Opportunistic Load Balancing in Optical Datacenter Networks using Spare Capacity

Yaxuan Ma, Yingxin Guo, Tong Ye
School of Electronic Information and Electrical Engineering,
Shanghai Jiao Tong University, Shanghai, China
{libmyx, guoyingxin, yetong}@sjtu.edu.cn

***Abstract*—A novel opportunistic load-balancing strategy via spare capacity is devised for datacenter networks with optical circuit switches deployed in the core layer. Simulation results show that our proposal outperforms previous schemes remarkably in delay performance.**

***Index Terms*—optical datacenter network, load balance, optical circuit switch**

I. Introduction

Optical circuit switches (OCSs), with their prominent advantages in capacity and power consumption [1], have been considered as a promising candidate of switching nodes for datacenter networks (DCNs). For example, Jupiter has delivered 5x capacity and 41% reduction in power consumption by constructing an aggregation-block-interconnection layer for the DCN using OCSs [2].

One of the big challenges in the design of OCS-based DCNs is how to cope with the spatiotemporal uncertainty of network traffic. The measurement studies in [3], [4] show that the traffic in each link exhibits an ON/OFF pattern with extremely different data rates in ON period and OFF period. Also, it is reported in [5], [6] that the traffic changes quite quickly both in rate and in active server pairs, yelling for frequent bandwidth reallocation. However, the reconfiguration time of the OCSs is relatively long (about tens of milliseconds), which makes the optical topology cannot reconfigure according to the rapid change of traffic pattern.

Load balancing mechanism was then introduced to avoid the frequent reconfiguration of OCSs in optical DCN. A well-known scheme is Valiant load balancing (VLB) strategy studied in [7], the key idea of which is to distribute the traffic of each block pair along all available lightpaths to smooth out the traffic uncertainty. However, the load balance in the VLB comes at the expense of bandwidth utilization. Almost all the traffic has to traverse two lightpaths, which nearly doubles the bandwidth consumption. To address this problem, Ref. [8] proposes a threshold-based load balancing scheme, called TROD. It divides the capacity of the lightpaths of each pair of blocks to two parts by a threshold. When the traffic rate of a block pair is higher than the threshold, the TROD performs load balancing for excessive traffic via the over-threshold capacity of all available lightpaths. As a result, more than half of the traffic can be transmitted via direct lightpaths. Though the TROD can enhance bandwidth utilization and reduce flow completion time (FCT), the bandwidth utilization of TROD is still not high. On one hand, part of the under-threshold capacity between a block pair will stay idle and thus be wasted when the traffic rate is under the threshold. On the other hand, additional over-threshold capacity is specially reserved for load balancing, and cannot be utilized by direct traffic.

In this paper, we extend the idea of [9] to propose a new load-balancing strategy for optical DCNs, named Opportunistic Load Balancing via Spare Capacity (OLB-SC). This strategy is motivated by the fact that the traffic leaving the same block bursts simultaneously with very small probability, which dramatically decreases with the growth of the number of blocks in the DCN. In other words, when the traffic on a lightpath is in the overflow state, most of the lightpaths that share the same source block are light loaded and leave a great proportion of capacity spared. The basic idea of the proposed OLB-SC is to make use of this spare capacity to perform load balancing, that is, to disperse the overflow traffic from the hot-spot lightpath to other intermediate blocks. Due to high bandwidth utilization, our simulations demonstrate that the OLB-SC can outperform the VLB and the TROD in delay performance.

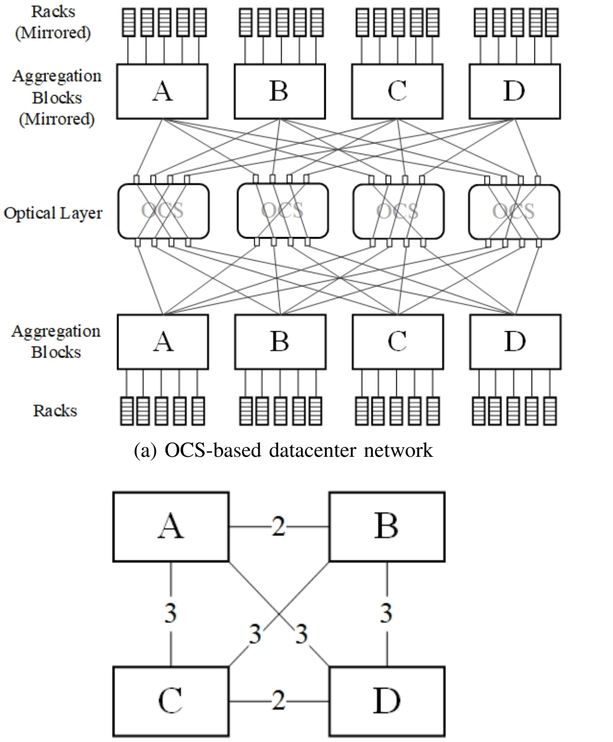
II. Opportunistic Load Balancing via Spare Traffic

The OCS-based datacenter network is shown in Fig. 1(a), where multiple Top-of-Rack (ToR) switches are connected to one aggregation block and the aggregation blocks are interconnected by OCSs. As mentioned before, OCSs can be reconfigured to adapt to different traffic demands, such that block pairs with higher traffic load can be allocated with more lightpaths. Fig. 1(b) depicts the corresponding logical topology, that is, the number of lightpaths allocated to each block pair, when the OCSs are reconfigured as in Fig. 1(a). For example, block A is connected to block B by the second and third OCS, so there are two bidirectional lightpaths between A and B.

Though the reconfiguration of OCSs allows the logical topology to change, it still can not adapt to the fluctuation of traffic in DCN. However, Ref. [5], [8] observes that the traffic of datacenter presents a weaker form of stability, which means

Tong Ye is corresponding author.

This work was supported by the National Science Foundation of China under Grant 62271306.



(a) OCS-based datacenter network

(b) Logical topology

Fig. 1. The physical structure and logical topology of datacenter network.

a logical topology could be computed based on the historical traffic patterns. According to this property, we propose OLB-SC, which includes two phases. In phase 1, we establish the logical topology on the aforementioned physical structure. The topology is established based on the historical traces, such that most traffic could be transmitted through the direct lightpath between each block pair. In phase 2, each block performs load balancing using spare capacity to mitigate the fluctuation of traffic.

A. Phase 1: Topology Establishment

In OLB-SC, the topology needs to provide enough capacity for most traffic. It also needs to make full use of all the optical layer resources to ensure all blocks have spare capacity when one of its lightpaths gets congested. To achieve this objective, we use the historical traces collected the previous day to determine the topology.

Let $X = [x_{ij}]_{|N|\times|N|}$ be the number of links allocated between block i and block j, $D = [d_{ij}]_{|N|\times|N|}$ be the average traffic rate from block i to block j calculated using the previous-day traces, where N is the block set. Let r be the bandwidth of each link and m be the maximal number of ports that a block can use to connect to the optical layer. The formulation is listed below. Constraint (2) states that the capacity of each lightpath should at least be sufficient for average traffic. And constraint (3)-(5) ensures that this logical topology is deployable on the physical structure. The optimization objective of this problem is to minimize the sum of the average link utilization of all lightpaths. This objective makes best effort to allocate more bandwidth to block pairs with higher payload. It also makes sure that the bandwidth resources can be fully utilized, providing more spare capacity for load balancing.

$$\begin{aligned}
\min \quad & z = \sum_{i\neq j} \frac{d_{ij}}{rx_{ij}}, && && (1)\\
s.t. \quad & rx_{ij} \geq d_{ij}, && i,j \in N, i \neq j, && (2)\\
& \sum_{i\neq j} x_{ij} \leq m, && j \in N, && (3)\\
& \sum_{j\neq i} x_{ij} \leq m, && i \in N, && (4)\\
& x_{ii} = 0, && i \in N. && (5)
\end{aligned}$$

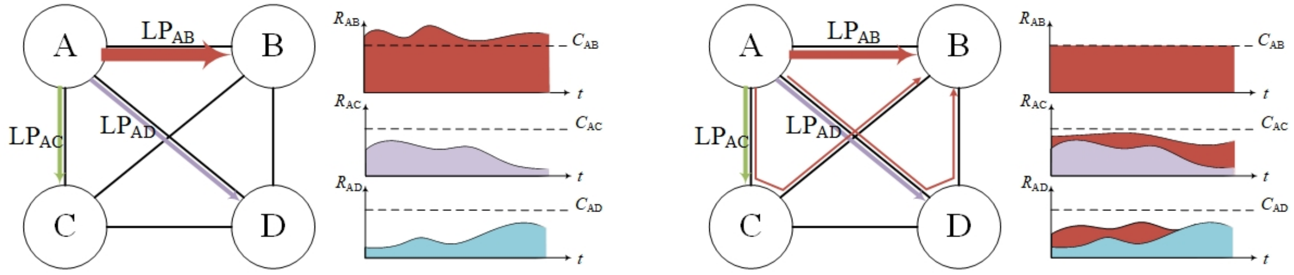
B. Phase 2: Load Balancing

After the logical topology is constructed based on the historical traffic, the bandwidth is able to meet most traffic demands as the bursts are handled by load balancing. Fig. 2 illustrates an example of the load balancing in OLB-SC. As Fig. 2 plots, the traffic from block A to block B is in burst state, while that from block A to block C and block D is not. In this case, it would be better to offload the excessive traffic from A to B to C and D via the spare capacity in lightpath (LP$_{\text{AC}}$) from A to C and that from A to D (LP$_{\text{AD}}$). This approach has the following advantages:

1) The capacity in LP$_{\text{AC}}$ and LP$_{\text{AD}}$ that stays idle can now be utilized;
2) It is unnecessary to reserve specific capacity in LP$_{\text{AC}}$ and LP$_{\text{AD}}$ to distribute the excessive traffic to C and D;
3) The excessive traffic from A to B does not have to be congested at A, and can be offered to C and D timely, such that the FCT can be reduced.

The detailed implementation of phase 2 is carried out inside each block without global information. To do this, each block equips two types of buffers, as Fig. 3 illustrates. The first type is called direct buffer. Each block installs a direct buffer for each destination block. The block sets a threshold for each direct buffer. The second type is called load balancing buffer, which is shared by all the destination blocks. Consider a packet a that originates from block i and will go to block j. The detail of opportunistic load balancing process is described as follows:

1) If packet a sees that the queue length of the direct buffer to block j is less than the threshold, a enters the direct buffer and will be transferred to block j when it becomes the head-of-line (HOL) packet.
2) Otherwise, packet a joins load-balancing buffer. When it becomes the m^{th} packet and there are n empty direct buffers ($m \leq n$), it will be randomly fed to one of these empty buffers and transmitted via the corresponding lightpaths to an intermediate block, say block k_1. The direct buffer of block i to block k_1 is empty means that the lightpaths from block i to block k_1 are now idle. Thus, offloading packet a from block i to block k_1 only



(a) LP$_{AB}$ is overloaded while LP$_{AC}$ and LP$_{AD}$ is not

(b) Excessive traffic is offloaded from LP$_{AB}$ to LP$_{AC}$ and LP$_{AD}$

Fig. 2. Principle of OLB-SC, where R_{AB} and C_{AB} stands for the traffic rate from A to B and the capacity of LP$_{AB}$.

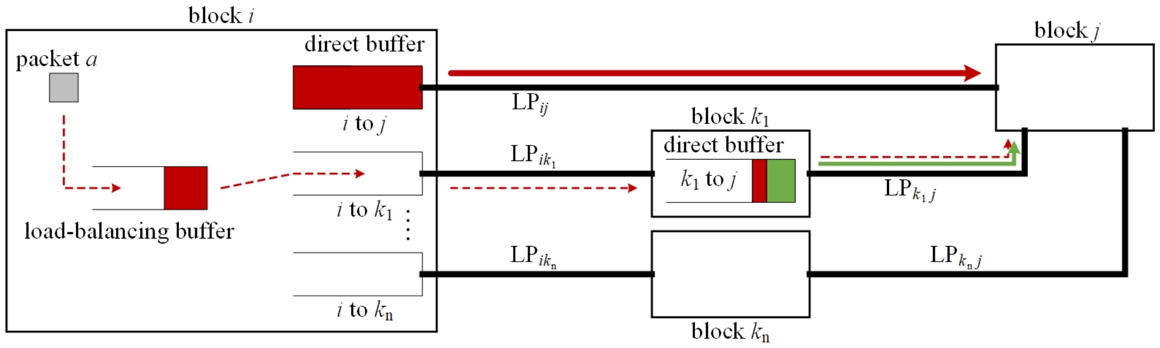


Fig. 3. Illustration of load balancing process of OLB-SC.

uses spare capacity of the lightpath from block i to block k_1. After packet a reaches block k_1, it will immediately join the direct buffer from block k_1 to block j, as Fig. 3 illustrates. Note that there is a chance that the direct buffer from block k_1 to block j is full when a arrives at block k_1. In this case, a packet in this direct buffer that does not experience the load balancing process will be squeezed into the load-balancing buffer of block k_1 to make space for packet a. This guarantees that each packet can reach the destination via at most one transit under the OLB-SC strategy.

In this scheme, the only parameter that needs to be further adjusted is the threshold of the direct buffer. If the threshold is too large, most of the packets have to wait in the direct buffer rather than being load-balanced to an intermediate block, when the lightpaths between a block pair are overloaded. Especially when it is set to infinite, the load balancing will never happen, and the OLB-SC becomes a scheduling scheme without load balancing. In this case, the delay performance will be bad. Clearly, the FCT declines with the increase of threshold, since the congested packets can be offloaded to some intermediate blocks timely. However, if the threshold is too small, supposedly zero, all packets will be offloaded to the intermediate blocks definitely. In this case, the FCT of OLB-SC would also be large since most packets need to traverse two lightpaths. Thus, an appropriate buffer threshold is necessary for the OLB-SC to play its role. However, our simulation results show that the threshold has a wide selection range, and will not be the obstruction of this scheme.

III. Implementation with P4 Switch

OLB-SC is highly implementable, and can be operated on the commercially available switches. In this section, we use P4 [10] switch as an example to describe how it should be implemented on an actual switch.

The overall implementation of OLB-SC on P4 switch is listed as follows:

- Ingress Parser: Information about the packet will be extracted here, including destination mac address and DSCP value. Here the DSCP value indicates whether this packet has been load balanced. If the DSCP value is Assured Forwarding (AF), then this packet has already experienced one transit and needs to reach the destination block directly. Else the packet can be load balanced according to buffer threshold.
- Ingress Control: Control block is composed mainly of match-action units called table. Each table matches the keys to the desired action. Here we apply three tables one after another.
    1) Table port_match: This table matches the destination address to output ports by longest prefix matching. The matching result will be stored in field outputPort and can be obtained by other tables.
    2) Table buffer_match: This table matches outputPort to its corresponding queue. The result is logged in field qid.

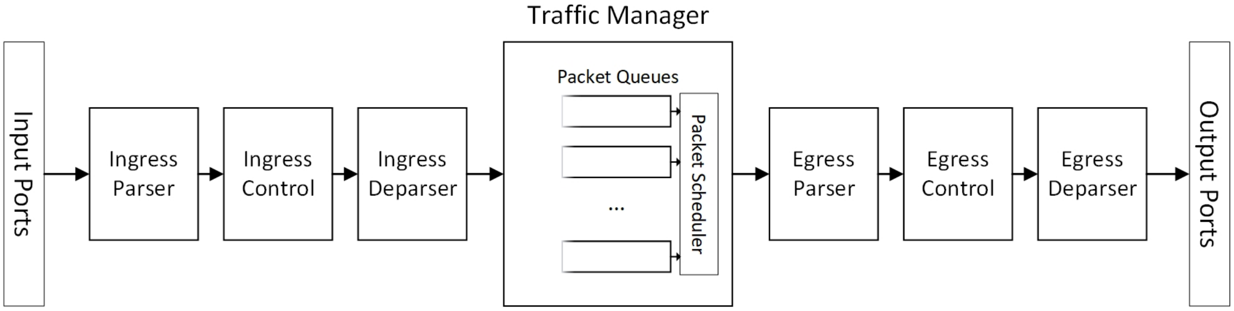


Fig. 4. Brief structure of a P4 switch.

3) Table lb_match: This table determines whether the packet needs to be load balanced to the load-balancing buffer lb_qid. If the DSCP value is default, then the qid of the packet will be rewritten to lb_qid if the depth of its original qid exceeds the threshold.

- Traffic Manager: Packets arriving at traffic manager will be stored in packet queues and will be dequeued by packet scheduler when they become the HOL packet. When an output port finishes sending out the previous packet, packet scheduler will first dequeue the packets from the corresponding queue of the port. It will only dequeue packets from lb_qid when the corresponding queue is empty, i.e., the lightpath is idle.
- Egress Deparser: If the qid of the incoming packet is lb_qid, then egress deparser will rewrite the DSCP field of this packet to AF.

IV. Performance Evaluation

Our evaluation is carried out on the extension of NetBench, a packet-level simulator. We use flow completion time (FCT) as the evaluation metric. FCT is one of the key indicators of network performance in datacenter network, and is closely related to user experience. However, the size of flows in real datacenters varies greatly. To eliminate the interference caused by flow size, many researchers now turn to FCT slowdown, which is defined as a flow's actual FCT normalized by its ideal FCT when the network only has this flow [11]. In this simulation, we compare OLB-SC with other load-balancing strategies like TROD and VLB. To facilitate the comparison, we also simulate an ideal scheme, where the blocks are interconnected by a high-capacity switch with ultra-fast reconfiguration speed. Optimal values are selected for all tunable parameters, i.e., buffer threshold for OLB-SC and p-value for TROD, so as to observe the best performance of each scheme. The load-balancing strategies typically suffer from out-of-order delivery. We thus enable the selective ACK (SACK) option in TCP as Ref. [8] suggests, to avoid retransmitting packets that arrive out of order.

A. *Buffer threshold influence on OLB-SC*

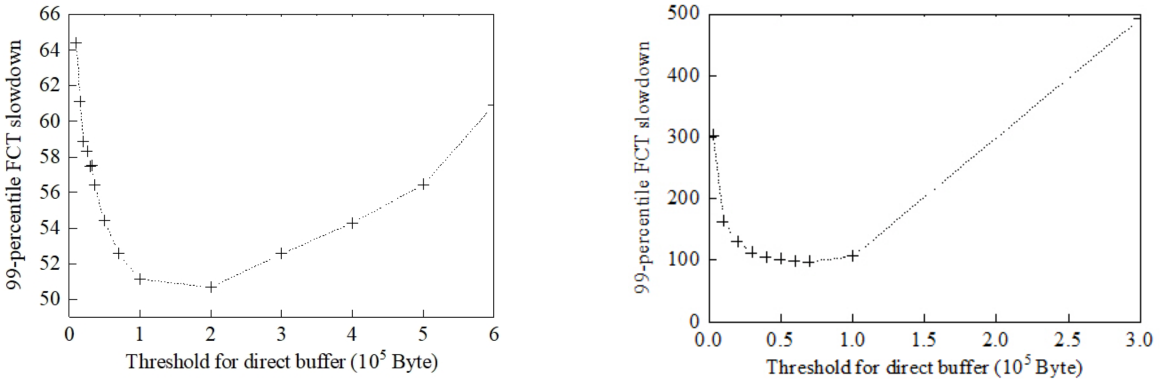
Fig. 5 first studies the FCT slowdown of our strategy changing as the selected threshold of direct buffers changes. Here we use two kinds of traffic. One is the ON/OFF traffic generated according to the testing data provided by Microsoft. The other is the real traffic data colleted from Facebook's traces. Fig. 5(a) and Fig. 5(b) sketches the 99-percentile FCT slowdown of ON/OFF traffic and Facebook traffic respectively. In both graphs, the curve has a flat region, where the 99-percentile FCT slowdown of corresponding thresholds is minimal, which is consistent with our analysis in the last section.

B. *FCT performance under ON/OFF traffic*

According to the observations in Ref. [3], we generate the ON/OFF traffic for each block pair. We assume that the statistical characteristics of the traffic for all block pairs are the same, such that we can observe how different load-balancing strategies perform under the same optical topology. Herein, we consider a DCN with 4 blocks and 4 lightpaths between each block pair. Without loss of generality, we consider the case where the capacity of each lightpath is 10 Gb/s. The traffic rate of each block pair fluctuates from 0 Gb/s to 140 Gb/s, and the average traffic rate is 17 Gb/s.

Fig. 6(a) depicts the performance of each scheme under the generated ON/OFF traffic. Obviously, the ideal scheme achieves the lowest FCT slowdown. For the rest of the schemes, OLB-SC is the closest to the ideal scheme in FCT performance, which can be interpreted as follows: (1) OLB-SC can make full use of all available bandwidth resources by getting rid of traffic rate threshold; (2) OLB-SC load balances packets through spare capacity, thus adding no further burden on the lightpath; (3) OLB-SC can timely sense the burstiness using the threshold buffer. TROD performs worse than OLB-SC, since it allocates dedicated bandwidth for direct and load-balanced traffic, hindering mutual bandwidth utilization. VLB performs the worst, as it almost sends all traffic through two lightpaths, causing extra bandwidth fee and packet delay.

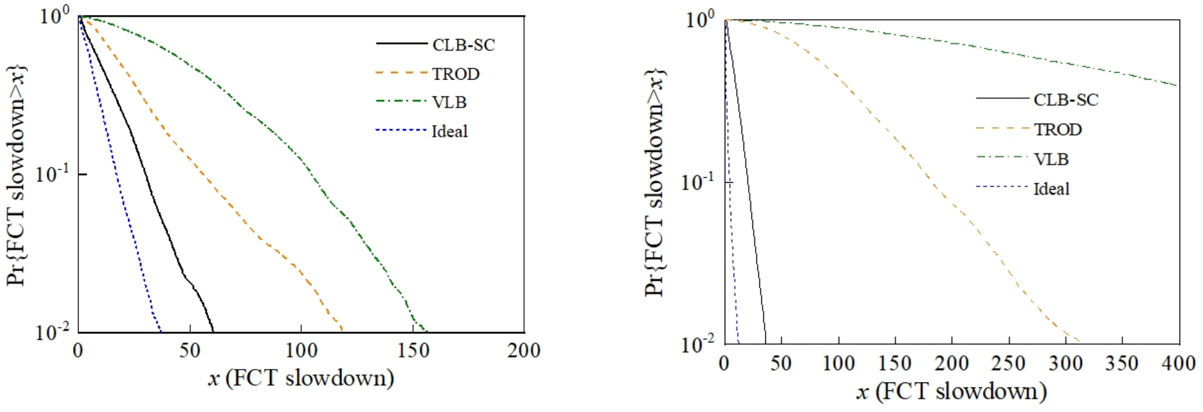
In Fig. 6(b), where the block population in the DCN increases from 4 to 8, the difference of delay performance between the OLB-SC and other two strategies becomes larger. This can be interpreted as follows. The probability that all the paths originating from the same block burst at the same time decreases very fast with the increase of block population in the DCN. In this case, a block would have more opportunity to find spare capacity for load balancing when some paths from this block are overloaded.



(a) Optimal buffer threshold under ON/OFF traffic

(b) Optimal buffer threshold under Facebook traffic

Fig. 5. Threshold selection for direct buffer.



(a) DCN with 4 blocks

(b) DCN with 8 blocks

Fig. 6. FCT performance comparisons under ON/OFF traffic.

C. FCT performance under Facebook traffic

We also compare the four schemes under the traffic traces from Facebook. We provide each block with a port bandwidth that is equal to half of the peak traffic rate. The results in Fig. 7 are similar to the FCT results under ON/OFF traffic, OLB-SC still performs better than TROD and VLB. This shows that OLB-SC can fully utilize bandwidth and timely balance congestion under different traffic patterns.

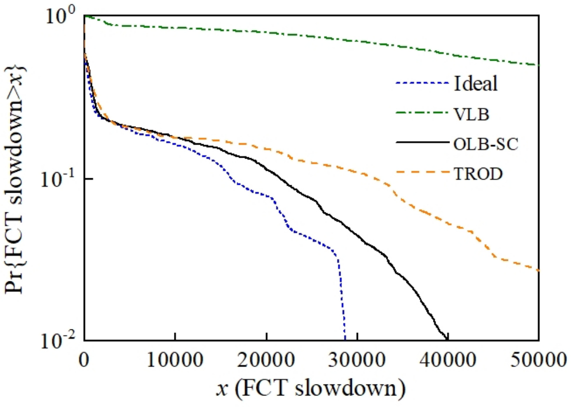


Fig. 7. FCT performance comparisons under Facebook traffic.

V. Conclusion

This paper proposes a load balancing strategy called OLB-SC for optical DCNs. OLB-SC has better capacity utilization and congestion alleviation. OLB-SC can already be implemented on the commercially available P4 switches. Simulation results show that OLB-SC greatly surpasses TROD and VLB in FCT, and can achieve almost optimal delay performance under both ON/OFF traffic and Facebook traffic.

References

[1] N. Farrington, *Optics in data center network architecture*. University of California, San Diego, 2012.

[2] L. Poutievski, O. Mashayekhi, J. Ong, A. Singh, M. Tariq, R. Wang, J. Zhang, V. Beauregard, P. Conner, S. Gribble *et al.*, "Jupiter evolving: Transforming google's datacenter network via optical circuit switches and software-defined networking," in *Proceedings of the ACM SIGCOMM 2022 Conference*, 2022, pp. 66–85.

[3] T. Benson, A. Akella, and D. A. Maltz, "Network traffic characteristics of data centers in the wild," in *Proceedings of the 10th ACM SIGCOMM conference on Internet measurement*, 2010, pp. 267–280.

[4] R. Kapoor, A. C. Snoeren, G. M. Voelker, and G. Porter, “Bullet trains: A study of nic burst behavior at microsecond timescales,” in *Proceedings of the ninth ACM conference on Emerging networking experiments and technologies*, 2013, pp. 133–138.

[5] A. Roy, H. Zeng, J. Bagga, G. Porter, and A. C. Snoeren, “Inside the social network’s (datacenter) network,” in *Proceedings of the 2015 ACM Conference on Special Interest Group on Data Communication*, 2015, pp. 123–137.

[6] S. Kandula, S. Sengupta, A. Greenberg, P. Patel, and R. Chaiken, “The nature of data center traffic: measurements & analysis,” in *Proceedings of the 9th ACM SIGCOMM conference on Internet measurement*, 2009, pp. 202–208.

[7] R. Zhang-Shen and N. McKeown, “Designing a fault-tolerant network using valiant load-balancing,” in *IEEE INFOCOM 2008-The 27th Conference on Computer Communications*. IEEE, 2008, pp. 2360–2368.

[8] P. Cao, S. Zhao, M. Y. The, Y. Liu, and X. Wang, “TROD: Evolving from electrical data center to optical data center,” in *2021 IEEE 29th International Conference on Network Protocols (ICNP)*. IEEE, 2021, pp. 1–11.

[9] T. Ye, J. Zhang, T. T. Lee, and W. Hu, “Deflection-compensated birkhoff–von-neumann switches,” *IEEE/ACM Transactions on Networking*, vol. 25, no. 2, pp. 879–895, 2016.

[10] P. Bosshart, D. Daly, G. Gibb, M. Izzard, N. McKeown, J. Rexford, C. Schlesinger, D. Talayco, A. Vahdat, G. Varghese *et al.*, “P4: Programming protocol-independent packet processors,” *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 3, pp. 87–95, 2014.

[11] Y. Li, M. Alizadeh, M. Yu, R. Miao, and F. Kelly, “HPCC: high precision congestion control,” in *the ACM Special Interest Group*, 2019.