LightFD: A Lightweight Flow Detection Mechanism for Traffic Grooming in Optical Wireless DCNs

Amer Al-Ghadhban, Abdulkadir Celik, Basem Shihada, and Mohamed-Slim Alouini
Computer, Electrical, and Mathematical Sciences & Engineering (CEMSE) Division
King Abdullah University of Science & Technology, Thuwal, 23955-6900, KSA.

***Abstract*—State of the art wireless technologies have recently shown a great potential for enabling re-configurable data center network (DCN) topologies by augmenting the cabling complexity and link inflexibility of traditional wired data centers (DCs). In this paper, we propose an optical traffic grooming (TG) method for mice flows (MFs) and elephant flows (EFs) in wireless DCNs which are interconnected with wavelength division multiplexing (WDM) capable free-space optical (FSO) links. Since handling the bandwidth-hungry EFs along with delay-sensitive MFs over the same network resources have undesirable consequences, proposed TG policy handles MFs and EFs over distinctive network resources. MFs/EFs destined to the same rack are groomed into larger rack-to-rack MF/EF flows over dedicated lightpaths whose routes and capacities are jointly determined in a load balancing manner. Performance evaluations of proposed TG policy show a significant throughput improvement thanks to efficient bandwidth utilization of the WDM-FSO links. As MFs and EFs are needed to be separated, proposed TG requires expeditious flow detection mechanisms which can immediately classify EFs with very high accuracy. Since these cannot be met by existing packet-sampling and port-mirroring based solutions, we propose a fast and lightweight in-network flow detection (LightFD) mechanism with perfect accuracy. LightFD is designed as a module on the Virtual-Switch/Hypervisor, which detects EFs based on acknowledgment sequence number of flow packets. Emulation results show that LightFD can provide up to 500 times faster detection speeds than the sampling-based methods with %100 detection precision. We also demonstrate that the EF detection speed has a considerable impact on achievable EF throughput.**

I. Introduction

Data Centers (DCs) have become an intrinsic element of emerging technologies such as big data, artificial intelligence, cloud services, cellular infrastructure, content delivery; all of which entails interconnected and sophisticated computing and storage resources. In order to meet ambitious demands of these emerging technologies, data center networks (DCNs) are required to improve their bandwidth efficiency, reliability, and delay sensitivity in a great extend. Scalability of DCs is generally expected to accommodate a huge number of servers to supply adequate speed and bandwidths, which yields a significant cabling complexity as network equipment of today's DCs communicates over either unshielded twisted pair cables or fiber-optic wires. Moreover, wired DCs lack the flexibility to adjust link capacities which are designed to be constant and uniform for the entire DCN. Since intra-rack and inter-rack traffic patterns are quite dissimilar for different rack pairs [1], [2], wired DCNs with uniform link capacities are either under-utilized or over-utilized with respect to rack pairs that exchange light and heavy traffic, respectively.

Fortunately, wireless DCs can leverage the state-of-art wireless communication technologies in order to obtain flexible and re-configurable DCN topologies [3], [4]. Wireless DCs can augment the cabling complexity by replacing cables with wireless links, alleviate the bandwidth inefficiency by flexibly allocating the transmission powers to adapt link capacity for dynamically changing traffic conditions, and reduce the maintenance costs and overhead. In particular, free-space optical (FSO) links can provide multi-terabit capacity over line-of-sight (LoS) collimated light beams which inherently yield an interference-free communication and improve the physical layer security. When it is combined with wavelength division multiplexing (WDM) techniques, FSO can further provide a large number of links among the rack pairs, which is referred to as high fan-out and desirable for better network management. For instance, outdoor WDM-FSO links have shown to achieve 1.28 Tbps (32x40 Gbps) capacity on 32 wavelengths over 212 meters distance [5]. Thanks to controlled and acclimatized DCN environment, indoor FSO links can achieve even better performances as they are not subject to hostile outdoor optical channel impairments such as scintillation, pointing error, and atmospheric turbulence, etc.

DC flows are typically classified as bandwidth-hungry elephant flows (EFs) or delay-sensitive mice flows (MFs). While a considerable portion of the DCN traffic volume is carried out by EFs, the majority of flow arrivals are MFs. Intuitevely, MFs experience intolerable delays when they are routed along the same path of EFs [6]. At this very point, traffic grooming (TG), which can be referred to as the aggregation of subwavelength flows onto high-speed lightpaths [7], becomes a fundamental network function since bandwidth demands of flows can be much lower than the available WDM channel capacities. TG is also useful to avoid unnecessary delays caused by the computational and control overhead of handling flows individually.

Even though TG is a mature field of research for passive optical networks [8], it is first presented for wired DCNs in [9], [10] where flows are simply groomed into three classes of wavelengths which are confined for broadcasting within racks and higher layer switches. However, proposed grooming method considers fixed wavelength capacity and do not deal with the flow characteristics of real-life DCNs. In [11], we conceptualized TG for MFs in WDM-FSO based wireless

DCNs consisting of optoelectronic switches. After formulating the optimal TG problem, a suboptimal TG policy is designed for mice flows (MFs) whereas elephant flows (EFs) are carried out separately via server-to-server express lightpaths without going through any grooming operation. Emulation results have shown that the proposed TG method provides superior performance in terms of throughput and flow completion times. Unlike our previous work [11] which assumes apriori flow classifications and employs electrical grooming merely for MFs, this paper considers optical TG for both MFs and EFs such that flows are combined to create larger rack-to-rack (R2R) MFs and EFs. Based on arrival rate, size, and completion time request of flows, lightpaths are provisioned by jointly determining the routes and capacity allocations accounting for balancing load across available FSO links.

Besides designing a high-performance TG policy, accuracy and speed of EF detection mechanism also play a crucial role in achievable network throughput. After its arrival, EFs must be detected as soon as possible with high accuracy in order to prevent undesired consequences of handling EFs and MFs over the same network resources. However, existing flow classification solutions have major drawbacks in terms of detection speed, controlling overhead, and/or precision. For example, OpenSample leverages sFlow packet sampling to provide measurements for both network load and individual flows [12], which has the following disadvantages: 1) Since sampling picks up flows in a randomized fashion, it is possible to have several (no) samples from classified (unclassified) flows, 2) Since a single packet cannot provide an insight into flow classes, sampling based methods are required to collect multiple samples from the same flow, which is hard to ensure if flows arrive and complete much faster than the sampling rate, and 3) A decision might be useless if samples are obtained through the end of EF completion time. On the other hand, Planck [13] utilizes port-mirroring which definitely requires extra overhead and cabling can miss some EFs due to the limited buffer size. Similar to sampling-based methods, Planck can also attempt to detect already classified flows.

In order to overcome these deficiencies, we propose an in-network lightweight flow detection (LightFD) scheme by following the current trend of relaxing network from complex functions by utilizing location privileges of virtual-switch/hypervisor in DCNs. Hence, the proposed scheme can be regarded as a module installed on the virtual-switches/hypervisors which detects EFs based on ACK sequence numbering of transmission control protocol (TCP). Emulation results show that LightFD can detect EFs in the order of milliseconds with %100 accuracy.

The rest of the paper is organized as follows. The node architecture, network topology, and TG policy design are explained in Section II. LightFD is then introduced in Section III. Thereafter, the emulation results are presented in Section IV. Finally, conclusions are drawn in Section V.

II. Optical Traffic Grooming Design for DCNs

This section first explains the considered node architecture and DCN topology, then presents the proposed TG policy.

A. Node Architecture and Network Topology

We consider a two-tier DCN architecture where every leaf layer (lower-tier) switch is connected to each of the spine layer (top-tier) switches as shown in Fig. 1. The leaf layer comprises N edge switches (ESs) which connect the servers within a rack. Denoting the spine/leaf ratio as η, ηN core switches (CSs) in the spine layer interconnects all racks to each other in a full-mesh manner. LoS links of DCN can be implemented using the physical topology shown in Fig. 1 where optical transceivers of ESs are directed to CS transceivers located at the top.

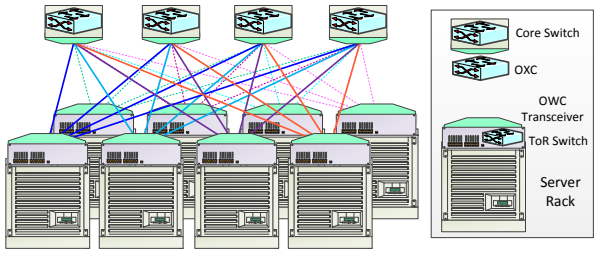


Fig. 1: Proposed topology for $N = 4$ and spine/leaf ratio of $\eta = 1/2$.

Each ES is considered as an optical cross-connect (OXC) with N I/O ports such that received optical beam at each input port is first demultiplexed into W wavelengths, then fed into the connection matrix which determines the connectivity between OXC's I/O ports, thereafter output lines of the connection matrix is groomed (combined) via a multiplexer at each wavelength based on the proposed TG policy, and finally output of the OXCs are forwarded to the relevant optical transmitter as per the routing protocol. On the other hand, CSs are modeled as routers without any grooming operation as explained in the next section.

WDM-FSO links are realized by laser-diode transmitters and photo-diode receivers, which are mounted on metallic breadboards and aligned to each other to form point-to-point FSO links. Thanks to the WDM, each wavelength can be treated as a parallel channel and assumed to operate on intensity-modulation direct-detection (IM-DD) scheme. The channel capacity of wavelength ω between optical transceivers k and l is given by [14]

$$C_{kl}^{\omega} = B\frac{1}{2}\log\left(1+\frac{e(h_k^l)^2(E_{k,l}^{\omega})^2}{2\pi}\right),\ \omega\in[1,W] \quad (1)$$

where B is the bandwidth of a single wavelength, $E_{k,l}^{\omega}$ is the light intensity allocated to wavelength ω, $h_k^l \triangleq \rho h_{k,l}^{\ell} h_{k,l}^{a} h_{k,l}^{p}$ is the optical channel gain which is assumed to be constant throughout a transmission block since the optical channel variations are very slow compared to the symbol duration [15], ρ is the detector responsivity, $h_{k,l}^{\ell}$ is the optical path loss,

$h^a_{k,l}$ is the atmospheric turbulence, and $h^p_{k,l}$ is the pointing error. Due to hardware and safety concerns, signal intensity has to satisfy a total and individual average intensity restrictions given by $\sum_\omega \mathrm{E}^\omega_{k,l} \leqslant \mathrm{E}_T$ and $\mathrm{E}^\omega_{k,l} \leqslant \mathrm{E}$, respectively.

B. Optical Traffic Grooming Policy

TG comprises of three joint subproblems: 1) virtual topology design, i.e., provisioning lightpaths over the physical topology; 2) assignment of wavelengths to the lightpaths; and 3) grooming policy design and routing the groomed traffic on the virtual topology. Since each of these subproblems are NP-hard [16], TG is also an NP-hard problem which belongs to mixed-integer linear programming (MILP) class. Due to the dynamic and heavy traffic characteristics of DCNs, obtaining an optimal TG policy requires impractical time complexity even for small-scale DCs. Hence, fast yet high-performance suboptimal solutions are necessary to implement TG for WDM-FSO based DCs.

Current energy and wavelength availability of the physical DCN topology is defined by graph $\mathcal{G}_p(\mathcal{V}, \mathcal{W}, \mathcal{E})$ where $\mathcal{V}$ is the set of nodes, $\mathcal{E}$ presents available light intensity, and $\mathcal{W}$ denotes the wavelength availability of FSO links. $\mathcal{G}_p$ is always kept updated over a dedicated broadcasting wavelength which is used merely for the control signaling. On the other hand, virtual topology is defined by lightpaths which is a pair of a path defined on the physical topology and a wavelegnth on this physical path. We must note that a wavelength on a certain link cannot be shared by different lightpaths due to the collision constraints. Moreover, a lightpath must operate on the same wavelength along the routing path as OXCs are assumed not to be capable of wavelength conversion. Since MFs experience sever delay if they routed along the same path of EFs [6], we treat them separately during grooming and routing processes. Hence, the virtual topology is represented by two disjoint graphs: $\mathcal{G}_e(\mathcal{V}, \mathcal{P}_e, \mathcal{W}_e, \mathcal{C}_e)$ for EFs and $\mathcal{G}_m(\mathcal{V}, \mathcal{P}_m, \mathcal{W}_m, \mathcal{C}_m)$ MFs, where $\mathcal{P}_e$/$\mathcal{P}_m$, $\mathcal{W}_e$/$\mathcal{W}_m$, and $\mathcal{C}_e$/$\mathcal{C}_m$ represent lightpaths' routes, assigned wavelengths, and capacities, respectively. In order to avoid any unnecessary computational and control overhead due to a large number of flow arrivals within a DCN, rack-to-rack (R2R) lightpaths are predetermined for MFs and EFs thanks to high fanout provided by the WDM-FSO links. That is, each R2R pair are dedicated with two distinct lightpaths: one for MFs and the other for EFs.

All incoming flows are first assumed to be MF for two reasons: 1) The policy maker is flow agnostic and does not aware of the flow classification upon arrival, and 2) The DCN traffic characteristics tell us that majority of the arriving flows are MF whereas a significant portion of data is carried out by EFs. Based on this assumption, flows are groomed in the ESs based on the following three-step optical grooming policy [11]: 1) *S2S grooming* takes place in servers such that all flow arrivals destined to a certain server is combined into a single flow, 2) *Server-to-Rack (S2R) Grooming* is also handled by servers such that S2S flows are further groomed according to destination rack. In this way, all flows outgoing to the same rack is groomed into a single flow and transferred to the ESs, and 3) *R2R grooming* occurs in ESs where received S2R flows from different servers are then groomed according to their destination racks to obtain R2R flows. Groomed flows are then directed to the relevant optical transmitter based on the predetermined R2R-MF routing paths. Therefore, CSs are not required to implement any optical grooming since they forward the laser beams to the ES transceivers of the destination racks. Once a flow is detected to be an EF, the source server immediately stops feeding its packets into the MF virtual topology and transfer it to the EF virtual topology where EFs are also optically groomed in three-step similar to the MF grooming.

Denoting the set of host within rack i by $\mathcal{R}_i$, arrival rates of MFs and EFs from $s \in \mathcal{R}_i$ to $s' \in \mathcal{R}_j$ are assumed to follow Poisson distribution with rates $\lambda^m_{ss'}$ and $\lambda^e_{ss'}$, respectively. Assuming that flow arrivals are independent from each other, overall arrival rate for the R2R flows also follow a Poisson distribution with the composite rate of $\Lambda^m_{i,j} = \sum_{s\in\mathcal{R}_i}\sum_{s'\in\mathcal{R}_i}\lambda^m_{ss'}$ and $\Lambda^e_{i,j} = \sum_{s\in\mathcal{R}_i}\sum_{s'\in\mathcal{R}_i}\lambda^e_{ss'}$, respectively. Since MF are generally delay sensitive, wavelength capacities are first guaranteed for the R2R-MFs based on the required flow completion time τ_m and flow size F_m as

$$E^\omega_{k,l} = \left(\frac{\left(2^{\frac{2\Lambda^m_{i,j}F_m}{\tau_m B}} - 1\right) 2\pi}{e(h^l_k)^2} \right)^{1/2} \leqslant E, \forall k, l \in \mathcal{P}^j_i, \omega \in \mathcal{W}^j_i \tag{2}$$

where $\mathcal{P}^j_i$ and $\mathcal{W}^j_i$ represent the routing path and assigned wavelength for R2R-MF between racks i and j, respectively. Notice that (2) follows from equating the wavelength capacity in (1) to the required exact R2R-MF capacity, $\frac{\Lambda^m_{i,j}F_m}{\tau_m}$, for all links along the path. At this point, we must note that intensity allocation and route determination is implemented in a joint manner such that the number of R2R-MFs and resulting capacity load is balanced by distributing the R2R lightpaths across all available FSO links. The residual light intensity of FSO links is then exploited by the R2R-EFs based on the previous joint intensity allocation and route determination method. In case of multiple R2R-EFs, whose overall requested light intensity exceed the residual light intensity, compete for a certain FSO link, their capacity allocation is determined in a proportional fair manner.

III. LightFD: Lightweight Flow Detection

Speed and accuracy of the flow detection mechanisms play a crucial role in reaping the full benefit of the proposed TG policy. Even though initially treating arriving flows as MFs is quite practical since the majority of the flows are MFs, EFs must be detected and separated as soon as possible in order to meet the delay-tolerance demands of MFs, which is addressed in the following subsections.

A. Overview of DCN Flow Characteristics

As the TCP carries out almost 99% of the DCN traffic [17], its attributes are leveraged by many existing flow classification

methods in order to build accurate and fast flow detection mechanisms [12], [18]. A TCP flow starts with three-way handshaking (i.e., SYN, SYN+ACK and ACK) to set the initial sequence number and prepare some connection parameters. Since the SYN, SYN+ACK, and FIN/RST flags appear only once in each flow for the connection establishment/termination purposes, we regard them as low-frequency phase of the TCP. Therefore, we exploit packets of the low-frequency for enabling the controller to maintain global network view by learning useful network statistics, such as number of flows in every path, flow arrival-rate, etc. Following the three-way handshaking, the source sends the data packets labeled with a sequence number, while the receiver sends ACK packets marked with an acknowledgment number to inform the source that the transmitted bytes have been successfully received, which is regarded as high-frequency phase of the TCP. Amount and frequency of these ACK packages are proportional to the flow size and allocated flow capacity, respectively. Since headers of the high-frequency ACK messages encapsulates valuable information about flow size, we exploit them in order to develop a fast and lightweight flow detection mechanism.

B. The Elephant-Flow Detection Algorithm

A single packet of every TCP flow offers partial information about the flow size which is useless if it is individually considered. However, when two packets are captured from the same TCP flow, one can discover the number of bytes that have been transmitted between the two captured packets. This technique is fundamental idea of the existing flow detection solutions (e.g., OpenSample-TCP and Planck) where the difference between the TCP sequence numbers of captured packets and time of capture are used to measure the link utilization. While OpenSample-TCP employs the conventional packet sampling (i.e., sFlow), Planck utilizes port-mirroring to accelerate the measurement process. Also, Planck needs to find an alternative way to directly connect the centralized collector with every edge switch to avoid adding an extra overhead and congestion from forwarding the mirrored packets through the data network. Moreover, the captured samples could be from the end of the flow and repeatedly from the same flow.

As shown in Fig. 2, the LightFD scheme is embedded in a kernel module which is installed on the hypervisor of edge servers or in the virtual switches. A hypervisor is an operating system (OS) layer which presents the virtual machine OS and the server (i.e., the physical host machine where the hypervisor runs) OS to each other. In other words, the hypervisor abstracts the hardware resources (e.g., CPU, memory, hard disk, network interface card, etc.) of the host machine to the VMs. Thus, the communications between the VMs and the hardware resources have to go through the hypervisor. The virtual switch is a software switch (e.g., Open virtual Switch (OvS)) connects the VMs of an individual server together with the ToR. The kernel of these two systems provides a degree of control on exchanging packets and a full access to their headers and unencrypted payload. For EF detection, LightFD exploits two main components: collector and classifier. The

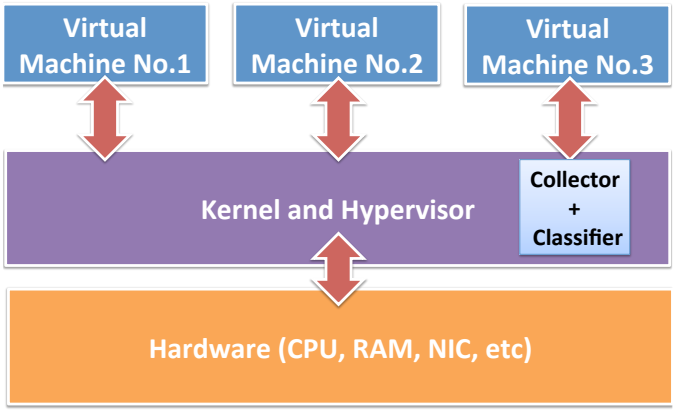


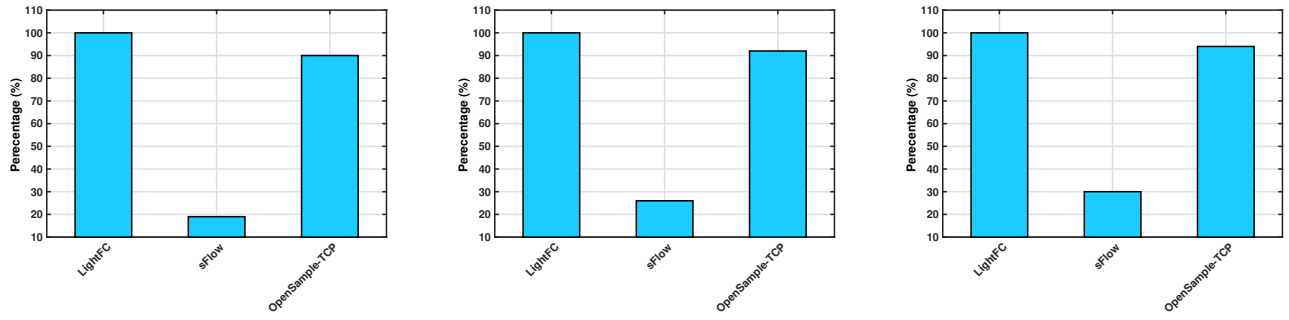
Fig. 2: The proposed LighFD module installed on the hypervisor.

collector is responsible for learning general flow information from the messages of the low-frequency phase. Moreover, it has a flow-information table to store the flow information (e.g., source/destination IP/MAC addresses and TCP sequence numbers). On the other hand, the classifier is accountable for detecting EFs from headers of the high-frequency phase messages.

Rather than using packet sampling, LightFD benefits from the privileges of its position in the virtual-switch/hypervisor to read the TCP header of every transmitted packet from the hosted VMs. The collector reads the initial sequence number (ISN) of every TCP flow, which is the first half of the required information to classify a flow, and stores it in the flow-information table. Likewise, the classifier needs to compare the ISN with every captured packet from the same flow until a threshold value is reached. However, instead of capturing the data packets themselves, LightFD classifier is programmed to read the headers of ACK packets. For instance, if the classifier reads ACK_1 and then ACK_t after a while, the number of transmitted bytes between these two ACKs is simply calculated as $t-1$. The ACK sequence number only presents the bytes that have been successfully received, that is, the lost or out-of-order packets are not counted. Therefore, the classifier needs only to compare the number of bytes with a predefined threshold value th to classify a flow as an EF. Alternatively, the count of the transmitted ACK messages can also be used as a method to detect elephant flows. Once $t-1 > th$ is satisfied, the flow is considered as an EF and the subsequent packets are marked to be forwarded via the virtual topology of EFs and the controller is reported regarding its classification change. That is, unlike the existing detection algorithm, proposed mechanism does not suffer from classification re-attempts. It is also free of overheads and random sampling drawbacks. Since it operates at the kernel level, it can detect the EFs in an expeditious manner with full precision.

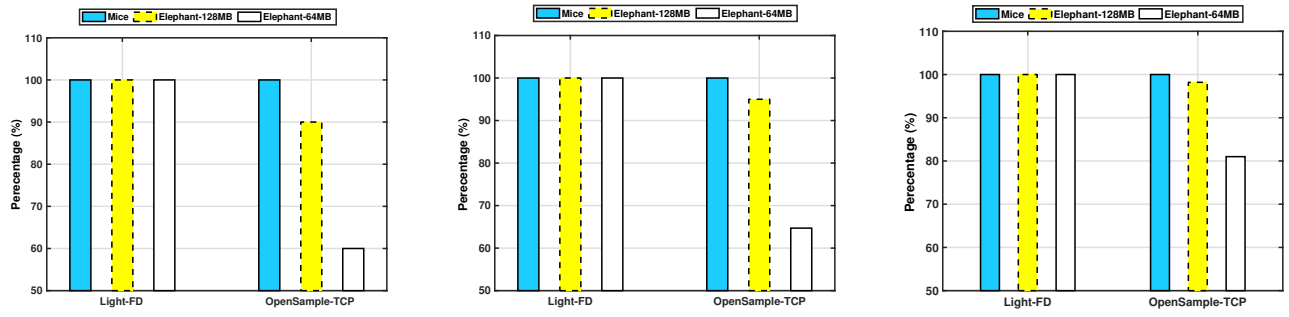
IV. Emulation Results

We conducted our evaluation using Mininet emulator [19] and POX [20] controller. The controller and Mininet topology



(a) 100 Flows (b) 500 Flows (c) 1000 Flows

Fig. 3: The accuracy of LightFD in different traffic configurations during the mix scenario.



(a) 10 Flows (b) 100 Flows (c) 500 Flows

Fig. 4: The accuracy of LightFD in different traffic configurations during the pure scenario.

are installed on the same physical machine. The leaf-spine topology has 8 CSs and 8 ESs each with 40 hosts/servers. The OpenSample-TCP and sFlow are configured with 1-to-1000 sampling-rate. The traffic is steered between the first and second subnet to ensure a certain number of flows (i.e., 100, 500, and 1K). The flows are generated by Iperf and arrive according to an exponential distribution with a mean of 2 ms. The percentage of mice is set to 90% of all flows. The EF detection threshold is determined as 1 MB. The size of MFs and EFs are set to 100KB and 128MB, respectively, unless explicitly stated otherwise.

A. Detection Accuracy and Speed

The EF detection algorithms may have true-negatives, (i.e., EF is considered as MF), false-positives, (i.e., considering a MF as an EF). In network load balancers, the reporting of few MFs as EFs are somewhat acceptable, however, the percentage of true-negative incidents is critical because keeping the EF on the path of R2R-MFs can cause a sever delay for MFs. In this evaluation, we measure the percentage of true-negative incidents of LightFD, OpenSample-TCP, and sFlow during different traffic loads and communication scenarios.

We investigated LightFD with different flow scenarios; pure MFs, pure EFs, and mixed MFs and EFs. We started with the evaluation of the mixed scenario as shown in Fig. 3 where the accuracy of LightFD is always 100% (i.e., 0% true-negative incidents) for a various number of flows. For the sake of

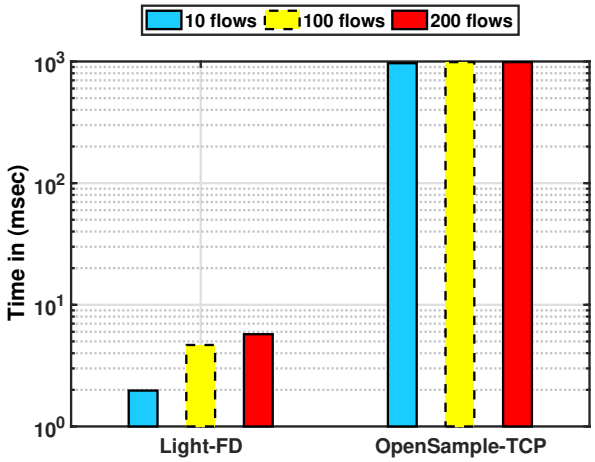
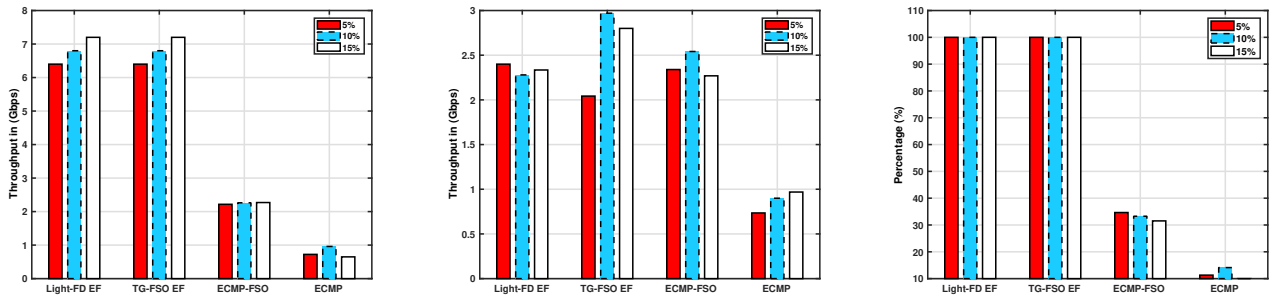


Fig. 5: Detection speed of LightFD and OpenSample-TCP during various link loads.

an accurate testing between the potentials of LightFD and the proponents, we follow these algorithms' suggestion in EF detection threshold settings and Mininet configurations, e.g., the link speed. Since existing solutions' accuracy varies for different threshold values, we compare the LightFD with the best case performance of the existing solutions. Fig. 3 clearly shows that sFlow provide quite a low accuracy compared to



(a) Comparison of MF throughput for different algorithms.

(b) Comparison of EF throughput for different algorithms.

(c) Comparison of MF FCT for different algorithms.

Fig. 6: The impact of LightFD on network throughput and FCT during different traffic scenarios.

the LightFD. However, OpenSample-TCP obtains around 90% accuracy for varying number of flows. Therefore, we consider only the OpenSample-TCP for comparison in the remainder of the evaluation.

Similar to the mixed case, LightFD achieved %100 accuracy in all pure MF and EF cases as shown in Fig. 4. Likewise, OpenSample-TCP reaches %100 accuracy in pure MF cases, which is not possible for pure EF cases. Apparently, the accuracy increases with the delay in the forwarded traffic. The delay could be sourced from the flow size or a congestion in the used link. Accordingly, the accuracy of OpenSample-TCP is higher with large flow size (128 MB) than the small one (64 MB). Also, the results show some increase when we increased the number of transmitted flows. For instance, the OpenSample-TCP accuracy in the 500 scenarios is better than the results of the 10 scenarios. This is mainly because of that the OpenSample-TCP has more chance to detect EFs by obtaining more samples when the flow size (hence, the flow completion time (FCT)) increases. In other words, when flow size or FCT is small, OpenSample-TCP do not have a chance to sample multiple packets to detect EFs. This also explains why it achieves %100 accuracy in pure MF case.

Next, we examine the speed of EF detection under different network loads; 10, 100, or 200 flows. The detection time can be extracted from a timer function on the classifier code or the network interface of the controlling unit. The average detection speeds are illustrated in Fig. 5 where EF detection time for OpenSample-TCP ranges from 975 to 990 ms according to the load on the path. On the other hand, LightFD provides 494$\times$ to 172$\times$ faster detection speeds than the OpenSample-TCP. We must note that the detection of EFs is slower than measuring the link utilization because the EF detection algorithm needs to wait until the size difference between the first and second captured packets is larger than the EF threshold value.

B. Network Throughput Results

Throughout this subsection, the size of MFs and EFs are set to 50KB and 128MB, respectively. Since the links in Mininet are limited by the processing capacity of the host machine, we configured the FSO-links with 10 Gbps and the cabled links with 1 Gbps which means FSO links are 10 times faster than wired DCN links. Each FSO link consists of 4 wavelengths, which are realized as virtual links in Mininet. Since the emulator is limited in DCN size, optical channel gains are not distinguishably different due to similar link distances and thus assumed to be identical without loss of generality. We use MapReduce to mimic workloads of real DCNs whose shuffle-phase communication pattern has k servers from every rack communicate with another k servers in a different rack. For instance, the hosts in $\mathcal{R}_i$ have been divided into two sets and every set has k servers, e.g. , 4 MF and one EF. Each of k server is communicating with k servers of rack j, $j \neq i$, different than other k servers of the same rack. For accurate results, we evaluated the routing algorithms with various percentage of EF: 5%, 10%, and 15%. We set $k = 20$ in all evaluation scenarios, that is, the number of MFs in the 5% scenario is 19 and the number of EFs is one.

For the network throughput results, we consider the following cases: 1) *Equal-Cost-Multi-Path routing (ECMP) [21]* is a widely used DCN routing method which uses the packet header information, such as the IP/MAC addresses and TCP port numbers, as a key for a hash function. The outgoing path is the output hash value modulo the number of outgoing paths. This strategy splits the flows among available paths. Since the header information for an individual flow is the same during the session, the packets of the same flow are always forwarded via the same path; 2) *ECMP-FSO* is an ECMP routing algorithm supported by the FSO technology. In this routing method, the link capacity is equally divided between the wavelengths, thath is, the capacity of every wavelength is fixed to 2.5 Gbps. Each flow was assigned to a single wavelength. However, when flows are more than the available number of lightpaths, the packets of the waiting flows are enqueued until a lightpath is available for transmission; 3) *TG-FSO* refers to the proposed 3-step optical TG algorithm with apriori knowledge of flow classifications as in [11]; and 4) *LightFD* is the proposed 3-step optical TG algorithm employing the proposed flow detection mechanism.

Since the time constraint is 1ms, the demand for 5%, 10%, and 15% scenarios are 7.6Gbps, 7.2Gbps, and 6.8Gbps, re-

spectively. As we initially treat all flows as MFs, EF detection delay has no impact on MF performance as shown in Fig. 6a. That is why LightFD achieves the same throughput of TG-FSO for MFs. Moreover, both of them outperformed ECMP and ECMP-FSO cases thanks to improved bandwidth efficiency of the proposed TG approach. On the other hand, Fig. 6b shows that the impact of time elapsed for flow detection has a distinguishable impact on throughput even if the fast and accurate EF detection performance of LightFD. Referring to Fig. 5, one can easily infer that employing the OpenSample-TCP scheme in proposed TG algorithms can severely deteriorate the overall network throughput because of the low detection speeds. Finally, Fig. 6c demonstrates how the LightFD and TG-FSO satisfied the flow completion demand of MFs, while about 70% of the MFs in ECMP-FSO and exceeded the time constraint and about 88% of the MFs in ECMP exceeded the time constraint (i.e., complete after the 1ms time constraint).

V. Conclusions

In this paper, we considered optical TG of EFs and MFs in WDM-FSO based wireless DCNs. Since handling EFs on the same network resources of MFs severely impacts the MF performance, proposed TG policy treat TG of EFs and MFs separately. Emulation results show that proposed TG policy can improve the overall network throughput thanks to more efficient utilization of link capacities. Since TG performance heavily depend on the flow classification speed and accuracy, a novel flow detection mechanism is also proposed to mitigate the deficiencies of the existing sampling and port-mirroring based algorithms. Performance evaluations clearly shows that the proposed EF detection mechanism can provide perfect detection accuracy up to 500 times faster detection speed than the sampling-based solution.

References

[1] A. Roy, H. Zeng, J. Bagga, G. Porter, and A. C. Snoeren, "Inside the social network's (datacenter) network," *SIGCOMM Comput. Commun. Rev.*, vol. 45, no. 4, pp. 123–137, Aug. 2015.

[2] 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 conference*. ACM, 2009, pp. 202–208.

[3] M. Ghobadi, R. Mahajan, A. Phanishayee, N. Devanur, J. Kulkarni, G. Ranade, P.-A. Blanche, H. Rastegarfar, M. Glick, and D. Kilper, "Projector: Agile reconfigurable data center interconnect," in *Proceedings of the 2016 ACM SIGCOMM Conference*. ACM, 2016, pp. 216–229.

[4] N. Hamedazimi, Z. Qazi, H. Gupta, V. Sekar, S. R. Das, J. P. Longtin, H. Shah, and A. Tanwer, "Firefly: A reconfigurable wireless data center fabric using free-space optics," in *Proc. the ACM SIGCOMM*, 2014, pp. 319–330.

[5] E. Ciaramella, Y. Arimoto, G. Contestabile, M. Presi, A. D'Errico, V. Guarino, and M. Matsumoto, "1.28 terabit/s (32x40 gbit/s) wdm transmission system for free space optical communications," *IEEE J. Sel. Areas Commun.*, vol. 27, no. 9, pp. 1639–1645, Dec. 2009.

[6] H. Zhang et al., "Resilient datacenter load balancing in the wild," in *Proceedings of the Conference of the ACM Special Interest Group on Data Communication*, ser. SIGCOMM '17, 2017, pp. 253–266.

[7] R. Ul-Mustafa and A. E. Kamal, "Design and provisioning of wdm networks with multicast traffic grooming," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 4, p. 53, 2006.

[8] S. Huang and R. Dutta, "Dynamic traffic grooming: the changing role of traffic grooming," *IEEE Commun. Surveys Tuts.*, vol. 9, no. 1, pp. 32–50, First 2007.

[9] G. C. Sankaran and K. M. Sivalingam, "Scheduling in data center networks with optical traffic grooming," in *proc. IEEE 3rd Intl.Conf. Cloud Netw. (CloudNet)*, Oct. 2014, pp. 179–184.

[10] ——, "Optical traffic grooming-based data center networks: Node architecture and comparison," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 5, pp. 1618–1630, May 2016.

[11] A. Celik, A. Al-Ghadhban, B. Shihada, and M. Alouini, "Design and provisioning of optical wireless data center networks: A traffic grooming approach," in *in Proc. IEEE Wireless Commun. Netw. Conf., WCNC, Barcelona, Spain, Apr. 14-18*, 2018.

[12] J. Suh et al., "Opensample: A low-latency, sampling-based measurement platform for commodity sdn," in *Proceedings of the 2014 IEEE 34th International Conference on Distributed Computing Systems*, ser. ICDCS '14, 2014, pp. 228–237.

[13] J. Rasley et al., "Planck: Millisecond-scale monitoring and control for commodity networks," in *Proceedings of the 2014 ACM Conference on SIGCOMM*, ser. SIGCOMM '14, 2014, pp. 407–418.

[14] A. Lapidoth, S. M. Moser, and M. A. Wigger, "On the capacity of free-space optical intensity channels," *IEEE Trans. Inf. Theory*, vol. 55, no. 10, pp. 4449–4461, Oct 2009.

[15] A. Chaaban, Z. Rezki, and M. S. Alouini, "Fundamental limits of parallel optical wireless channels: Capacity results and outage formulation," *IEEE Trans. Commun.*, vol. 65, no. 1, pp. 296–311, Jan. 2017.

[16] K. Zhu and B. Mukherjee, "Traffic grooming in an optical wdm mesh network," *IEEE J. Sel. Areas Commun.*, vol. 20, no. 1, pp. 122–133, Jan 2002.

[17] T. Benson et al., "Understanding data center traffic characteristics," *ACM SIGCOMM Computer Communication Review*, vol. 40, no. 1, pp. 92–99, 2010.

[18] A. R. Curtis et al., "Mahout: Low-overhead datacenter traffic management using end-host-based elephant detection," in *INFOCOM, 2011 Proceedings IEEE*, April 2011, pp. 1629–1637.

[19] N. Handigol, B. Heller, V. Jeyakumar, B. Lantz, and N. McKeown, "Reproducible network experiments using container-based emulation," in *Proc. the 8th Intl. Conf. Emerg. Netw. Exper. Tech.* ACM, 2012, pp. 253–264.

[20] POX. [Online]. Available: http://www.noxrepo.org/pox/about-pox/.

[21] C. Hopps, "Analysis of an equal-cost multi-path algorithm," *RFC 2992, Internet Engineering Task Force*, 2000.