we carry out simulations to investigate the average end-to-end delay and packet loss of $HFOS_4$ under varying traffic patterns to evaluate the network performance assuming dynamic traffic patterns. FOSs of radix-8 are employed to build an $HFOS_4$ network with two hyper clusters, each consisting of 512 TORs. Hence the network interconnects 40960 servers. Since the intra-TOR traffic is handled within the electrical domain, in this section, only the inter-TOR traffic is considered to evaluate the performance of $HFOS_4$ network under various traffic patterns.

Figure 7 shows the server-to-server average delay of the $HFOS_4$ under traffic patterns P1, P2, and P3. The server-to-server average delay increases when the network load increases. The delay consists of transmission delay, propagation delay, and buffering delay. When the load is 0.1, the network delay for three traffic patterns is almost the same because there are very few contentions and retransmissions in the network, so only the propagation, transmission delay, and buffering delay contribute to the average delay. For heavier traffic loads, the buffering delay occurs at each hop and retransmission delay caused by packet loss increases the average server-to-server delay. For the traffic load less than 0.8, it is observed that the server-to-server delay of traffic pattern P1 (35% intra-cluster traffic) is larger than that of the traffic pattern P2 (37.5% intra-cluster traffic), as well as traffic pattern P3 (40% intra-cluster traffic). The reason is that assuming traffic load of P1, the amount of inter-cluster traffic is higher than that of traffic pattern P2, and hence, more amount of the traffic volume traverses more than one hop which increases the propagation delay, as well as occupied

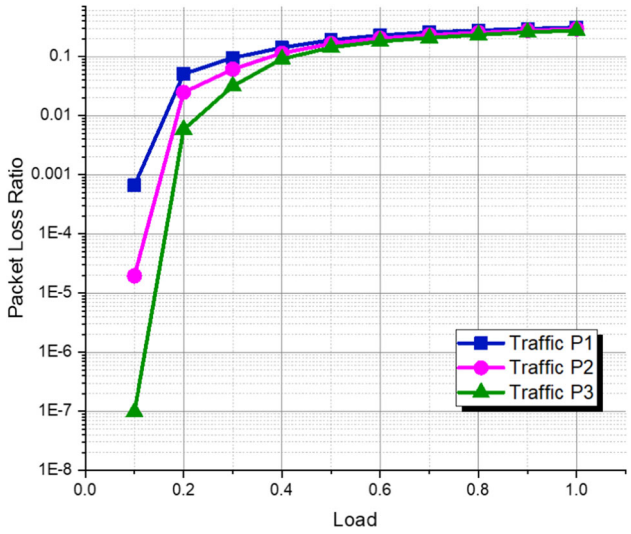


**Fig. 8** The packet loss of HFOS$_4$ under different traffic patterns

intermediate links and buffers. Assuming heavy loads of 0.9 and 1, the server-to-server delay for the three traffic patterns is almost the same, since at very high loads the aggregation level FOS (intra-cluster FOS) has to handle the heavy load of intra-cluster traffic, along with inter-cluster traffic, and hence the buffers are fully occupied.

Figure 8 shows the network packet loss ratio as a function of load for three traffic patterns P1, P2, and P3. These three traffic patterns have different traffic localities. As shown in Fig. 8, the packet loss ratio for the case of P1 is larger than that of case P2. The reason is the percentage of inter-cluster traffic ratio of P1 is more than that of P2, hence a larger percentage of inter-cluster packets has to pass multiple hops towards destination resulting in more occupied middle links and buffers.

Similarly, the packet loss ratio of P2 is larger than the packet loss ratio of P3. However, the packet loss ratio of HFOS$_4$ increases slightly with an increase in traffic loads. The level of quality of service determines the acceptable packet loss of the network. The packet loss ratios at load 0.1 for the traffic patterns P1, P2, and P3 are equal to 9.7E-8, 1.95E-5, and 6.5E-4 respectively which is regarded as"excellent"according to [35]. The accepted packet loss ratio is around 2.5% according to [35] which is guaranteed at load 0.3 traffic pattern P3. The packet loss ratio is under 0.32 even for heavy traffic pattern P1 of load 1 when all links are fully occupied. The simulation results show that HFOS$_4$ can handle the traffic with heavy loads. This characteristic is because of having 4 parallel levels resulting in the high path diversity of the architecture. Due to the high path diversity, packets have large number of paths to route so the HFOS$_4$ can maintain the network performance even under heavy traffic patterns and loads.

## 5.2 Scalability assessment of HFOS$_4$

In this section we investigate the scalability of the proposed architecture. HFOS$_L$ can be scaled-out easily by expanding the number of nodes in HFOS$_{L-1}$. For example, in HFOS$_4$ of size 1024 TORs adding a hyper cluster of size 512 TORs results in the network size of 61440 servers. The addition of a hyper cluster does not require modification of the current network switches. To explore the scalability of HFOS$_4$, we focus on the performance of the network under the heavy traffic pattern P2 for three network sizes of 40960, 66440, and 81920 servers to investigate how the performance changes when the network size scales.

Figure 9.a shows the average server-to-server delay of HFOS$_4$ as a function of the load. It can be seen that when the load is below 0.4, the average server-to-server delay for 81920 servers is around 1$\mu$s more than that of 61440 servers, and 2$\mu$s more than that of 81920 servers. For heavier traffic loads, the average delays slightly increase when the network size increases. Hence, we can conclude that HFOS$_4$ maintains its performance under high traffic loads as the network scales out.

Figure 9.b shows the average network delay for three different network sizes of HFOS$_4$. The network delay is calculated based on the delay between the $T_{src}$ and $T_{dst}$. Similar to the case of server-to-server delay for loads smaller than 0.4, the delay of the largest size network is larger than the smaller size. For heavier loads, network delay maintains while scaling the network size.

Figure 9.c shows the packet loss of HFOS$_4$ for the three aforementioned network sizes. The packet loss around 1% is reported at load 0.4 for the DCN size of 10240 servers and never exceeds 0.3 even for the heaviest load of 1 regardless of DCN size.

## 5.3 Comparing HFOS$_4$ against similar DCN architectures

This section compares the performance of HFOS$_4$ with FOScube and OPSquare in terms of average server-to-server and network delays, as well as packet loss for three DCN sizes of 10240, 40960, and 81920 servers. We simulate the three aforementioned network architectures under the same traffic pattern P2, where each TOR sends 37.5% of its traffic to $M$ neighboring TORs, and 12.5% of traffic is destined outside the clusters. In order to have a fair comparison, in the case of 256 TORs, $M$ is equal to 4, while in the cases of 1024 and 2048 TORs, $M$ is equal to 8. The network configurations are based on Table 4.

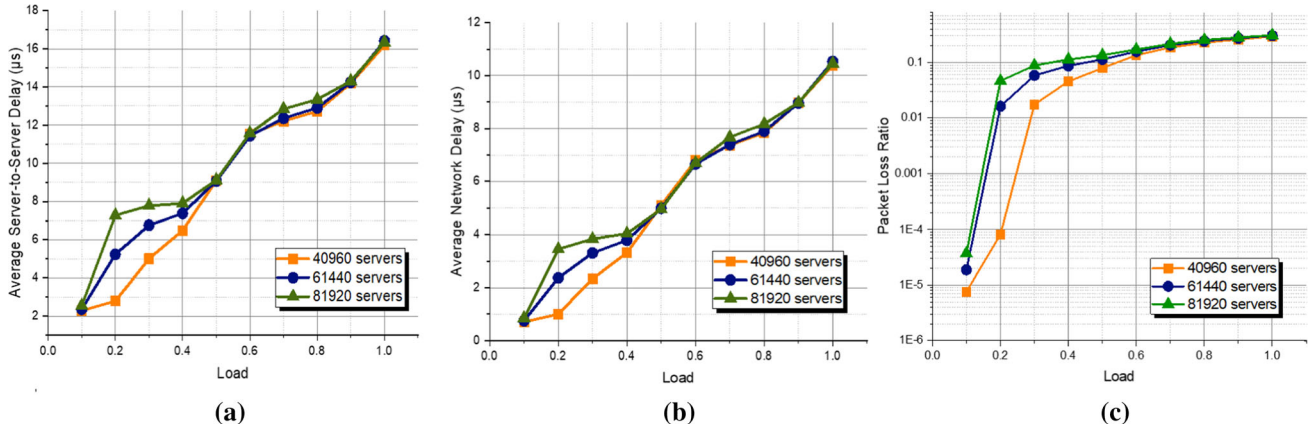


**Fig. 9** **a** The average server-to-server delay textbfb The average network delay **c** The packet loss

**Table 4** The network configuration based on the required FOS radix

| **Network size** / **Topology** | 10240 servers (R/N) | 40960 servers (R/N) | 81920 servers (R/N) |
|---|---|---|---|
| OPSquare | 16/32 | 32/64 | 64/128 |
| FOScube | 8/96 | 16/192 | 16/384 |
| $HFOS_4$ | 4/256 | 8/512 | 8/1024 |

### 5.3.1 Comparison of small size DCNs

Figure 10 shows the average server-to-server delay for three DCN architectures for 10240 servers. As it can be seen, in the case of load 0.1, delay behaves similarly for various DCNs due to the lack of packet contention. At the load 0.2, the delay of $HFOS_4$ is almost equal to that of OPSquare, and $2\mu$s more than the delay of FOScube. The reason is that assuming low traffic loads, although there is still low packet contention, inter-cluster packets of $HFOS_4$ has to traverse more hops than that of FOScube towards destination, so propagation delay of $HFOS_4$ is larger than that of the FOScube. At loads higher than 0.2, the delay of $HFOS_4$ grows slowly compared to OPSquare and FOScube. Because $HFOS_4$ has larger connectivity than that of FOScube and OPSquare resulting in lower packet contention.

Figure 11 compares the packet loss of the three aforementioned DCNs. At the load of 0.2, the packet loss of $HFOS_4$ is larger than that of FOScube due to the multiple-hop path of inter-cluster packets. At heavier loads, the packet loss ratio of $HFOS_4$ is clearly less than that of OPSquare and FOScube. Path diversity of $HFOS_4$ results in lower packet contention resulting in lower packet loss ratio.

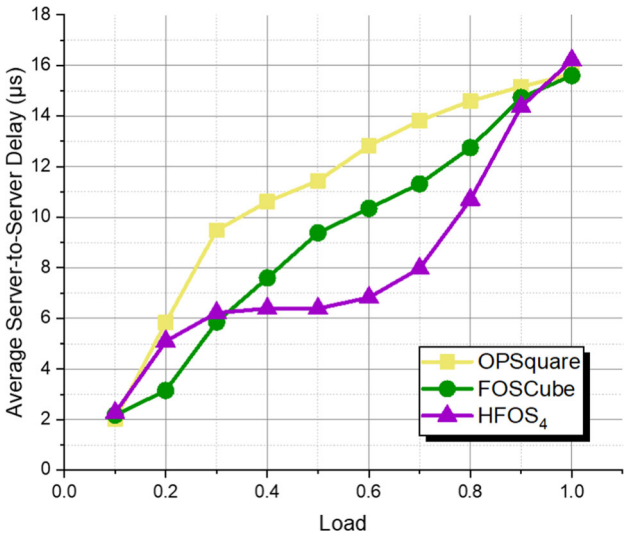


**Fig. 10** Comparison of average server-to-server delays for DCN size of 10240 servers

### 5.3.2 Comparison of medium and large size DCNs

Performance comparison of $HFOS_4$ against OPSquare and FOScube DCNs interconnecting 40960 and 81920 servers is investigated in this section. Figure 12 shows the average server to server delay for three DCNs. Similar to the small network sizes, in the case of small traffic load of 0.1, various DCNs result in the same average delay due to the lack of packet contention.

For the case of 40960 servers, the delay of $HFOS_4$ outperforms the other architecture for loads heavier than 0.2. $HFOS_4$ improves server to server delay by 35 and 33% compared with FOScube and OPSquare at load of 0.5 respectively. At heavy load of 1, delay improvement of $HFOS_4$ is 16 and 10% compared with FOScube and OPSquare.

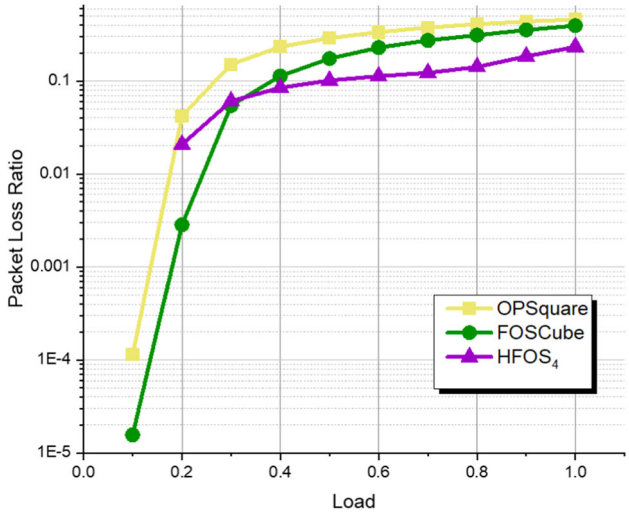


**Fig. 11** The packet loss for network size of 10240 servers

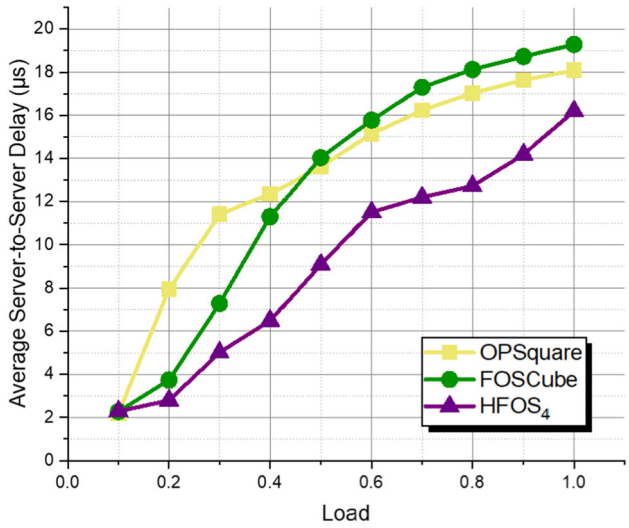


**Fig. 12** The delay comparison for DCN size of 40960 servers

The average server to server delay for the case of 81920 servers is depicted in Fig. 13. The trend is almost similar to that of case 40960 servers but at load below 0.2, the delay of HFOS$_4$ is more than the delay of FOScube. The reason is that in the case of very large network size, inter-cluster traffic of HFOS$_4$ needs to traverse more hops than that of FOScube, and hence, the propagation delay increases. However, for loads heavier than 0.2, HFOS$_4$ clearly outperforms FOScube and OPSquare. For the network size of 81920 servers at load of 0.5, HFOS$_4$ improves average server to server delay by 40 and 29% compared with FOScube and OPSquare, respectively. To sum up, the high path diversity of HFOS$_4$ allows inter-cluster traffic to traverse to the destination through multiple paths which clearly leads to lower delay compared to OPSquare and FOScube.

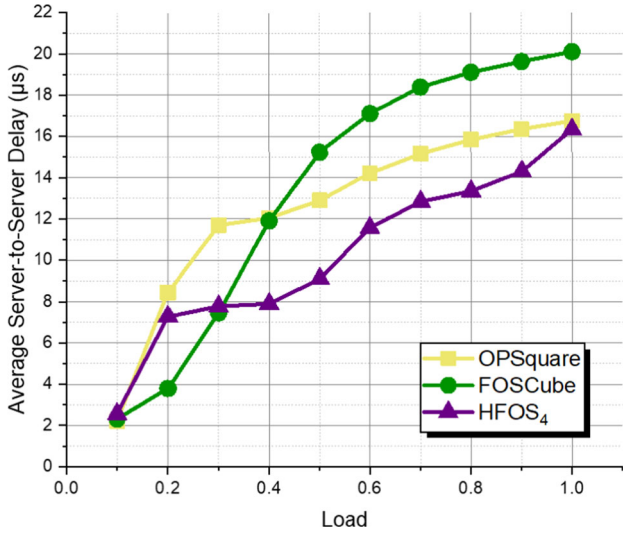


**Fig. 13** The delay comparison for DCN size of 81920 servers

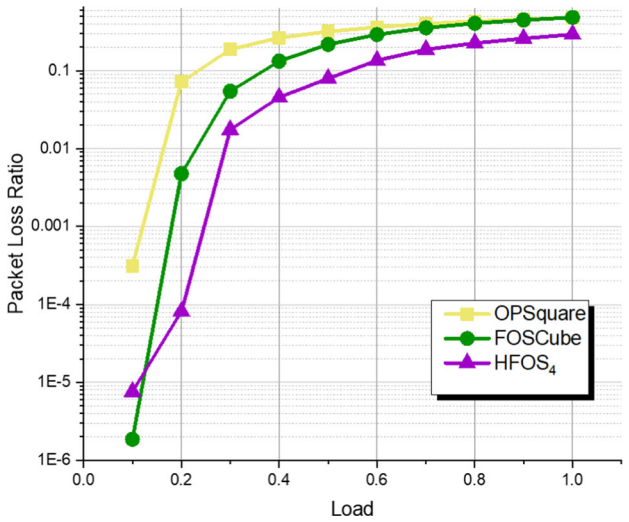


**Fig. 14** The packet loss for network size of 40960 servers

Figures 14 and 15 report the packet loss of HFOS$_4$ and FOScube and OPSquare as a function of load. For the network size of 40960 servers, shown in 14, at loads 0.1, packet loss is slightly larger than that of FOScube. This transient increase in traffic load of 0.1 is because of multiple hops of HFOS$_4$ results in more packet contention at intermediate nodes. The impact of packet contention at higher loads is mitigated by more connections of TORs in HFOS$_4$. Similar discussion for the case of 40960 applies to the 81920 network size which is depicted in 15.

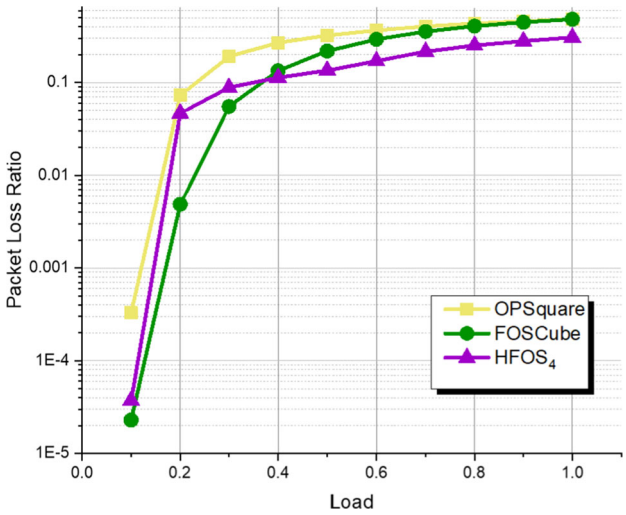


**Fig. 15** The packet loss for network size of 81920 servers

**Table 5** The cost and power consumption of network elements [36]

| Element | | Cost($) | Power(W) |
|---|---|---|---|
| SM_1 TRX(10Gbps) | | 70 | 1 |
| WDM TRX ($4 \times 50Gbps$) | | 750 | 4 |
| FOS | $4 \times 4$ | 1250 | 217 |
| | $8 \times 8$ | 6220 | 441 |
| | $16 \times 16$ | 22860 | 985 |
| | $32 \times 32$ | 87980 | 2457 |
| | $48 \times 48$ | 177100 | 4443 |

## 6 Cost and power consumption analysis of $HFOS_L$ network

In this section, we investigate and compare the cost and power consumption of $HFOS_L$ for $2 \leq L < 6$ with Fat tree architecture. For estimating cost and power consumption, we consider the cost and power consumption of DCN elements, while the number of servers for all architectures are assumed to be equal. $HFOS_L$ is built upon $L$ parallel $HFOS_{L-1}$, each containing $N^{(L-1)}$ FOSs. Hence, the total number of FOSs ($N_F$) in $HFOS_L$ is $N_F = L \times N^{(L-1)}$ . In this way, $HFOS_L$ accommodates $S_L$ servers using (8):

$$S_L = K \times N_T = K \times N^{(L-1)} \times N = K \times N^L \tag{8}$$

where $N_T$ is the number of TORs in the network. A list of cost and power consumption of optical network elements is reported in Table 5.

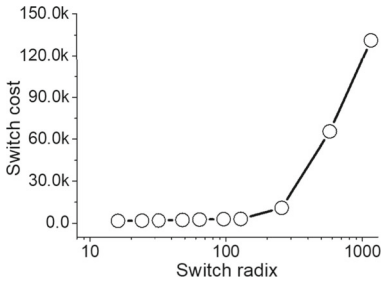


**Fig. 16** The cost of electrical switch regarding switch radix [26]

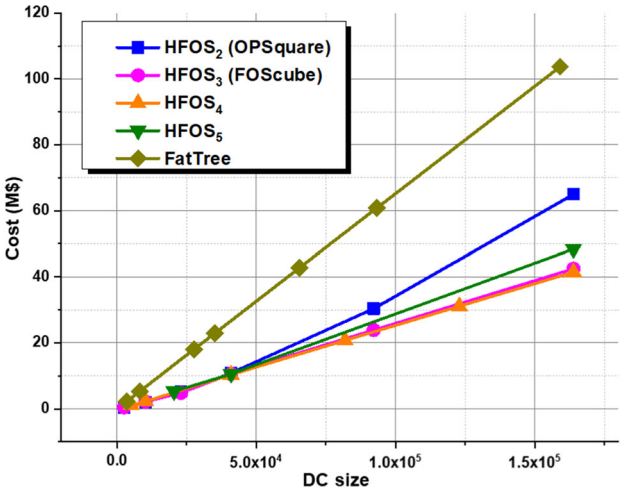


**Fig. 17** The Cost comparison of $HFOS_L$ and Fat tree

### 6.1 Cost calculation

The cost of $HFOS_L$ based on the required number of TORs and optical elements can be estimated based as follows:

$$\begin{aligned} C_{HFOS_L} &= N_T \times C_T + N_F \times C_F \\ &+ (p + L - 1) \times N_T \times C_{50GTRX} \\ &+ L \times N_F \times C_S \end{aligned} \tag{9}$$

where, $C_T$, $C_F$, $C_S$, and $C_{50GTRX}$ stand for the cost of TOR, FOS, Single Mode Fiber (SMF), and 50Gbps transceivers, respectively. The costs of optical fibers are estimated as follows: SMF cost is 0.3$ per meter and Multi Mode Fiber (MMF) cost is 0.9$ per meter. The costs of electrical switches are shown in Fig. 16. It is seen that for electrical switch radix smaller than 128, the cost increases slowly, while the cost of switch sizes larger than 128 increases rapidly due to the required multiple main boards.

Figure 17 depicts cost comparison between $HFOS_L$ DCNs for $2 < L < 6$ and Fat-tree. As shown in this figure, for the network sizes smaller than 50K servers, the costs of $HFOS_L$ with various levels are very close to each other. For DCN sizes larger than 100K servers, the cost of $HFOS_4$ is slightly less than $HFOS_2$ and $HFOS_3$ since $HFOS_L$ can employ low-radix

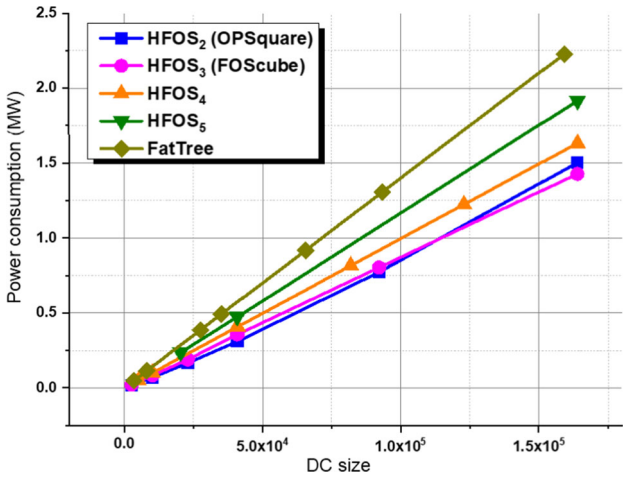


**Fig. 18** The power consumption comparison of HFOS$_L$ and Fat tree

FOSs to build large DCN sizes. However, by increasing the number of levels of HFOS$_L$, it is seen that the cost of HFOS$_5$ is larger than that of HFOS$_4$ since a linear increase of levels results in superlinear increase of the number of required FOSs. Moreover, the number of adopted TRXs increases when the number of levels increases. Therefore, $L = 4$ is a saturation point for cost saving. Moreover, HFOS$_4$ outperforms Fat tree by 60% in cost saving for the network size of around 160,000 servers. The cost saving of HFOS$_4$ is due to removing the costly transceivers of Fat tree architecture in optical networks.

### 6.2 Power consumption

The power consumption of HFOS$_L$, based on the number of TORs and the required optical devices, is calculated as shown in (10):

$$\begin{aligned} P_{HFOS_L} &= N_T \times P_T \\ &\quad + N_F \times P_F \\ &\quad + (p + L - 1) \times N_T \times C_{50GTRX} \end{aligned} \tag{10}$$

where, $P_T$, $P_F$, and $P_{50GTRX}$ represent the power consumption of TOR, FOS, and the 50Gbps transceivers, respectively. Figure 18 compares power consumption of HFOS$_L$ DCNs for $2 < L < 6$ and Fat tree. The analysis of the result is similar to the cost comparison, shown in Fig. 16 for $L > 2$. As depicted in Fig. 18, the power consumption of HFOS$_{i+1}$ is larger than that of HFOS$_i$. The reason is that adding another level to HFOS$_i$, adds $N_i$ FOSs to build HFOS$_{i+1}$, which leads to the fast growth in the required number of FOSs.

Unlike the conclusion achieved for cost comparison, the power consumption of HFOS$_{i+1}$ is slightly larger than that of HFOS$_i$. The reason is that when the FOS radix doubles, the power consumption scales around 2 times while the cost scales around 4 times. Finally, the power consumption shows that for the DCN accommodating 160,000 servers, the HFOS$_4$ saves cost by 26.7% compared to Fat tree.

## 7 Conclusion

To build mega size DCN, a scalable $L$ parallel level DCN architecture based on low-radix FOS switches, HFOS$_L$, is presented and analyzed. Benefiting from multiple level subnetworks, HFOS$_L$ supports more than a hundred of thousands of servers employing small radix FOS. Numerical assessments of cost and power consumption show that HFOS$_L$ with 4 parallel levels has the most efficient structure in terms of cost and power consumption for $L > 2$.

The comprehensive simulations of the networks using OMNet++ have been performed to investigate the network performance under the realistic data center traffic. Employing low-radix FOSs of 16, HFOS$_4$ can scale up to more than 2.5 million servers while maintaining the network performance. Specifically, HFOS$_4$ shows very good performance at high network loads which makes it suitable for DCNs with intense traffic volume. The assessments of end-to-end delay show HFOS$_4$ outperforms OPSquare and FOScube by 33 and 35% under the network size of 40960 servers at load of 0.5. For the case of 81920 servers, HFOS$_4$ improves end-to-end delay by up to 29 and 40% at the load of 0.5.

The large path diversity of HFOS$_4$ increases network fault tolerance and allows HFOS$_4$ to support various traffic patterns with various localities and helps the implementation of load balancing algorithm in the network. Therefore, HFOS$_4$ accommodating extremely large DCN sizes with high performance even for high loads make this network a scalable alternative for modern data center environments.

**Funding** Not applicable

**Availability of data and material** Not applicable

## Declarations

**Conflict of interest** The authors declare that there is no conflict of interest regarding the publication of this paper.

**Code availability** Not applicable

## References

1. Global data center IP traffic 2013-2021 Statista. [Online] Available: https://www.statista.com/statistics/227246/global-datacenter-ip-traffic-development-forecast.
2. Cheng, Q., Glick, M., & Bergman, K. (2020). Optical interconnection networks for high-performance systems. *In Optical Fiber Telecommunications VII* (pp. 785–825). Academic Press.

3. Ghiasi, A. (2015). Large data centers interconnect bottlenecks. *Optics Express, 23*(3), 2085–2090. https://doi.org/10.1364/OE.23.002085
4. Ballani, H., Costa, P., Behrendt, R., Cletheroe, D., Haller, I., Jozwik, K., ... & Williams, H. (2020). Sirius: A flat datacenter network with nanosecond optical switching. In *Proceedings of the Annual conference of the ACM Special Interest Group on Data Communication on the applications, technologies, architectures, and protocols for computer communication* (pp. 782–797).
5. Minkenberg, C., Farrington, N., Zilkie, A., Nelson, D., Lai, C. P., Brunina, D., Byrd, J., Chowdhuri, B., Kucharewski, N., Muth, K., Nagra, A., Rodriguez, G., Rubi, D., Schrans, T., Srinivasan, P., Wang, Y., Yeh, C., & Rickman, A. (2018). Reimagining datacenter topologies with integrated silicon photonics. *Journal of Optical Communications and Networking, 10*(7), B126–B139. https://doi.org/10.1364/JOCN.10.00B126
6. Testa, F., & Pavesi, L. (Eds.). (2017). *Optical switching in next generation data centers*. Berlin: Springer.
7. Xu, M., Diakonikolas, J., Modiano, E., & Subramaniam, S. (2019). A hierarchical WDM-based scalable data center network architecture. In *ICC 2019-2019 IEEE International Conference on Communications (ICC)* (pp. 1–7).
8. Prifti, K., Santos, R., Shin, J., Kim, H., Tessema, N., Stabile, P., Kleijn, S., Augustin, L., Jung, H., Park, S., Baek, Y., Hyun, S., & Calabretta, N. (2020). All-optical cross-connect switch for data center network application. In *2020 Optical Fiber Communications Conference and Exhibition (OFC)* (pp. 1–3).
9. Foerster, K., & Schmid, S. (2019). Survey of reconfigurable data center networks: Enablers, algorithms, complexity. *SIGACT News, 50*(2), 62–79. https://doi.org/10.1145/3351452.3351464
10. Chen, K., Singla, A., Singh, A., Ramachandran, K., Xu, L., Zhang, Y., Wen, X., & Chen, Y. (2014). OSA: An optical switching architecture for data center networks with unprecedented flexibility. *IEEE/ACM Transaction On Networking, 22*(2), 498–511. https://doi.org/10.1109/TNET.2013.2253120
11. Nance Hall, M., Foerster, K. T., Schmid, S., & Durairajan, R. (2021). A survey of reconfigurable optical networks. *Optical Switching and Networking*. https://doi.org/10.1016/j.osn.2021.100621
12. Yin, Y., Proietti, R., Ye, X., Nitta, C., Akella, V., & Yoo, S. (2013). LIONS: An AWGR-based low-latency optical switch for high-performance computing and data centers. *IEEE Journal of Selected Topic Quantum Electron., 19*(2), 3600409–3600409. https://doi.org/10.1109/JSTQE.2012.2209174
13. Kachris, C., & Tomkos, I. (2013). Power consumption evaluation of all-optical data center networks. *Cluster Computing, 16*, 611–623. https://doi.org/10.1007/s10586-012-0227-6
14. Chen, K., Wen, X., Ma, X., Chen, Y., Xia, Y., Hu, C., Dong, Q., & Liu, Y. (2017). Toward a scalable, fault-tolerant, high-performance optical data center architecture. *IEEE/ACM Transaction On Networking, 25*(4), 2281–2294. https://doi.org/10.1109/TNET.2017.2688376
15. Keykhosravi, K., Rastegarfar, H., & Agrell, E. (2018). Multicast scheduling of wavelength-tunable, multiqueue optical data center switches. *Journal of Optical Communications and Networking, 10*(4), 353–364. https://doi.org/10.1364/JOCN.10.000353
16. Mellette, W. M., Das, R., Guo Y., McGuinness, R., Snoeren, A. C., & Porter, G. (2020). Expanding across time to deliver bandwidth efficiency and low latency. 17th USENIX Symposium on Networked Systems Design and Implementation (NSDI 20), (pp. 1–18).
17. Johannes, Z., Wolfgang, K., & Andreas, B. (2021). What you need to know about optical circuit reconfigurations in datacenter networks. *2021 33th International Teletraffic Congress (ITC-33)*. 1–9.
18. Peng, L., Xiaoshan, Y., Huaxi, G., & Yunfeng, L. (2021). FlexNet: A optical switching architecture for optical data center networks. In *2021 19th International Conference on Optical Communications and Networks (ICOCN)*. 1-3. https://doi.org/10.1109/ICOCN53177.2021.9563652
19. Yan, F., Xue, X., & Calabretta, N. (2018). HiFOST: A scalable and low-latency hybrid data center network architecture based on flow-controlled fast optical switches. *IEEE/OSA Journal of Optical Communications and Networking, 10*(7), 1–14. https://doi.org/10.1364/JOCN.10.0000B1
20. Xi, K., Kao, Y. H., & Chao, H. J. (2013). A petabit bufferless optical switch for data center networks. *Optical interconnects for future data center networks* (pp. 135–154). New York, NY: Springer.
21. Bao, J., Dong, D., Zhao, B., & Huang, S. (2019). HyFabric : Minimizing FCT in optical and electrical hybrid data center networks. In *Proceedings of the ACM SIGCOMM 2019 Conference Posters and Demos (SIGCOMM Posters and Demos '19)*. 57–59. https://doi.org/10.1145/3342280.3342306
22. Bakopoulos, P., Christodoulopoulos, K., Landi, G., Aziz, M., Zahavi, E., Gallico, D., & Avramopoulos, H. (2018). NEPHELE: An end-to-end scalable and dynamically reconfigurable optical architecture for application-aware SDN cloud data centers. *IEEE Communications Magazine, 56*(2), 178–188. https://doi.org/10.1109/MCOM.2018.1600804
23. Sato, K. I., Hasegawa, H., Niwa, T., & Watanabe, T. (2013). A large-scale wavelength routing optical switch for data center networks. *IEEE Communications Magazine, 51*(9), 46–52. https://doi.org/10.1109/MCOM.2013.6588649
24. Wang, K., Zhao, L., Gu, H., Yu, X., Wu, G., & Cai, J. (2015). ADON: A scalable AWG-based topology for datacenter optical network. *Optical and Quantum Electronics, 47*(8), 2541–2554. https://doi.org/10.1007/s11082-015-0136-z
25. Chen, K., Wen, X., Ma, X., Chen, Y., Xia, Y., Hu, C., Dong, Q., & Liu, Y. (2017). Toward a scalable, fault-tolerant, high-performance optical data center architecture. *IEEE/ACM Transactions on Networking, 25*(4), 2281–2294. https://doi.org/10.1109/TNET.2017.2688376
26. Yan, F., Miao, W., Raz, O., & Calabretta, N. (2017). Opsquare: A flat DCN architecture based on flow-controlled optical packet switches. *IEEE/OSA Journal of Optical Communications and Networking, 9*(4), 291–303. https://doi.org/10.1364/JOCN.9.000291
27. Khani, E., Yan, F., Guo, X., & Calabretta, N. (2019). Theoretical analysis on multiple layer fast optical switch based data center network architecture. In *2019 24th OptoElectronics and Communications Conference (OECC) and 2019 International Conference on Photonics in Switching and Computing (PSC)* (pp. 1–3).
28. Yan, F., Xue, X., Pan, B., Guo, X., & Calabretta, N. (2018). FOScube: a scalable data center network architecture based on multiple parallel networks and fast optical switches. In *2018 European Conference on Optical Communication (ECOC)* (pp. 1–3).
29. Levi, C., & Segal, M. (2021). Avoiding bottlenecks in networks by short paths. *Telecommunication Systems: Modelling, Analysis, Design and Management, 76*(4), 491–503. https://doi.org/10.1007/s11235-020-00720-7
30. Miao, W., Luo, J., Di Lucente, S., Dorren, H., & Calabretta, N. (2014). Novel flat datacenter network architecture based on scalable and flow-controlled optical switch system. *Optics Express, 22*(3), 2465–2472. https://doi.org/10.1364/OE.22.002465
31. Benson, T., Akella, A., & Maltz, D. A. (2010, November). Network traffic characteristics of data centers in the wild. In *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement* (pp. 267–280).
32. Benson, T., Anand, A., Akella, A., & Zhang, M. (2010). Understanding data center traffic characteristics. *ACM SIGCOMM Computer Communication Review, 40*(1), 92–99.
33. Noormohammadpour, M., & Raghavendra, C. S. (2017). Datacenter traffic control: Understanding techniques and tradeoffs. *IEEE*

*Communications Surveys & Tutorials, 20*(2), 1492–1525. https://doi.org/10.1109/COMST.2017.2782753
34. Kandula, S., Sengupta, S., Greenberg, A., Patel, P., & Chaiken, R. (2009). The nature of data center traffic: measurements & analysis. *IMC '09: Proceedings of the 9th ACM SIGCOMM conference on Internet measurement*, 9, (pp. 202–208).
35. Tutorial on Internet Monitoring and PingER at SLAC. [Online]. Available:www.slac.stanford.edu/comp/net/wanmon/tutorial.html (accessed Aug. 09, 2020).
36. Categories-Elpeus Technology. [Online]. Available: http://www.elpeus.com/categories.



**Elham Khani** started her Ph.D. at Sharif University of Technology, Tehran, Iran in 2015. She joined ECO group of department of Electrical Engineering, Eindhoven University of Technology, Eindhoven, the Netherlands as a guest researcher in 2019. She is currently a Ph.D. candidate at Sharif University of Technology. Her research interests include optical interconnection networks, data center designs, and reconfigurable optical networks.

**Shaahin Hessabi** is an associate professor of Computer Engineering at Sharif University of Technology, Tehran, Iran. He received the BS and MS degrees in electrical engineering from Sharif University of Technology in 1986 and 1990, respectively, and the PhD degree in electrical and computer engineering from the University of Waterloo, Ontario, Canada in 1995. He joined Sharif University of Technology in 1996, and is the director of the Efficient Processing and Communication Architectures (EPCA) Lab. He has published more than 130 refereed papers in the related areas. His research interests include optical interconnection networks, cyber-physical systems, reconfigurable and heterogeneous architectures, and system-on-chip.

**Somayyeh Koohi** is an Assistant Professor of Computer Engineering at Sharif University of Technology, Tehran, Iran. She completed her B.Sc. double degree from Sharif University of Technology in both Electrical Engineering and Computer Engineering in 2005. She then received her M.Sc. and Ph.D. degrees from Sharif University of Technology in Computer Engineering in 2007 and 2012, respectively. Her research group focuses on developing novel optical tools and techniques for biomedical research and applications, optical processing and communication, biophotonics, and optical network-on-chip as a novel solution for future systemson-chip.

**Fulong Yan** got the bachelor and master degree from Beihang University, Beijing, China in 2010 and 2013, respectively. He earned his Ph.D. degree from Eindhoven University of Technology, Eindhoven, the Netherlands, in 2019. Afterwards, he started as a Post-Doc researcher at Eindhoven University of Technology. He is IEEE member since 2019. His research interests include the high performance data center network architecture, low latency scheduling algorithms in packet switching network, traffic modeling and traffic prediction. He serves as reviewer for multiple IEEE/OSA journals, including Computer Communications, Journal of Optical Communications and Networking, Journal of Lightwave Technology, Optics Express, and Optics Letters. He has co-authored over 40 journal and conference papers and holds one patent. His research interests are optical packet switching, packet scheduling algorithm, and high-performance optical networks.

**Nicola Calabretta** received a Ph.D. degree from the Eindhoven University of Technology (TUE) in 2004. From 2004 to 2007, he was a researcher with Scuola Superiore Sant'Anna Pisa and with the Technical University of Denmark. He is currently with TUE. He has co-authored over 200 journal and conference papers and holds three patents. His research interests are optical packet switching, signal processing, and high-performance optical networks.