FOSphere: A Scalable and Modular Low Radix Fast Optical Switch Based Data Center Network

Fulong Yan, Elham Kahan, Xiaotao Guo, Fu Wang, Bitao Pan, Xuwei Xue, Shaojuan Zhang and Nicola Calabretta
Eindhoven University of Technology, Eindhoven, the Netherlands
f.yan@tue.n

Abstract: We propose a novel scalable and modular low-radix fast optical switch based DCN with sphere topology (FOSphere). Numerical analyses on 10880-server indicates that FOSphere achieves 4.1 μs server-to-server latency and 2.6E-3 packet loss at load 0.4. © 2020 The Authors

1. Introduction

The exponentially increasing of the data center (DC) traffic has imposed stringent requirements on the data center network (DCN) [1]. The current electrical switches based DCN may be inefficient in terms of cost and power consumption as scaling the number of servers, and fail to meet the requirements of high bandwidth, low latency, and large connectivity. To address those issues, DCNs adopting optical switching technologies, including semiconductor optical amplifier (SOA) based fast optical switch (FOS), optical circuit switch (OCS), and arrayed waveguide grating (AWGR) with tunable lasers, have been proposed [2]. However, the reconfiguration time of OCS is on the order of milliseconds, and therefore it fails to handle the dynamic traffic in the DC [3]. Besides, the high cost of the tunable lasers makes the AWGR solution expensive as the network scales.

Among the FOS based DCNs, OPSquare features good scalability, high bandwidth, and low latency by employing two parallel intra-cluster and inter-cluster subnetworks [4]. HiFOST further improves the cost and power efficiency of OPSquare while maintains the same performance by removing one level of FOS [5]. However, to build DCN supporting >100,000 servers, both OPSquare and HiFOST need challenging FOS with radix of 128. To address the large connectivity, FOScube employs three subnetworks with three genres of FOS to achieve a scalability of N^3 [6]. With the rapid expanding of the DCN, there is a trends of building modular DC aiming at fast deployment and convenient upgrade. However, the above FOS based DCN architectures do not sufficiently consider the traffic locality, the flexibility, and cost-efficient pay-as-it grows approach. The size of the traffic locality can be measured by counting the most frequent communication between the neighboring TORs.

In a modular DC, there are tens or even hundreds of modular clusters (MCs). The majority of the traffic is exchanged inside the MC supporting hundreds of servers. With the help of a standard 40-feet shipping-container, the servers and switches can be packed together to form one MC. In this way, the operator can upgrade the DC by deploying more MCs or relocate the DC at new places conveniently with the moving of MCs. Besides, the modular DC has also the advantages of high system and power density, lower cooling and manufacturing cost, and flexibility. In this paper, we propose a novel modular DCN based on FOS formed in sphere topology (FOSphere) to fully address the traffic locality and the DC mobility requirements.

2. FOSphere network operation

The FOSphere architecture shown in Fig. 1(a) interconnects at most N^2+1 MCs, where N is the FOS radix. Each MC connects N^2 top of rack switches (TORs) as shown in Fig.1 (b). Therefore, the FOSphere can support a DCN with $N^2(N^2+1)$ TORs. Considering that each TOR interconnects 40 servers, the FOSphere can interconnect 4160 TORs and thus 166.400 servers employing distributed FOSs with only 8 port count.

The MC interconnection of the N^2 TORs (see Fig. 1 (b)) is implemented by two distributed genres of FOSs (G1_FOS and G2_FOS). The *i*-th G1_FOS (G_1_FOS_i) connects sequential N TORs forming groups where the index of the first TOR is equal to N×(*i*-1)+1. Namely, the index of the last TOR connected by G_1_FOS_i is N×*i*. While the *j*-

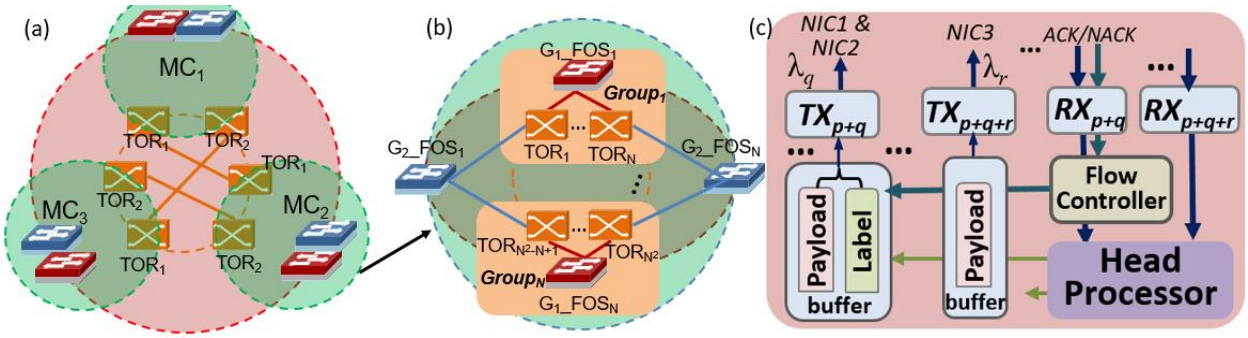

Fig. 1. (a)Downsize FOSphere DCN with 6 TORs, (b) MC interconnect architecture, (c) The schematic of the TOR.

th G2_FOS (G_2_FOS_j) connects *j*-th TORs in each group of N TORs. Note that at most two hops of FOS are needed to connect any TORs. As an example shown in Fig. 1(b), TOR_1 and TOR_{N2} can be connect via TOR_1→G_1_FOS_1→TOR_N→G_2_FOS_N→TOR_{N2}, or alternatively TOR_1→G_2_FOS_1→TOR_{N2-N+1}→G_1_FOS_N→TOR_{N2}. The interconnection between the MCs is implemented through direct connection between TORs of different MCs as shown in Fig. 1(a). More specifically, the *j*-th TOR (TOR_j) of *i*-th MC (MC_i) ($1 \le i \le j \le N$) connects *i*-th TOR (TOR_i) of (*j*+1)-th MC (MC_{j+1}). To clearly show the topology of modular FOSphere, we set a small FOSphere network with only 3 MCs as shown in Fig. 1(a). The TOR_1 and TOR_2 in the MC_1 connect TOR_1 in MC_2 and MC_3, respectively. While the TOR_2 in the MC_2 connects TOR_2 in MC_3.

Fig. 1(c) shows the schematic of the TOR. The TOR is equipped with three network interface cards (NIC). The NIC1 and NIC2 connect the TOR to the G_1_FOS and G_2_FOS, respectively, for intra-MC communications. NIC3 is used to directly connect the TOR to another TOR in a different MC. As shown in Fig. 1(c), there are *p*, *q* and *r* TRXs for the interconnection of G_1_FOS, G_2_FOS, and the TOR in different MC, respectively. The value of *p*, *q* and *r* can be set elastically based on the traffic locality to be served with guaranteed performance.

When multiple packets arrive at the TOR, they are first processed by the head processor and forwarded to the buffers associated with the corresponding NIC1, NIC2 or NIC3 based on the destination. An optical label is attached to the transmitted packets to determine the destined TOR if the next hop is FOS. While if the next hop is the directly connected TOR in different MC, no optical label is generated. The SOA broadcast&select based FOS can parallel processes multiple WDM input packets by using distributed controlled 1×N photonic switches. Benefitted from the modular structure and parallel processing of the WDM channel, the contentions among the N input ports can be solved in a distributed manner which results in port-count independent reconfiguration time of 20ns. More details on the FOS can be found in [4]. Optical fast flow control is implemented to solve the contentions happening at the FOS [7]. At the FOS, in case of contention, the packet with highest priority wins the contention, and a positive acknowledge signal (ACK) is sent back to the TOR to release the packet in the buffer. While the negative acknowledge signal (NACK) will be sent to the rest TORs to trigger a re-transmission. No optical flow control is needed for the direct inter-MC connections between TORs.

3. Simulation setup and results

We use the OMNeT++ to build the FOSphere network model simulation. In the simulations, each TOR supports 40 servers operating at 10Gb/s, and each server generates ON/OFF traffic independently [5]. The generated packets are buffered in the unit of cells with length of 64 bytes, and 25 cells with the same destination forms one optical packet whose preamble length is set as 125 bytes. The delay caused by the header processing and buffering at the TOR input is taken as 80ns and 51.2ns, respectively, based on previous measure employing a FPGA. The TOR's WDM transceivers (TRX) operate at 50 Gb/s, and the buffer size per TRX is 50 KB. The link distances between TOR and FOS, TOR and TOR are all 50 m.

Firstly, we investigate and compare the network performance of FOSphere with OPSquare under DC size of 10,000 servers. The radix FOS adopted to build the OPSquare and FOSphere DCN are 16 and 4, respectively. During the simulation, the traffic locality pattern adopted is: 50% intra-TOR traffic, 37.5% traffic are exchanged between 8 neighboring TORs (2 sequential groups in OPSphere), while the rest 12.5% traffic was transmitted to TORs in other MCs. Figure 2(a) shows that the latency of FOSphere is slightly higher than OPSquare when load is less than 0.3. This is because at low load the latency is mainly dominated by the link delay and more hops are traversed for the inter-MC traffic of FOSphere. However, at high load there are sufficient bandwidth inside the MC to handle the intra-MC traffic. Therefore, we can observe in Fig. 2(a) that at high load the packet loss ratio and average latency of FOSphere outperforms OPSquare. At load 0.4, the latency and packet loss of FOSphere are 4.1 μs and 2.6E-3, respectively.

Secondly, the cumulative distribution function (CDF) of the latency is investigated. Fig. 2(b) shows that the maximum server-to-server latency increase as the load increase due to the high contention probability at heavy load. When the load is 0.1, 99% of the latency is lower than 5.3 μs. Moreover, 90% of the latency is lower than 4.6 μs and 7.0 μs at load of 0.3 and 0.5, respectively. The maximum number of re-transmissions is not limit in the simulations. Limiting the maximum number of retransmissions would result in lower server-to-server latency especially at load higher than 0.5, but at the expense of high packet loss. The maximum latency is 78.2 μs at load of 1.

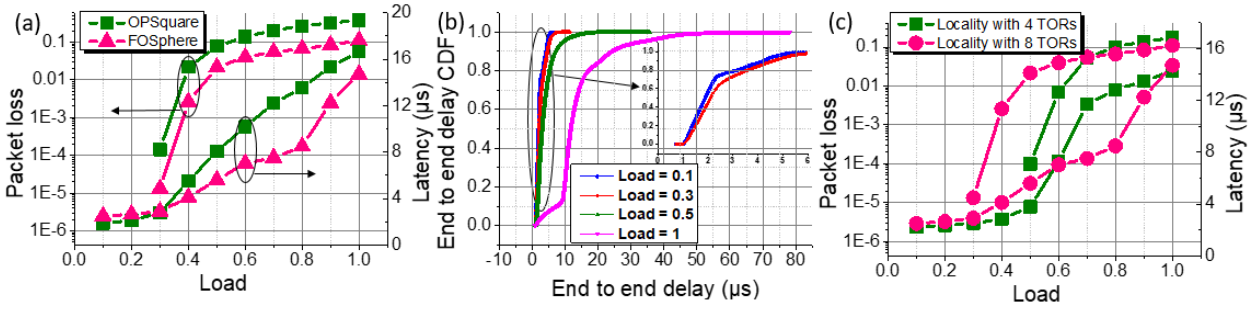

Fig. 2. (a) Performance comparison, (b) Latency CDF, and (c) Traffic locality.

The FOSphere network performance is also investigated as degree of traffic locality changes. We increase the degree of the traffic locality as 0.25 by tuning the inter-TOR traffic so that 37.5% traffic are exchanged between 4 neighbouring TORs in one group, while the rest 12.5% traffic was transmitted to TORs in other MCs. As shown in Fig. 3(c), both the latency and packet loss decreases as the traffic locality increases from 0.125 to 0.25 at low load. The reason is that, the traffic need to pass less hops as the traffic locality increases. However, this is not the case at high load when the network starts to saturate since a larger amount of contentions happens which results in grievous retransmission.

4. Cost and power consumption analysis

The cost and power consumption of FOSphere is compared with the electrical DCN architecture Fat-Tree and optical DCN architectures OPSquare and FOScube. The servers' costs are not considered as common to all architectures, and only the contributions of DCN components are considered. The cost and power consumption of the network components are reported in Table 1 [5]. The cost and power consumption of FOS increase nearly quadratic and linearly, respectively, with respect to the FOS radix. The cost of single-mode fiber and multi-mode fiber is 0.3 $/m and 0.9 $/m, respectively.

Tab. 1: Cost and power consumption of DCN components.

Components	Radix	Cost ($)	Power (W)
TRX (10 Gbit/s)	-	70	1
TRX (50 Gbit/s)	-	750	4
Electrical switch	≤128	20/per port	2/per port
	256	10922	622
	512	65532	2490
	1024	131064	5050
FOS	8×8	6220	441
	16×16	22860	985
	32×32	87980	2457
	64×64	345900	6937

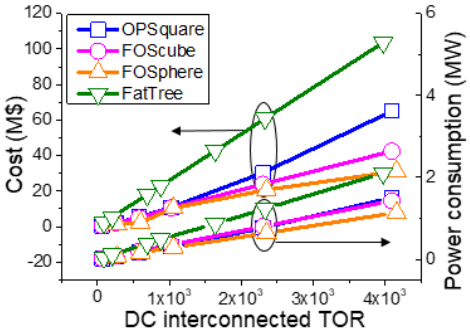

Fig. 3. Cost and power consumption comparison.

Figure 3 reports the cost and power consumption as the network size scales. For a network supporting around 4,000-TOR, FOSphere saves 45.9% and 24.1% power consumption, respectively, compared with Fat-Tree and OPSquare. The main reason is that a large part of the TRXs power consumption in Fat-Tree is eliminated in the optical DCN architectures. The power consumption of FOSphere also outperforms power consumption efficient FOScube. Similarly, the FOSphere solution has a cost saving of 70% and 52.2% with respect to Fat-Tree and OPSquare, respectively. Moreover, FOSphere has the minimal cost due to the adoption of low radix FOS.

5. Conclusion

We propose and investigate the network performance of a novel scalable modular DCN architecture FOSphere based on FOS. Assessment results demonstrate that FOSphere outperforms OPSquare, and more specifically, it achieves 4.1μs latency, 2.6E-3 packet loss at load of 0.4 under 10880-server. At load of 0.5, 90% of the latency is lower than 7.0 μs, and the tail value of the latency is 35.7μs. Moreover, FOSphere can achieve 45.9% power consumption and 70% cost saving, 24.1% power consumption and 52.2% cost saving for interconnecting around 4,000-TOR compared with Fattree and OPSquare, respectively.

Acknowledgements:

The authors thank the Olympics project (grant number ESTAR17207) for partially supporting this work.

References:

[1] C. V. Networking, "Cisco Global Cloud Index: Forecast and Methodology, 2016-2021. White paper," *Cisco Public, San Jose*, 2017.

[2] R. Proietti, Z. Cao, Y. Li, and S. J. B. Yoo, "Scalable and distributed optical interconnect architecture based on AWGR for HPC and data centers," *Conf. Opt. Fiber Commun. Tech. Dig. Ser.*, vol. 1, no. c, pp. 2–4, 2014.

[3] N. Farrington *et al.*, "Helios: a hybrid electrical/optical switch architecture for modular data centers," *Proc. ACM SIGCOMM 2010 Conf. SIGCOMM - SIGCOMM '10*, p. 339, 2010.

[4] F. Yan, W. Miao, O. Raz, and N. Calabretta, "OPSquare: A Flat DCN Architecture Based on Flow-Controlled Optical Packet Switches," *J. Opt. Commun. Netw.*, 2017.

[5] F. Yan, X. Xue, and N. Calabretta, "HiFOST: A Scalable and Low-Latency Hybrid Data Center Network Architecture Based on Flow-Controlled Fast Optical Switches," *IEEE/OSA J. Opt. Commun. Netw.*, vol. 10, no. 7, pp. 1–14, 2018.

[6] F. Yan, X. Xue, G. Guelbenzu, and N. Calabretta, "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.

[7] W. Miao, J. Luo, S. Di Lucente, H. Dorren, and N. Calabretta, "Novel flat datacenter network architecture based on scalable and flow-controlled optical switch system," *Opt. Express*, 2014.