Asynchronous Optical Traffic Offloading of Hybrid Optical/Electrical Data Center Networks

Tong Ye, *Member, IEEE*, Jianke Li, Xiaodan Pan, and Tony T. Lee, *Fellow, IEEE*

Abstract—In recent years, hybrid optical/electrical data center network has been considered a promising interconnection technology for large-scale data centers, since it can efficiently provide sufficient bandwidth. A key issue in hybrid optical/electrical data center networks is finding a way to offload burst traffic through optical circuit switches (OCSs), such that the burst traffic can be offloaded timely while the system operation overhead is low. This article extends the idea of Mahout to propose a local-push asynchronous optical traffic offloading strategy, in which each ToR switch offloads the traffic via the OCSs if its queue length is larger than a preset threshold, and transfers the traffic back to electrical packet switches (EPSs) when the backlog is empty. To seek a proper threshold, we develop a fluid-flow model to analyze the performance of the proposed traffic offloading strategy, from which we demonstrate that there is a trade-off between the mean delay of the traffic and the system operation overhead in a typical data center network. Based on such a trade-off, we provide a rule to select the buffer threshold. We show via simulation that the proposed optical traffic offloading strategy with the threshold selection rule outperforms C-through.

Index Terms—Hybrid optical/electrical, data center, traffic offloading, fluid-flow model

1 Introduction

As a component of the infrastructure of the information society, data centers play a vitally important role in the Internet [1], [2]. Data centers provide different types of information services, such as Web service, E-tailer, video service, data storage, data processing, and data analysis. Data centers perform these service functions via a large number of servers interconnected by data center networks. Therefore, improving the performance of data center networks is one of the keys to enhancing the information service capability of data centers.

One of the biggest challenges in the design of data center networks is managing burst traffic. It is well-known that the traffic within data centers is highly bursty due to the diversity of services [1], [2]. In a mega data center, there are thousands of racks of servers. Each pair of top of rack (ToR) switches in the data center network has a little traffic most of the time, but may generate a lot of traffic occasionally. As [3] reported, about 95 percent of traffic flows are mice flows, the size of which is less than 1 MB, and only 5 percent of flows are elephant flows, the size of which is larger than 100 MB. Therefore, in a time period, more than 95 percent of the traffic in the data center is generated by 5 percent of the ToR switches [2], [4], [5], [6]. Mice flows only require a little bandwidth but are delay-sensitive, while elephant flows need a large bandwidth but are delay-insensitive. In traditional data center networks where there are electrical packet switches (EPSs) only, the mice flows and the elephant flows are treated in the same way. However, as a kind of burst traffic, the elephant flows result in unacceptable queueing delay and serious packet loss due to temporary creation of full buffers in the data center network [7], [8]. Though this problem can be solved by bandwidth over-provisioning, i.e., the network assigns the bandwidth to users according to their peak traffic, this leads to a large bandwidth waste since the peak traffic rate rarely appears [9], [10].

Recently, the concept of hybrid optical/electrical data center networks [9], [10], [11], [12] has been proposed to deal with the burst traffic in data centers. Fig. 1 shows a typical hybrid optical/electrical data center network, where each ToR switch connects with all the EPSs and optical circuit switches (OCSs) in the core layer. The basic idea is to combine the advantage of EPSs and OCSs to cope with different types of traffic. The EPS can perform fine-grained switching since its reconfiguration time is very short, but its bandwidth is limited and it can only provide a relatively small throughput. Thus, the EPS is quite suitable for small but delay-sensitive traffic, such as mice flows. On the other hand, the biggest advantage of the OCS is that it can provide high capacity, though its reconfiguration time is much longer than that of the EPS. For example, the switching time of the silicon micro-electromechanical system (MEMS) based 64×64 optical crossbar presented in [13] is 0.91 μs, and that of the silicon MEMS-based 240×240 optical crossbar reported in [14] is 0.8 μs. Therefore, the OCS can be used to offload coarse-grained but delay-insensitive traffic, such as elephant flows. This paper refers to the procedure that uses optical bandwidth to offload the elephant flows as optical traffic offloading.

The main issue of optical traffic offloading is how to timely allocate optical bandwidth to the elephant flows to avoid network congestions [9], [10], [11], [12]. Currently, the resource allocation of hybrid optical/electrical data center

- *T. Ye, J. Li, and X. Pan are with the State Key Laboratory of Advanced Optical Communication Systems and Networks, Shanghai Jiao Tong University, Shanghai 200240, China. E-mail: {yetong, mrdingjay, pxd0506}@sjtu.edu.cn.*
- *T. T. Lee is with the Chinese University of Hong Kong (Shenzhen), Shenzhen 518172, China. E-mail: tonylee@cuhk.edu.cn.*

Manuscript received 8 May 2019; revised 18 Apr. 2020; accepted 1 May 2020. Date of publication 0 . 0000; date of current version 0 . 0000.
(Corresponding author: Tong Ye.)
Recommended for acceptance by K. Chen.
Digital Object Identifier no. 10.1109/TCC.2020.2992489

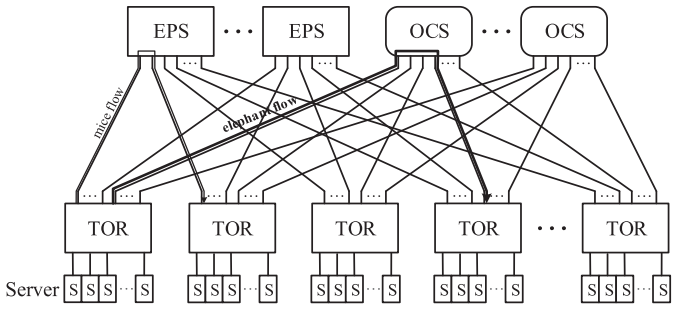


Fig. 1. A typical hybrid optical/electrical data center network.

networks is implemented by a central controller [9], [10], [11], [12]. If the controller is able to collect more state information about all communications, such as real-time data rate fluctuation, it can allocate bandwidth more accurately and timely, and keep the communication delay as small as possible. However, the collection of large amounts of information increases system operation overhead [15]. On one hand, the state information collection consumes the communication bandwidth. On the other hand, the controller needs computation resources and takes time to process the state information and make decisions. For example, an NOX controller can process 30000 requests per second, as [16] mentions. Therefore, a good traffic offloading strategy is highly desirable, such that the system can react to traffic fluctuation accurately and timely to reduce the mean delay, while incurring little system operation overhead.

1.1 Previous Work

Until now, there have been roughly three types of strategies to offload burst traffic: controller pulling strategy [9], [17], [18], [19], sampling method [20], [21], [22], [23], and local-push scheme [10], [24], [25] in this paper.

The controller pulling strategy was first proposed in [18]. In this scheme, the controller periodically collects the statistic information of traffic flows from the ToR switches, based on which the controller allocates paths for elephant flows such that the bandwidth utilization can be maximized. This method was then extended and applied to hybrid optical/electrical data center networks [9], [17], [19]. In [9], the time is divided into scheduling cycles. In each cycle, the controller collects the traffic information and generates a traffic matrix, based on which it calculates a configuration for the OCS to maximize network throughput. Unlike in [9], the controller in [19] determines a set of configurations for the OCS, and the OCS, in turn, runs these configurations in each scheduling cycle. The main disadvantage of such a scheme is that the controller needs to poll all the ToR switches during each information collection. This procedure is not only time-consuming but also requires a lot of communication bandwidth between the controller and the switches. As [9] shows, the polling procedure can even last for hundreds of milliseconds, which is too long for fast-changing traffic demands [24]. In addition, calculating the optimal OCS reconfiguration for all the flows would take a long time [9].

Another method is the sampling method proposed in [20], [21], [22], [23]. In this method, the ToR switch periodically samples its outgoing packets, for example, one sample out of every 1000 packets [20], and sends the sampled packet to the controller. If the controller continually receives the sampled packets from the same flow, it determines this flow as an elephant flow, and then assigns additional bandwidth to offload the traffic of this flow. This sampling method can detect elephant flows accurately. However, compared to the controller pulling strategy, the sampling method consumes a lot of communication bandwidth [24], because the controller pulling strategy only sends the flow statistics to the controller whereas the sampling method transmits the sampled packets. In addition, the sampling method imposes a heavy processing burden on the controller since the controller has to analyze every sampled packet to decide which flow it belongs to [24]. Thus, this method still does not apply to the hybrid optical/electrical data center network.

A typical reprehensive of the local-push scheme is Mahout in [25], which was proposed for the traffic offloading in electrical data center networks. In this scheme, each server allocates a buffer for each flow and monitors its queue length. Once the queue length of a flow exceeds a preset threshold, the server identifies this flow as an elephant flow and marks its packets. When the first marked packet arrives at the ToR switch that the server attaches to, it is forwarded to the controller, which allocates a new path to offload this elephant flow. This scheme is superior to the above two strategies in the following three aspects. First, this scheme only requires a little bandwidth overhead between the ToR switches and the controller, since the ToR switch only needs to push the detection information of elephant flows to the controller. Second, the processing overhead of the controller could be small since the controller only has to make a decision when the elephant flow is detected through buffer monitoring. Third, the controller can timely offload traffic once it receives the report from the ToR switches. Another kind of local-push scheme is C-through in [10], which performs optical traffic offloading in the hybrid optical/electrical data center network also based on the information of local buffer occupancy. Unlike Mahout [25], C-through is a kind of cyclic synchronous offloading scheme, which divides the time into scheduling cycles. Consider a data center with N ToR switches. At the beginning of each cycle, the controller collects the buffer occupancies of all ToR pairs to form a traffic matrix, based on which it builds up lightpaths for up to N ToR switch pairs using maximum weighted matching (MWM) algorithm. Within this cycle, the traffic of N selected ToR pairs is transmitted by the lightpaths, whereas that of other ToR pairs is routed through the EPS network. C-through has several drawbacks. First, it may lead to bandwidth waste, since the chosen ToR pairs in each cycle may have different amount of data to send and the cycle time is determined by the most-loaded ToR pair among them. Second, the traffic of the ToR pairs that are loaded but not selected by the MWM algorithm has to wait at least one more cycle to be offloaded, and thus may suffer from a large delay. Third, it is not scalable and cannot be applied to large-scale networks, since the MWM algorithm is typically time consuming [10].

It is clear that buffer threshold selection is quite important for the local-push scheme [25]. If the threshold is too small, the ToR switch (or the server) will claim too many flows as elephant flows, which increases the information exchanges between the ToR switches and the controller, the

processing burden of the controller, and the reconfiguration frequency of the data center network. As Section 2 shows, the network reconfiguration is another kind of operational overhead. This is especially true in optical circuit switching networks, since it takes several microseconds to reconfigure the OCS. On the other hand, if the threshold is excessively large, elephant-flow detection will be insensitive and the traffic offloading cannot be triggered timely, which will worsen the delay performance. Therefore, for the choice of threshold, there is a trade-off between the delay performance and the system overhead. However, Ref. [25] selected the buffer threshold only by experience. Hence, it is worth determining a principle for proper threshold selection.

1.2 Our Work

In this paper, we apply the idea of Mahout to hybrid optical/electrical networks and propose a local-push asynchronous optical traffic offloading strategy, such that the burst traffic can be offloaded timely while the system overhead is small. Our traffic offloading strategy is implemented by each ToR switch. In this scheme, each ToR switch maintains a buffer and a threshold of the queue length for the traffic to each destination ToR switch. The ToR switch informs the controller to set up a lightpath via the OCS to offload the burst traffic when the queue length exceeds the threshold, and notifies the controller to tear down the lightpath when the backlog in the buffer is cleaned up.

This paper aims to find a systematic method to select a proper threshold for the optical traffic offloading scheme, such that the mean delay of the traffic can meet the system requirements while the operation overhead could be small. To achieve this goal, we develop a fluid-flow model to delineate the burst traffic and the optical traffic offloading process. We derive the mean delay and the lightpath set-up frequency that is used to measure the operation overhead in this paper. We show there is a trade-off between the mean delay and the lightpath set-up frequency in a typical data center network, from which we give a threshold selection rule for the optical traffic offloading scheme. We demonstrate that our optical traffic offloading strategy can be applied to commercial data center networks.

In summary, the contribution of this paper is as follows:

1. Propose a local-push asynchronous optical traffic offloading strategy based on the idea of Mahout,
2. Develop a fluid-flow model for the asynchronous optical traffic offloading strategy, and
3. Devise a threshold selection rule for the asynchronous optical traffic offloading strategy.

The rest of this paper is organized as follows. In Section 2, we introduce the optical traffic offloading scheme for hybrid optical/electrical data center networks. In Section 3, we develop a fluid-flow model to analyze the optical traffic offloading scheme, and derive the close-form expressions of the mean delay and the lightpath set-up frequency. Based on the results in Section 3, Section 4 shows how the traffic burstiness affects the performance, from which we provide the threshold selection rule. In Section 5, we show the effectiveness of our selection rule through case studies. Section 6 concludes this paper.

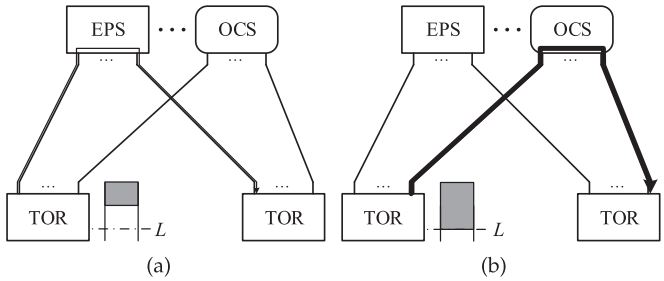


Fig. 2. Optical traffic offloading Process: (a) Electrical service state, (b) Optical service state.

2 Optical Traffic Offloading Process

As Section 1 shows, the traffic in data center networks is highly bursty by nature. According to the test data of several data centers in the wild by Microsoft Research [1], the traffic of all the data centers exhibits an ON-OFF pattern. The mean ON period is on the order of seconds and the mean OFF period is almost 20 to 70 seconds. The purpose of optical traffic offloading, as this paper discusses, is to deal with burst traffic in data center networks.

Fig. 2 illustrates the optical traffic offloading strategy this paper considers. When a ToR switch communicates with another ToR switch via the EPSs, it assigns the bandwidth according to the average input traffic rate, instead of the peak traffic rate, since our optical traffic offloading strategy deals with the burst traffic via the OCSs. The ToR switch allocates a buffer for this communication, such that all the packets to this destination can be queued in this buffer. The ToR switch sets a threshold for the queue length and monitors the buffer during the communication. When the queue length is less than the threshold, the traffic is transmitted via the EPSs according to the flow table of the ToR switch. Buffering the data at the ToR switch has the advantage that the ToR switch can react to the elephant flow immediately when the queue length excesses the threshold. In Section 5.2, we will show that this is feasible in practice, because the commercial ToR switch already can provide enough buffer and the buffer is shared by all the ports [26].

Once the queue length exceeds the threshold due to an ON period, which may be corresponding to the advent of an elephant flow, the ToR switch informs the controller of the detection of burst traffic, and the controller calculates a lightpath for this pair of ToR switches. After that, the controller reconfigures the OCS to set up a lightpath to offload this burst traffic. At the same time, the controller also sends the ToR switch a new flow entry, such that all the packets to this destination can be switched to this lightpath. The queue length begins to decline when the lightpath serves the traffic.

The controller will store a lightpath set-up request in a buffer, called request buffer, in case it fails to find an OCS available for this request when it receives the request from the ToR switch. Once there is a lightpath just torn down, the controller will find if there is a buffered request that can be satisfied by the released optical resource. If there is, the controller will set up a lightpath for this request. To timely offload the elephant flows when they are detected, the data center should install enough OCSs to ensure the probability that the request has to wait in the buffer is very small (e.g., 0.2 percent).

Once the buffer is cleaned up, the ToR switch tears down the lightpath and deletes the relevant flow entry from its flow table. In the meantime, the ToR switch transfers the traffic flows back to the EPS connection. We select such lightpath release opportunity to avoid frequent teardown and re-buildup of the lightpaths, which incur a large system overhead as the next paragraph mentions.

We consider two system parameters for the selection of the buffer threshold: mean delay and lightpath set-up frequency. The mean delay is the mean duration from the time when a packet enters the buffer to the time when it is transmitted, and the lightpath set-up frequency is defined as the number of lightpaths built up by each ToR pair per second. We use the lightpath set-up frequency to indicate the operation overhead of the system. The higher lightpath set-up frequency, the larger the system overhead [16], [18]. First, setting up or tearing down a lightpath requires the information exchange between the ToR switch and the controller. Second, routing calculation for a lightpath consumes the computing resource of the controller. Third, the reconfiguration time of OCSs during the lightpath establishment is another concern. As Section 1 mentions, the reconfiguration time of OCSs is about 1 μs if silicon MEMS-based optical switches [13], [14] are employed. Thus, lightpath establishment not only consumes the bandwidth and the computing resources, but also suffers from the delay including the communication time between the ToR switch and the controller, the routing calculation time, and the reconfiguration time of OCSs. Thus, we use the lightpath set-up frequency to delineate the system overhead. For a given traffic load, our goal is to find a proper threshold such that the mean delay is low while the lightpath set-up frequency is not high.

Intuitively, the buffer threshold selection is susceptive to traffic burstiness. It is well-known that, for a typical queuing system, the average queuing delay of the packets tends to increase with the burstiness of the input traffic [27], [28], [29], [30]. In addition, the traffic burstiness of different ToR pairs could be different. The method to select a proper threshold according to the traffic burstiness is important for performance optimization of each ToR switch pair. Thus, we will study the effect of traffic burstiness on the mean delay and the lightpath set-up frequency, based on which we give the threshold selection rule.

3 Modelling of Optical Traffic Offloading Process

As Section 2 mentions, the mean delay and the lightpath set-up frequency are two basic criteria to select a proper buffer threshold. Thus, we analyze the optical traffic offloading process and derive the mean delay and the lightpath set-up frequency in this section.

3.1 Fluid-Flow Model

According to the test data collected by Microsoft Research [1], the input traffic in data centers exhibits an ON-OFF pattern. We assume that the input traffic is a two-state ON-OFF fluid flow as shown in Fig. 3, where the ON state describes the elephant flow while the OFF state represents the case of no elephant flow. In this paper, we simply refer to the duration of a traffic ON-OFF state as a traffic ON-OFF period.

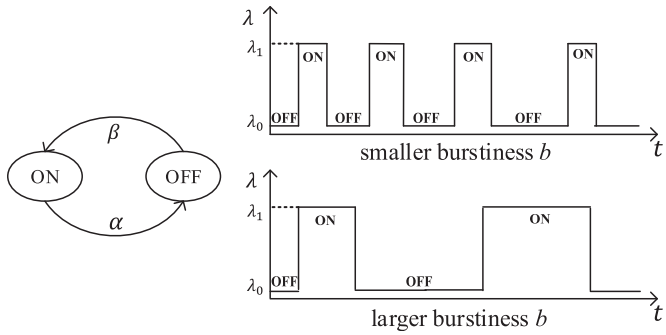


Fig. 3. ON-OFF traffic arrival process.

Suppose that the traffic-ON period is an exponentially distributed random variable with mean $1/\alpha$, and the traffic-OFF period is also an exponentially distributed random variable with mean $1/\beta$. The probabilities that the input traffic is in the ON state and the OFF state are given by $\pi_{\text{on}} = \beta/(\alpha+\beta)$ and $\pi_{\text{off}} = 1 - \pi_{\text{on}}$, respectively. Furthermore, the test data in [7] shows that, in a data center with 150 server racks, a ToR switch generates 2 concurrent elephant flows on average and the number of concurrent elephant flows produced by a ToR switch exceeds 5 almost with probability 0. According to this distribution, we set $\pi_{\text{on}} = 1/61$ and $\pi_{\text{off}} = 60/61$ with a ratio $\alpha : \beta = 60 : 1$ throughout this paper. Let λ_1 and λ_0 be the traffic rates of the traffic-ON period and the traffic-OFF period, respectively, where $\lambda_1 > \lambda_0$, and then the average input traffic rate is given by $\overline{\lambda} = \pi_{\text{off}}\lambda_0 + \pi_{\text{on}}\lambda_1$.

Define $b = 1/(\alpha+\beta)$ as the traffic burstiness, from the ON-OFF traffic model described in Fig. 3, we have

$$b = \frac{\pi_{\text{off}}}{\alpha} = \frac{\pi_{\text{on}}}{\beta},$$

which implies that the burstiness b is related to the traffic ON-OFF period once π_{on} or the ratio $\alpha : \beta$ is fixed. When the traffic ON-OFF period is long, the traffic injected to the network in each traffic-ON period is large, which implies a large b. According to the test data by Microsoft Research [1], the length of the traffic-OFF period is between 20 s and 70 s in most commercial data centers, which means the range of the burstiness b is typically between 0.328 and 1.148.

The implementation of optical traffic offloading is governed by the queue length. We assume that the buffer installed in each ToR switch is infinitely large. In our offloading strategy, when the queue length is lower than the threshold L, the ToR switch transmits traffic via EPSs with service rate μ_0. Here, we say the ToR switch is in the electrical service state. We have $\mu_0 = \overline{\lambda}$ according to the optical traffic offloading strategy. When the queue length exceeds L, the ToR switch transmits traffic via the OCS with service rate μ_1, where $\mu_1 > \lambda_1 > \mu_0$. Here, we say the ToR switch is in the optical service state.

As Section 2 shows, the lightpath set-up takes time. To simplify the analysis, we initially ignore the lightpath set-up delay in our analysis, and revisit it in Section 5.1 to complete the analysis. Also, we assume that there are enough OCSs in the network, such that the probability that the lightpath set-up request has to wait in the request buffer is negligible. For example, 6 OCSs are enough for the network with 80 server racks, as Section 5.2 shows.

Furthermore, to facilitate the presentation, we list the symbols employed in the paper as follows:

b	Traffic burstiness
S	Input traffic state
M	Service state
X	Queue length
D	Mean delay
f	Lightpath set-up frequency
$1/\alpha$	Mean traffic-ON period
$1/\beta$	Mean traffic-OFF period
λ_1	Input traffic rate during the traffic-ON period
λ_0	Input traffic rate during the traffic-OFF period
$\overline{\lambda}$	Average input traffic rate
π_{on}	Probability that the input traffic is in the ON state
π_{off}	Probability that the input traffic is in the OFF state
$P(x)$	Cumulative distribution function (CDF) of X in the equilibrium state
$p(x)$	Probability density function (PDF) of X in the equilibrium state
$P_{i,j}(x)$	CDF of X in the equilibrium state $(S=i, M=j)$
$p_{i,j}(x)$	PDF of X in the equilibrium state $(S=i, M=j)$

3.2 System State Equations

The state of a source ToR switch is defined by a two-tuple (S, M), where $S = 0, 1$ and $M = 0, 1$ denote the input traffic state and the service state, respectively. According to the optical traffic offloading process described in Section 2, a source ToR switch may experience the following states:

1. $(S = 0, M = 0)$: the input traffic is in the OFF state, and the service state is in the electrical service state;
2. $(S = 0, M = 1)$: the input traffic is in the OFF state, and the service state is in the optical service state;
3. $(S = 1, M = 0)$: the input traffic is in the ON state, and the service state is in the electrical service state;
4. $(S = 1, M = 1)$: the input traffic is in the ON state, and the service state is in the optical service state.

Fig. 4 plots the state transitions of a source ToR switch. There are two kinds of state transitions. The first kind of state transition is the transition independent of the queue length of the ToR switch, which happens when the input traffic turns ON or OFF. These kinds of transitions are denoted by the solid arrows in Fig. 4. The second kind of state transition is the transition between the state with $M = 0$ and that with $M = 1$, which are controlled by the queue length. These kinds of transitions are represented by the dashed arrows in Fig. 4. These two kinds of transitions do not occur at the same time, and thus the transitions between (0,0) and (1,1) and that between (1,0) and (0,1) do not happen. Let X be the queue length, the state transitions are described as follows:

- If the ToR switch is in state (0,0), only one kind of state transition happens. When the input traffic turns ON, state (0,0) transits to state (1,0). However, the transition from state (0,0) to state (0,1) does not happen, because $\lambda_0 < \mu_0$ in state (0,0), and thus the queue length X can never increase to the threshold L.
- If the ToR switch is in state (1,0), $\lambda_1 > \mu_0$ and the queue length continuously increases. When the queue length increases to the threshold L, the ToR switch requests the controller to establish a lightpath, and thus state (1,0) transits to state (1,1). Also, state (1,0) transits to state (0,0) if the input traffic switches OFF.
- If the ToR switch is in one of two states (0,1) and (1,1) where $M = 1$, the backlog is served by a lightpath with service rate $\mu_1 > \lambda_1 > \lambda_0$, and the queue length decreases. When the buffer becomes empty, the controller tears down the lightpath immediately, which triggers the state transition from (0,1) to (0,0) or from (1,1) to (1,0). In addition, state (0,1) transits to state (1,1) when the input traffic turns ON, and state (1,1) transits to state (0,1) when the input traffic turns OFF.

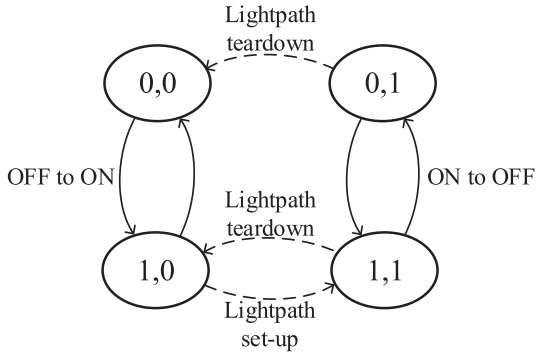


Fig. 4. State transition of each ToR switch.

Define the cumulative distribution function (CDF) of the queue length $X(t)$ at time t as

$$P_{i,j}(x,t) \triangleq Pr\{S(t)=i, M(t)=j, X(t)\le x\},$$

and its probability density function (PDF) as $p_{i,j}(x,t)$. According to the state transitions in Fig. 4, as Appendix A shows, we can establish the following partial differential equations:

$$\begin{aligned}
\frac{\partial P_{0,0}(x,t)}{\partial t} =& -(\lambda_0-\mu_0)\frac{\partial P_{0,0}(x,t)}{\partial x} - \beta P_{0,0}(x,t) \\
&+ \alpha P_{1,0}(x,t) - (\lambda_0-\mu_1)p_{0,1}(0_+,t), \\
\frac{\partial P_{1,0}(x,t)}{\partial t} =& -(\lambda_1-\mu_0)\frac{\partial P_{1,0}(x,t)}{\partial x} + \beta P_{0,0}(x,t) \\
&- \alpha P_{1,0}(x,t) - (\lambda_1-\mu_1)p_{1,1}(0_+,t), \\
\frac{\partial P_{0,1}(x,t)}{\partial t} =& -(\lambda_0-\mu_1)\frac{\partial P_{0,1}(x,t)}{\partial x} - \beta P_{0,1}(x,t) \\
&+ \alpha P_{1,1}(x,t) - (\lambda_0-\mu_1)p_{0,1}(0_+,t), \\
\frac{\partial P_{1,1}(x,t)}{\partial t} =& -(\lambda_1-\mu_1)\frac{\partial P_{1,1}(x,t)}{\partial x} + \beta P_{0,1}(x,t) \\
&- \alpha P_{1,1}(x,t) - (\lambda_1-\mu_1)p_{1,1}(0_+,t).
\end{aligned} \tag{1}$$

When the system reaches the equilibrium state as $t \to \infty$, CDF $P_{i,j}(x,t)$ approaches to the following equilibrium CDF as $\frac{\partial P_{i,j}(x,t)}{\partial t} \to 0$ in (1):

$$P_{i,j}(x) = \lim_{t\to\infty} P_{i,j}(x,t).$$

We can express the differential equations of $P_{i,j}(x)$ in the matrix form given by (2).

Solving the differential Equations in (2), we derive the solution of $P_{i,j}(x)$ in (3),where

$$\begin{pmatrix} \frac{\mathrm{d}P_{0,0}(x)}{\mathrm{d}x} \\ \frac{\mathrm{d}P_{1,0}(x)}{\mathrm{d}x} \\ \frac{\mathrm{d}P_{0,1}(x)}{\mathrm{d}x} \\ \frac{\mathrm{d}P_{1,1}(x)}{\mathrm{d}x} \end{pmatrix} = \begin{pmatrix} \frac{-\beta}{\lambda_0-\mu_0} & \frac{\alpha}{\lambda_0-\mu_0} & 0 & 0 \\ \frac{\beta}{\lambda_1-\mu_0} & \frac{-\alpha}{\lambda_1-\mu_0} & 0 & 0 \\ 0 & 0 & \frac{-\beta}{\lambda_0-\mu_1} & \frac{\alpha}{\lambda_0-\mu_1} \\ 0 & 0 & \frac{\beta}{\lambda_1-\mu_1} & \frac{-\alpha}{\lambda_1-\mu_1} \end{pmatrix} \times \begin{pmatrix} P_{0,0}(x) \\ P_{1,0}(x) \\ P_{0,1}(x) \\ P_{1,1}(x) \end{pmatrix} + \begin{pmatrix} \frac{-(\lambda_0-\mu_1)p_{0,1}(0_+)}{\lambda_0-\mu_0} \\ \frac{-(\lambda_1-\mu_1)p_{1,1}(0_+)}{\lambda_1-\mu_0} \\ p_{0,1}(0_+) \\ p_{1,1}(0_+) \end{pmatrix} \tag{2}$$

$$\begin{pmatrix} P_{0,0}(x) \\ P_{1,0}(x) \\ P_{0,1}(x) \\ P_{1,1}(x) \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ (\lambda_1-\mu_1)c_2e^{mx} \\ (\mu_1-\lambda_0)c_2e^{mx} \end{pmatrix} + \begin{pmatrix} \frac{(\alpha+\beta)^2(-A-B)}{2\beta(\lambda_0-\lambda_1)^2} \\ \frac{(\alpha+\beta)^2(-A-B)}{2\alpha(\lambda_0-\lambda_1)^2} \\ 0 \\ 0 \end{pmatrix} x^2 + \begin{pmatrix} d_1 \\ \frac{-(\alpha+\beta)(A+B)}{\alpha(\lambda_0-\lambda_1)} + \frac{\beta}{\lambda}d_1 \\ \frac{-\alpha(A+B)}{\beta(\lambda_1-\mu_1)+\alpha(\lambda_0-\mu_1)} \\ \frac{-\beta(A+B)}{\beta(\lambda_1-\mu_1)+\alpha(\lambda_0-\mu_1)} \end{pmatrix} x + \begin{pmatrix} \alpha c_0 + \frac{-[(\lambda_0-\lambda_1)d_1-A])}{\alpha+\beta} \\ \beta c_0 + \frac{-A}{\alpha+\beta} \\ \alpha c_1 + \frac{-A(\lambda_1-\mu_1)}{\beta(\lambda_1-\mu_1)+\alpha(\lambda_0-\mu_1)} \\ \beta c_1 + \frac{-B(\lambda_0-\mu_1)}{\beta(\lambda_1-\mu_1)+\alpha(\lambda_0-\mu_1)} \end{pmatrix}. \tag{3}$$

$$m = \frac{-\beta(\lambda_1-\mu_1)-\alpha(\lambda_0-\mu_1)}{(\lambda_0-\mu_1)(\lambda_1-\mu_1)},$$

and c_0, c_1, c_2, d_1, A, B are undetermined coefficients. These undetermined coefficients can be determined from the following boundary conditions:

B1. $P_{1,0}(0) = 0$.

This boundary condition can be understood from the state transitions. As Fig. 4 shows, state (1,0) can be visited by states (1,0), (0,0), and (1,1). If the ToR switch stays in (1,0), the queue length does not stop at $X = 0$, since $\mu_0 < \lambda_1$ and thus the queue length keeps increasing. If the ToR switch initially stays in (0,0) long enough, its backlog is cleaned up since $\mu_0 > \lambda_0$. In this case, once the input traffic turns ON, the ToR switch transits from (0,0) to (1,0), and the queue length increases and becomes larger than 0 since $\mu_0 < \lambda_1$. If the ToR switch initially stays in (1,1) long enough, the queue length eventually decreases to 0, and the ToR switch transits from (1,1) to (1,0). In this case, the queue length also increases and becomes larger than 0 since $\mu_0 < \lambda_1$. In summary, though the queue length may experience $X = 0$, it will never stop at $X = 0$ when the ToR switch is in state (1,0), i.e., the probability that $X \leq 0$ when the ToR switch is in state (1,0) is

$$P_{1,0}(0) = Pr\{S = 1, M = 0, X \leq 0\} = 0.$$

B2. $P_{0,1}(0) = 0$ and $P_{1,1}(0) = 0$.

If the ToR switch is in the states with $M = 1$, the service is provided by a lightpath and the optical service state remains unchanged until the queue length decreases to 0. In this case, no matter whether the input traffic is ON or OFF, the queue length will eventually reduce to 0, since $\mu_1 > \lambda_1 > \lambda_0$. When the buffer becomes empty, the service state immediately changes from the optical service state to the electrical service state, i.e., $M = 0$. This implies that the queue length may experience $X = 0$, but it does not stop at $X = 0$, when the ToR switch is in the optical service state. It follows that

$$P_{0,1}(0) = Pr\{S = 0, M = 1, X \leq 0\} = 0,$$

and

$$P_{1,1}(0) = Pr\{S = 1, M = 1, X \leq 0\} = 0.$$

Note that the queue length can stay at $X = 0$ since $\mu_0 > \lambda_0$ when the ToR switch is in state (0,0), which means

$$P_{0,0}(0) = Pr\{S = 0, M = 0, X \leq 0\} > 0.$$

B3. $p_{0,0}(L) = \frac{\mathrm{d}P_{0,0}(x)}{\mathrm{d}x}|_{x=L} = 0$ and $p_{0,1}(L) = \frac{\mathrm{d}P_{0,1}(x)}{\mathrm{d}x}|_{x=L} = 0$.

According to Fig. 4, state (0,0) can be accessed by states (0,0), (0,1), and (1,0). If the ToR switch remains in (0,0), the queue length can never reach L since $\lambda_0 < \mu_0$. If the ToR switch initially stays in (0,1) long enough, the buffer will be emptied by the lightpath, and the ToR switch will transit from (0,1) to (0,0). After the state transition, the queue length stays at $X = 0$ until the input traffic turns ON, since $\mu_0 > \lambda_0$. Suppose the ToR switch initially stays in (1,0). In this case, the queue length increases since $\mu_0 < \lambda_1$. If the queue length does not reach the threshold L while the input traffic turns OFF, the ToR switch transits from (1,0) to (0,0). Even if the state transition happens when the queue length is very close to L, the queue length declines immediately and cannot reach L because $\lambda_0 < \mu_0$. In a word, it is because $\lambda_0 < \mu_0$ that the queue length can never reach L when the ToR switch is in state (0,0). By definition, PDF $p_{i,j}(x)$ is the relative likelihood that the queue length would equal x. Thus, we have the relative likelihood that $X = L$ as follows:

$$p_{0,0}(L) = \frac{\mathrm{d}P_{0,0}(x)}{\mathrm{d}x}|_{x=L} = 0.$$

As Fig. 4 plots, the ToR switch enters (0,1), only when it is in (1,1) and the input traffic turns OFF. Recall that the service state of (1,1) is the optical service state, and thus the queue length continuously decreases and does not stay at $X = L$. It follows that the queue length must be less than L, when the transition from (1,1) to (0,1) happens. In other words, when the ToR switch is in state (0,1), the queue length can never equal L. Thus, we have

$$p_{0,1}(L) = \frac{\mathrm{d}P_{0,1}(x)}{\mathrm{d}x}|_{x=L} = 0.$$

B4. $P_{1,0}(L) + P_{1,1}(L) = \pi_{\mathrm{on}} = \frac{\beta}{\alpha+\beta}$.

Since the queue length ranges from 0 to L, probability π_{on} that the input traffic is ON is equal to the sum

of $P_{1,0}(L)$ and $P_{1,1}(L)$, yielding the following boundary condition:

$$P_{1,0}(L) + P_{1,1}(L) = \pi_{\text{on}} = \frac{\beta}{\alpha+\beta}.$$

Using boundary conditions B1 through B4, we can determine all undetermined coefficients in (3) and thus $P_{i,j}(x)$ in (3). Finally, we can derive the CDF of X as follows:

$$\begin{aligned} P(x) &= Pr(X \le x) \\ &= P_{0,0}(x) + P_{1,0}(x) + P_{0,1}(x) + P_{1,1}(x). \end{aligned} \tag{4}$$

3.3 Mean Delay

The mean delay, denoted as D, is an important criterion for the buffer threshold selection that guarantees the input traffic will not suffer from a large delay. The mean delay D can be easily derived from the CDF of the queue length.

Since the mean queue length Q is the expectation of the queue length X, we have

$$Q = \int_0^L xp(x)dx. \tag{5}$$

According to Little's Law, we immediately obtain the following mean delay:

$$\begin{aligned} D &= \frac{Q}{\overline{\lambda}} \\ &= \frac{\pi_{\text{on}}\left\{\frac{K_0L^2}{2} + \frac{K_1L^3}{b} + K_2b^2\left(\frac{\Omega_1}{b}L - 1 + e^{-\frac{\Omega_1}{b}L}\right)\right\}}{\overline{\lambda}\left\{K_3L + \frac{K_4L^2}{b} + K_5b\left(1 - e^{-\frac{\Omega_1}{b}L}\right)\right\}}, \end{aligned} \tag{6}$$

where

$$\Omega_1 = \frac{\mu_1 - \overline{\lambda}}{(\lambda_0 - \mu_1)(\lambda_1 - \mu_1)},$$

$$K_0 = \frac{1}{\pi_{\text{off}}(\lambda_1 - \lambda_0)} + \frac{1}{\mu_1 - \overline{\lambda}},$$

$$K_1 = \frac{1}{6\pi_{\text{off}}\pi_{\text{on}}(\lambda_0 - \lambda_1)^2},$$

$$K_2 = \frac{\pi_{\text{off}}(\lambda_1 - \lambda_0)(\lambda_0 - \mu_1)^2(\lambda_1 - \mu_1)}{(\overline{\lambda} - \mu_1)^3},$$

$$K_3 = \frac{1}{\pi_{\text{off}}(\lambda_1 - \lambda_0)} + \frac{\pi_{\text{on}}}{\mu_1 - \overline{\lambda}},$$

$$K_4 = \frac{1}{2\pi_{\text{off}}(\lambda_0 - \lambda_1)^2},$$

and

$$K_5 = \frac{\pi_{\text{off}}(\lambda_0 - \mu_1)^2}{(\overline{\lambda} - \mu_1)^2}$$

are all constants independent of b and L.

3.4 Lightpath Set-Up Frequency

The lightpath set-up frequency, denoted as f, is another important criterion for the buffer threshold selection. The threshold should be selected to keep the lightpath set-up frequency low, such that the operation overhead imposed on the data center network is not high.

Consider a very long period $[0, T]$, during which the ToR switch requests to set up lightpaths n times. Let P_O be the probability that the ToR switch uses the lightpath to transmit the traffic, and H be the average holding time of the lightpath. We have

$$TP_O = nH. \tag{7}$$

It follows that the lightpath set-up frequency f is given by

$$f = \frac{n}{T} = \frac{P_O}{H}. \tag{8}$$

Since the input traffic could be ON or OFF when the ToR switch transmits the traffic via lightpaths, P_O can be obtained as follows:

$$P_O = P_{0,1}(L) + P_{1,1}(L). \tag{9}$$

Therefore, to derive the lightpath set-up frequency f, we need to find the average holding time of the lightpath H.

To facilitate the calculation of H, we define two following variables:

1. $T_{\text{on}}(x)$: the average residual time to empty the buffer by a lightpath when the queue length is $X = x$ and the input traffic is in the ON state;
2. $T_{\text{off}}(x)$: the average residual time to empty the buffer by a lightpath when the queue length is $X = x$ and the input traffic is in the OFF state.

As we mention in Section 3.2, the lightpath can be set up only when the input traffic is in the ON state. In this case, the ToR switch uses the electrical network, and the queue length increases to L. This implies that the input traffic must be in the ON state and the queue length is L at the time when the lightpath is just established. Thus, we have

$$H = T_{\text{on}}(L). \tag{10}$$

Let z be the first traffic-ON period that the input traffic experiences after lightpath establishment, as Fig. 5 illustrates. It is clear that z is an exponentially distributed random variable with parameter α. If the buffer is emptied before the input traffic switches OFF, i.e., z is long enough such that $(\mu_1 - \lambda_1)z \ge L$, the holding time of the lightpath will be $\frac{L}{\mu_1 - \lambda_1}$. If $(\mu_1 - \lambda_1)z < L$, the backlogged traffic that remains in the buffer at the time when the input traffic turns OFF will be $L - (\mu_1 - \lambda_1)z$. In this case, the average holding time of the lightpath is $z + T_{\text{off}}[L - (\mu_1 - \lambda_1)z]$. It follows that

$$\begin{aligned} H =& T_{\text{on}}(L) \\ =& \int_{\frac{L}{\mu_1 - \lambda_1}}^{+\infty} \frac{L}{\mu_1 - \lambda_1}\alpha e^{-\alpha z}dz \\ &+ \int_0^{\frac{L}{\mu_1 - \lambda_1}} \{z + T_{\text{off}}[L - (\mu_1 - \lambda_1)z]\}\alpha e^{-\alpha z}dz. \end{aligned} \tag{11}$$

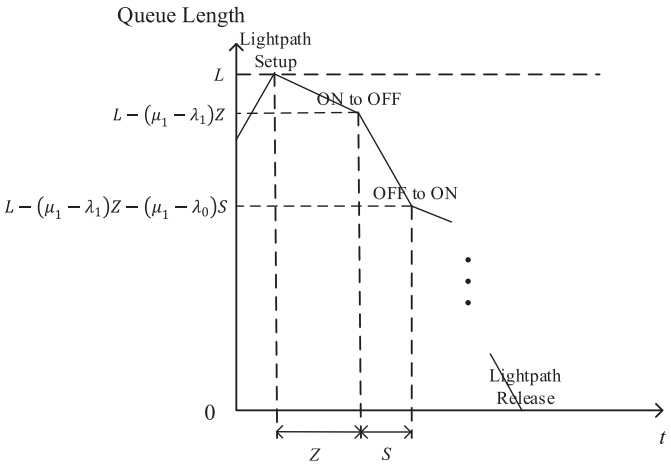


Fig. 5. Queue length evolution in optical service state.

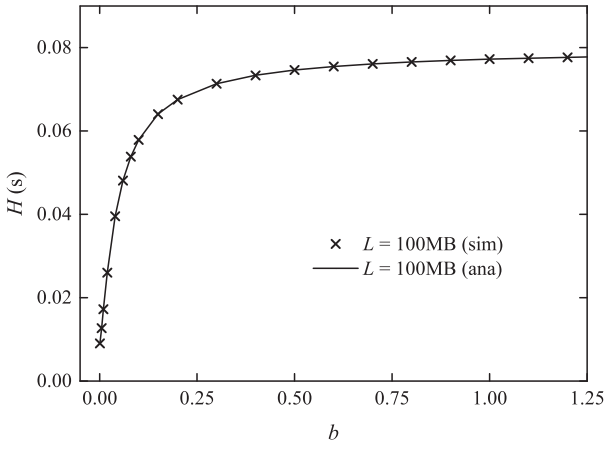


Fig. 6. Average holding time of a lightpath H changing with the burstiness b.

Let s be the first traffic-OFF period that the input traffic experiences after lightpath establishment, as Fig. 5 plots. It is clear that s is an exponentially distributed random variable with parameter β. If the buffer is emptied before the input traffic switches ON, i.e., s is long enough such that $(\mu_1-\lambda_1)z+(\mu_1-\lambda_0)s \geq L$, the holding time of the lightpath will be $\frac{L-(\mu_1-\lambda_1)z}{\mu_1-\lambda_0}$. Otherwise, the backlogged traffic that remains in the buffer at the time when the input traffic turns ON will be $L-(\mu_1-\lambda_1)z-(\mu_1-\lambda_0)s$, and the average residual time to empty the buffer is $s+T_{\text{on}}[L-(\mu_1-\lambda_1)z-(\mu_1-\lambda_0)s]$. Thus, we have the equation given by (12). Fig. 5 illustrates a possible queue length evolution during the lifetime of a lightpath. From the above derivation, we can find that H should be calculated in an iterative manner, which is very difficult.

$$\pi_{\text{off}}[L-(\mu_1-\lambda_1)z] = \int_{\frac{L-(\mu_1-\lambda_1)z}{\mu_1-\lambda_0}}^{+\infty} \frac{L-(\mu_1-\lambda_0)z}{\mu_1-\lambda_0}\beta e^{-\beta s}ds + \int_0^{\frac{L-(\mu_1-\lambda_1)z}{\mu_1-\lambda_0}} \{s+T_{\text{on}}[L-(\mu_1-\lambda_1)z-(\mu_1-\lambda_0)s]\}\beta e^{-\beta s}ds \tag{12}$$

Instead, we pursue an approximate solution for H. We can see from (11) that the iterative process is induced by the term $T_{\text{off}}[L-(\mu_1-\lambda_1)z]$ on the right side, which has to be calculated from (12). Thus, we solve (12) approximately using the following two arguments:

(a) When b is small, the input traffic switches ON and OFF back and forth rapidly, and thus the input traffic can be approximately considered as a constant flow with rate $\overline{\lambda}$. In this case, the traffic that remains in the buffer after the first traffic-ON period can be emptied at the rate of $\mu_1-\overline{\lambda}$, and thus (12) can be rewritten as:

$$T_{\text{off}}[L-(\mu_1-\lambda_1)z] \approx \frac{L-(\mu_1-\lambda_1)z}{\mu_1-\overline{\lambda}}. \tag{13}$$

(b) When b is large, the traffic-ON period is large, implying that the probability of $(\mu_1-\lambda_1)z < L$ is small and thus the second term in (11) is only a negligible portion of H. In this case, replacing $T_{\text{off}}[L-(\mu_1-\lambda_1)z]$ in (11) with $\frac{L-(\mu_1-\lambda_1)z}{\mu_1-\overline{\lambda}}$ will only bring a little error.

From (a) and (b), we can see that (13) is a good approximation of T_{off} for both small b and large b, which hints that (13) may be a good approximation of T_{off} for all bs. Following this idea, we adopt the following approximation of H for all bs:

$$\begin{aligned} H &\approx \int_{\frac{L}{\mu_1-\lambda_1}}^{+\infty} \frac{L}{\mu_1-\lambda_1}\alpha e^{-\alpha z}dz \\ &\quad + \int_0^{\frac{L}{\mu_1-\lambda_1}} \left(z+\frac{L-(\mu_1-\lambda_1)z}{\mu_1-\overline{\lambda}}\right)\alpha e^{-\alpha z}dz \\ &= \frac{L}{\mu_1-\overline{\lambda}} + \frac{\lambda_1-\overline{\lambda}}{\alpha(\mu_1-\overline{\lambda})}\left(1-e^{-\frac{\alpha L}{\mu_1-\lambda_1}}\right). \end{aligned} \tag{14}$$

To check the accuracy of (14), we plot H changing with b in Fig. 6, where the traffic rate in the ON state is $\lambda_1 = 90$ Gb/s and in the OFF state is $\lambda_0 = 0.17$ Gb/s; the electrical service rate is $\mu_0 = 1.64$ Gb/s; the optical service rate is $\mu_1 = 100$ Gb/s; and the threshold is $L = 100$ MB. Fig. 6 confirms that the approximation in (14) is very good.

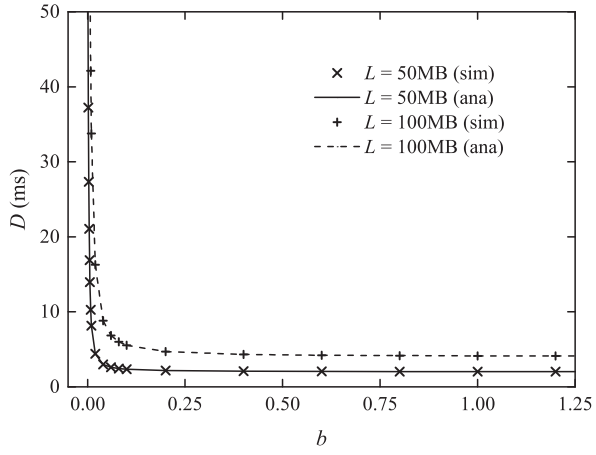
From (8), (10), and (14), we can find the approximate solution of f in (15), where K_6 is a constant as follows:

$$f \approx \frac{\pi_{\text{on}}\left\{K_6 b\left(1-e^{-\frac{\Omega_1}{b}L}\right)+L\right\}}{\left\{K_3 L+\frac{K_4 L^2}{b}+K_5 b\left(1-e^{-\frac{\Omega_1}{b}L}\right)\right\}\left[L+\frac{(\lambda_1-\overline{\lambda})b}{\pi_{\text{off}}}\left(1-e^{-\frac{\pi_{\text{off}}L}{(\mu_1-\lambda_1)b}}\right)\right]} \tag{15}$$

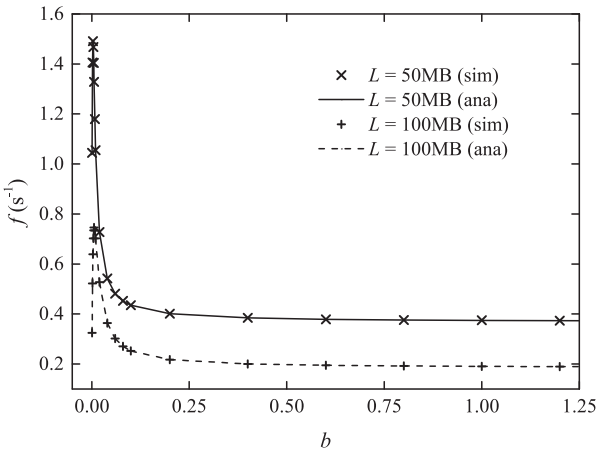
$$K_6 = \frac{-\pi_{\text{off}}(\lambda_1-\lambda_0)(\lambda_0-\mu_1)}{\mu_1-\overline{\lambda_1}}. \tag{16}$$

4 Buffer Threshold Selection

As Section 2 mentions, the traffic burstiness b may have a significant impact on the performance of data center networks, and the aim of our optical traffic offloading strategy is to cope with the burst traffic. In this section, we study the impact of the traffic burstiness b on the system performance in Section 4.1, based on which we provide the selection rule of the buffer threshold in Section 4.2.



(a) D vs. b



(b) f vs. b

Fig. 7. System performance versus burstiness b.

4.1 Effect of b on System Performance

In this section, we numerically study the effect of the burstiness b on the system performance, using the results derived in Section 3. In our numerical study, the traffic rate in the ON state is $\lambda_1 = 90$ Gb/s and that in the OFF state is $\lambda_0 = 0.17$ Gb/s, and $\alpha : \beta = 60 : 1$. Thus, the average input traffic rate is $\overline{\lambda} = \pi_{\text{off}}\lambda_0 + \pi_{\text{on}}\lambda_1 = 1.64$ Gb/s. The electrical service rate is $\mu_0 = \overline{\lambda} = 1.64$ Gb/s and the optical service rate is $\mu_1 = 100$ Gb/s. In addition, we consider two cases, where the threshold $L = 50$ MB and $L = 100$ MB.

We plot the mean delay D and the lightpath set-up frequency f changing with b in Fig. 7, where the analytical results are obtained by (6) and (14). Fig. 7 shows that our analytical results agree with the simulation results very well. As Fig. 7a shows, the mean delay D is very large when b is slightly larger than 0, and decreases very fast with the increase of b. Finally, D converges to a constant when $b > 0.32$. For example, D finally approaches to 2.02 ms when $L = 50$ MB, and 4.06 ms when $L = 100$ MB. On the other hand, as Fig. 7b shows, the lightpath set-up frequency f first sharply climbs up from a small value and then drops down rapidly with the increase of b, and finally approaches a constant when $b > 0.32$. It is very interesting to see from Fig. 7 that, when b is large enough (for example $b > 0.32$), both D and f converge to constants and almost do not change with b any more. In other words, for a selected buffer threshold, the burstiness b has only a little impact on system performance, when b is sufficiently large.

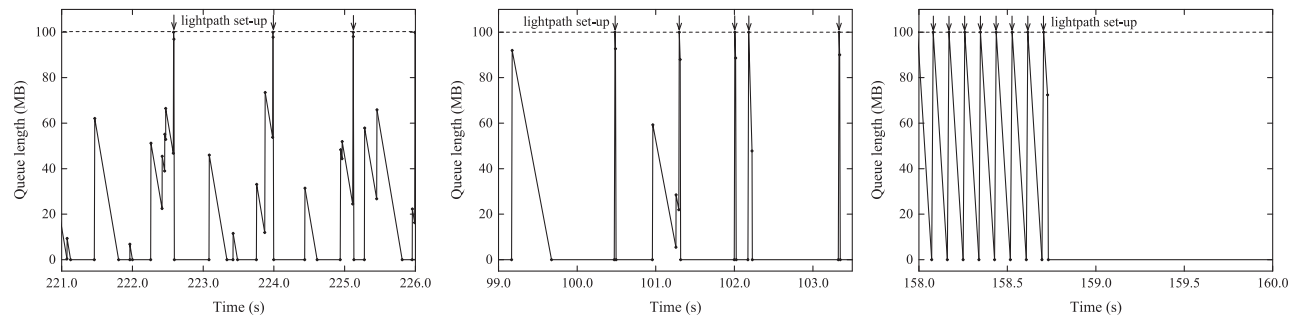
To better understand the results in Fig. 7, we inspect the evolution process of the queue length under different bs in Fig. 8, where we set the threshold $L = 100$ MB as an example.

When b is very small, say $b = 0.004$ in Fig. 8a, both the traffic-ON period and the traffic-OFF period are very small. In this case, if the service state is in the electrical service state, the queue length changes back and forth, and thus needs a long time to reach the threshold to trigger lightpath establishment, as Fig. 8a shows. The backlog cannot be emptied in time in this case, which could result in a large mean delay D and a small lightpath set-up frequency f. In the case presented in Fig. 8a, $D = 74.46$ ms and $f = 0.7022\text{s}^{-1}$.

When b increases to 0.02, both the traffic-ON period and the traffic-OFF period increase, making it easier for the queue length to reach the threshold L to set up lightpaths, as Fig. 8b plots. In this case, the mean delay D drops down to 16.27 ms since the buffer can be emptied quickly, and the lightpath set-up frequency f increases to 0.5278s^{-1}.

When b is large enough, for example $b = 1$, as shown in Fig. 8c, the traffic-ON period has a sufficiently long duration, in which the queue length fluctuation looks like a triangle wave because:

1) The queue length increases linearly with time when the ToR switch serves the traffic via the electrical network;
2) The queue length declines linearly with time to 0 once the lightpath is established.



(a) $b = 0.004$ (b) $b = 0.02$ (c) $b = 1$

Fig. 8. Queue length evolution under different traffic burstiness bs.

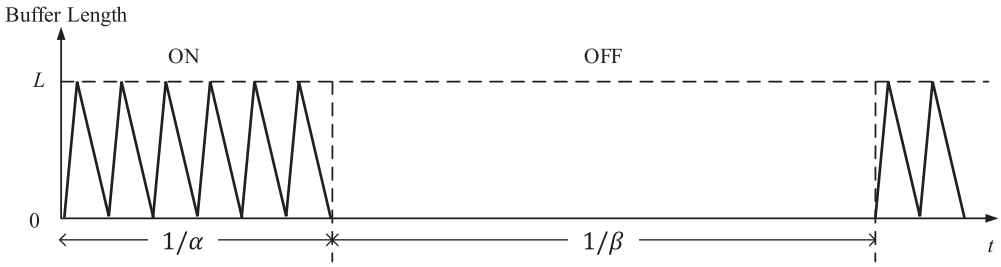


Fig. 9. Queue length evolution when b is sufficiently large.

In particular, it first takes $\frac{L}{\lambda_1-\mu_0}$ seconds for the queue length to increase from 0 to L, and it then takes $\frac{L}{\mu_1-\lambda_1}$ seconds for the lightpath to empty the buffer. This process continuously repeats until the traffic-ON period finishes. On the other hand, the input traffic spends a long time in each traffic-OFF period, during which the queue length is almost zero since the input traffic rate is now less than both the optical service rate and the electrical service rate. In this case, the mean delay and the lightpath set-up frequency converge to 4.13 ms and 0.1907s^{-1}, respectively, and are almost invariant with respect to b, which is interpreted as follows.

Let's consider an ideal case in Fig. 9, where each traffic-ON period and each traffic-OFF period is a constant and sufficiently large. In this case, the mean queue length is approximately equal to the area of the triangles during the traffic-ON period divided by $1/\alpha + 1/\beta$, and the lightpath set-up frequency is approximately equal to the number of the triangles divided by $1/\alpha + 1/\beta$. Since $1/\alpha$ and $1/\beta$ are sufficiently large, we have the number of the triangles in an ON period

$$n = \left\lfloor \frac{\frac{1}{\alpha}}{\frac{L}{\lambda_1-\mu_0}+\frac{L}{\mu_1-\lambda_1}} \right\rfloor \to \frac{\frac{1}{\alpha}}{\frac{L}{\lambda_1-\mu_0}+\frac{L}{\mu_1-\lambda_1}},$$

and the mean queue length approaches to 0 during the OFF period. The area of a triangle is $\frac{L}{2}(\frac{L}{\lambda_1-\mu_0}+\frac{L}{u_1-\lambda_1})$. It follows that, the mean queue length, denoted by Q, is equal to

$$Q = \frac{\frac{\frac{1}{\alpha}}{\frac{L}{\lambda_1-\mu_0}+\frac{L}{\mu_1-\lambda_1}} \times \frac{L}{2}\left(\frac{L}{\lambda_1-\mu_0}+\frac{L}{\mu_1-\lambda_1}\right)}{\frac{1}{\alpha}+\frac{1}{\beta}} = \frac{\frac{1}{\alpha}}{\frac{1}{\alpha}+\frac{1}{\beta}} \times \frac{L}{2} = \pi_{\text{on}}\frac{L}{2},$$

and thus the mean delay can be calculated as follows

$$D^* = \frac{Q}{\overline{\lambda}} = \frac{\pi_{\text{on}}}{\overline{\lambda}}\frac{L}{2}, \tag{17}$$

and the lightpath set-up frequency is given by

$$f^* = \frac{\frac{\frac{1}{\alpha}}{\frac{L}{\lambda_1-\mu_0}+\frac{L}{\mu_1-\lambda_1}}}{\frac{1}{\alpha}+\frac{1}{\beta}} = \frac{\pi_{\text{on}}(\lambda_1-\mu_0)(\mu_1-\lambda_1)}{(\mu_1-\mu_0)L}. \tag{18}$$

Equations (17) and (18) clearly show that when the ON-OFF period and thus the burstiness b are sufficiently large, the mean delay D^* and the lightpath set-up frequency f^* are independent of b.

The following theorem shows that this observation is also valid in a general case.

Theorem 1. *When the traffic burstiness b is sufficiently large, the mean delay D and lightpath set-up frequency f are given by:*

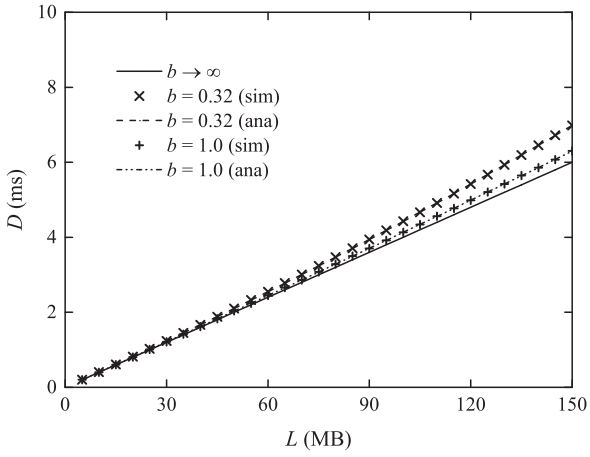
$$D \to D^* = \frac{\pi_{\text{on}}}{\overline{\lambda}}\frac{L}{2}, \tag{19}$$

and

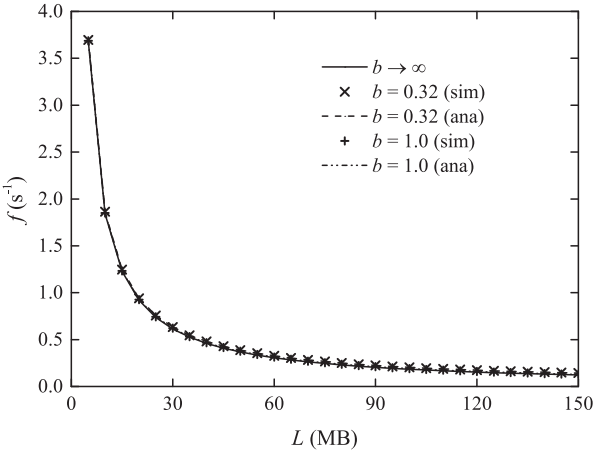
$$f \to f^* = \frac{\pi_{\text{on}}(\lambda_1-\mu_0)(\mu_1-\lambda_1)}{(\mu_1-\mu_0)L}. \tag{20}$$

Proof. It is easy to obtain (19) and (20) by applying L'Hospital's rule to (6) and (15) when $b \to \infty$. □

Fig. 10 compares the system performance under $b = 0.32$, $b = 1$ and $b \to \infty$, using the same parameters as that in Figs. 7 and 8. Fig. 10 clearly shows that the result of $b = 0.32$ is already very close to that of $b \to \infty$, especially when the



(a) D vs. L



(b) f vs. L

Fig. 10. System performance under $b = 0.32$, $b = 1$, and $b \to \infty$.

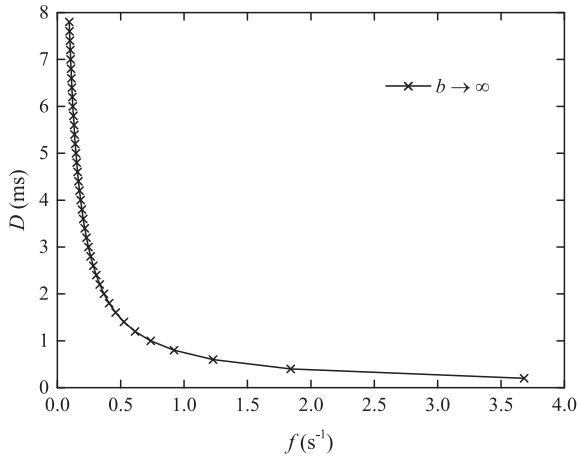


Fig. 11. Trade-off between D and f.

threshold L is not very large. On the other hand, Ref. [1] shows that the traffic-OFF period in commercial data center networks ranges from 20 s to 70 s, which implies that the traffic burstiness in a typical data center ranges from 0.328 to 1.148 according to our definition. On the other hand, Fig. 10 and the test results in [1] indicate that the precondition in Theorem 1 can be easily satisfied in practice, and thus Theorem 1 can apply to commercial data center networks and form the basis of the buffer threshold selection, as we show in Section 4.2.

4.2 Trade-off Between D and f

Intuitively, there is a trade-off between the mean delay D and the lightpath set-up frequency f when we set the threshold L. If L is small, the lightpaths are established frequently, and D is small since the backlog is emptied quickly by the lightpaths. If L is large, the lightpaths are rarely set up, and D is large since the traffic has to wait a long time in the buffer for the lightpath to clean it up. Such a trade-off between the mean delay and the lightpath set-up frequency is delineated by Theorem 1, from which we can see D^* is inversely proportional to f^* as follows:

$$D^* = \frac{\pi_{\text{on}}^2(\lambda_1 - \mu_0)(\mu_1 - \lambda_1)}{2\overline{\lambda}(\mu_1 - \mu_0)f^*}. \tag{21}$$

We visualize (21) in Fig. 11, from which we can see that increasing f slightly from 0 to 0.5s^{-1} can remarkably decrease D, but further increasing f not only has little contribution to the reduction of D but also remarkably imposes a large operation overhead to the system, as we mention in Section 1.

Consider the most important thing in data center networks is to guarantee the delay performance [7]. According to Theorem 1, we suggest a threshold selection rule as follows.

Rule 1. *For a given mean delay requirement $\widehat{D}$, the threshold size L should be:*

$$L = \frac{2\overline{\lambda}\widehat{D}}{\pi_{\text{on}}}. \tag{22}$$

5 Applications

In reality, it takes time to set up a lightpath, as we show in Section 1. We define lightpath set-up delay, denoted by τ, as the time interval from the time when the queue length reaches the threshold L to the time when the lightpath is built up. The lightpath set-up delay τ mainly consists of the following parts:

1) Communication delay between the ToR switch and the controller, which is about 10 microseconds [18];
2) Time for the controller to calculate a lightpath, which can be on the order of nanosecond if the ultra-fast routing algorithm in [31] is used;
3) Reconfiguration time of the OCS, which can be as small as 1 microsecond [13], [14].

Therefore, τ is on the order of tens of microseconds. Because of the existence of τ, the ToR switch cannot switch to the optical service state immediately, when the queue length reaches the threshold L. During the lightpath set-up process, the traffic is still served by the electrical network, and the queue length keeps changing, which may influence the mean delay D and the lightpath set-up frequency f. Thus, we take the lightpath set-up delay τ into consideration and amend Theorem 1 in Section 5.1. Section 5.2 provides an example that shows how the selection rule can be used in practice.

5.1 Threshold Selection With Lightpath Set-up Delay

Though the queue length fluctuates during the lightpath set-up process, the effect of such fluctuations on the mean delay D and the ligthpath set-up frequency f is small if the traffic burstiness b is sufficiently large. Let's consider the case where $b = 1$ as an example. The average traffic-ON period is 1.017 s, which is several orders of magnitude larger than the lightpath set-up delay τ. During the traffic-ON period, the queue length repeatedly experiences the following process:

(a) The queue length increases linearly with time from 0 to the threshold L when the ToR switch serves the traffic through the electrical network;
(b) The ToR switch still sends the traffic via the electrical network, and thus the queue length climbs up linearly with time from L to $L + (\lambda_1 - \mu_0)\tau$ when the lightpath is under establishment;
(c) The queue length declines linearly with time from $L + (\lambda_1 - \mu_0)\tau$ to 0 after the lightpath is established.

Thus, the queue length fluctuation in this case looks like a triangle wave also, as Fig. 12 plots.

Using this argument, we have the following result.

Corollary 1. *If the lightpath set-up delay $\tau > 0$, the mean delay D_τ^* and the lightpath set-up frequency f_τ^* are given as follows:*

$$D_\tau^* \approx \frac{\pi_{\text{on}}[L + (\lambda_1 - \mu_0)\tau]}{2\overline{\lambda}}, \tag{23}$$

and

$$f_\tau^* \approx \frac{\pi_{\text{on}}(\lambda_1 - \mu_0)(\mu_1 - \lambda_1)}{(\mu_1 - \mu_0)[L + (\lambda_1 - \mu_0)\tau]}. \tag{24}$$

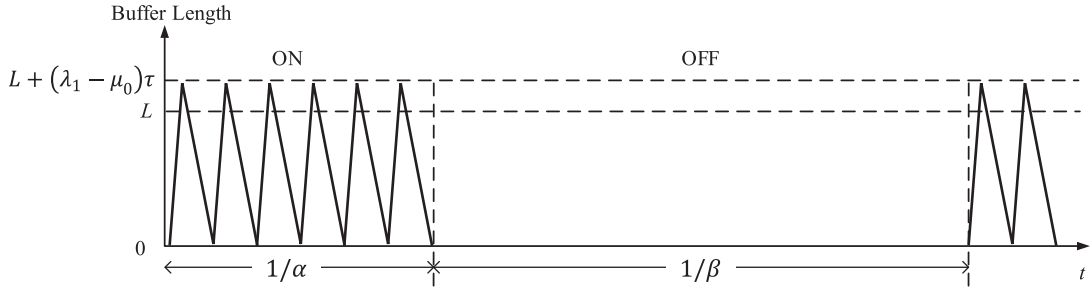
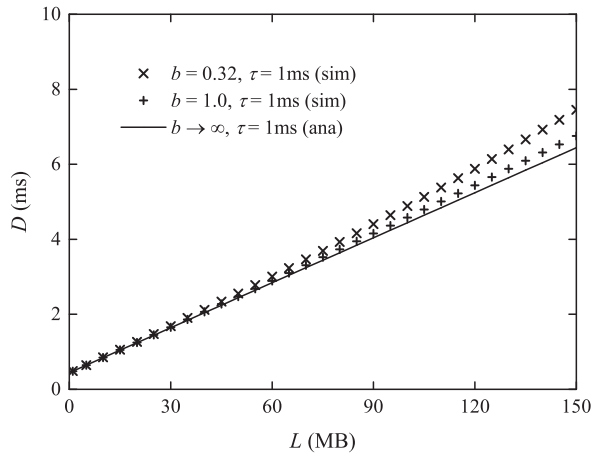


Fig. 12. Queue length evolution with the presence of lightpath set-up delay τ when b is sufficiently large, where $\Delta L = (\lambda_1 - \mu_0)\tau$.

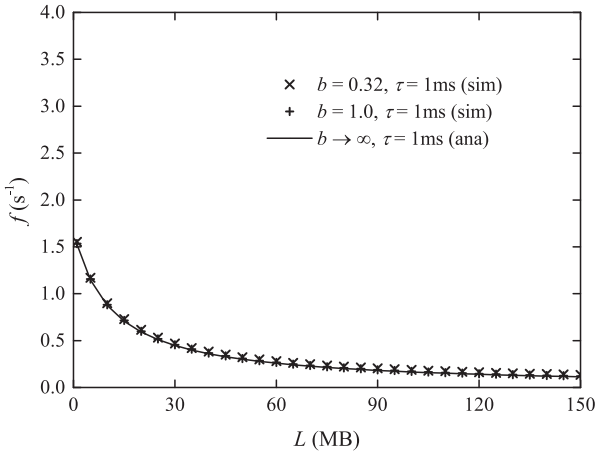
We compare the results in Corollary 1 with the simulation results when $b = 0.32$ and $b = 1$ in Fig. 13. We use the same parameters in Fig. 10, except that τ is now equal to 1 ms. Again, the curve of $b = 0.32$ is already very close to that of $b \to \infty$. Compared with Fig. 9, the only difference is that the curves in Fig. 13 are both left-shifted by

$$(\lambda_1 - \mu_0)\tau = \frac{(90 \times 10^{-9} - 1.64 \times 10^{-9}) \times 10^{-3}}{8 \times 10^6} = 11.045 \text{ MB}.$$

Thus, when the rule in (22) is applied in the practical scenarios where the lightpath set-up delay is τ, it should be amended according to Corollary 1 as follows:



(a) D vs. L



(b) f vs. L

Fig. 13. System performance with $\tau = 1$ ms.

$$L = \frac{\pi_{\text{on}}}{2\overline{\lambda}\widehat{D}} - (\lambda_1 - \mu_0)\tau. \quad (25)$$

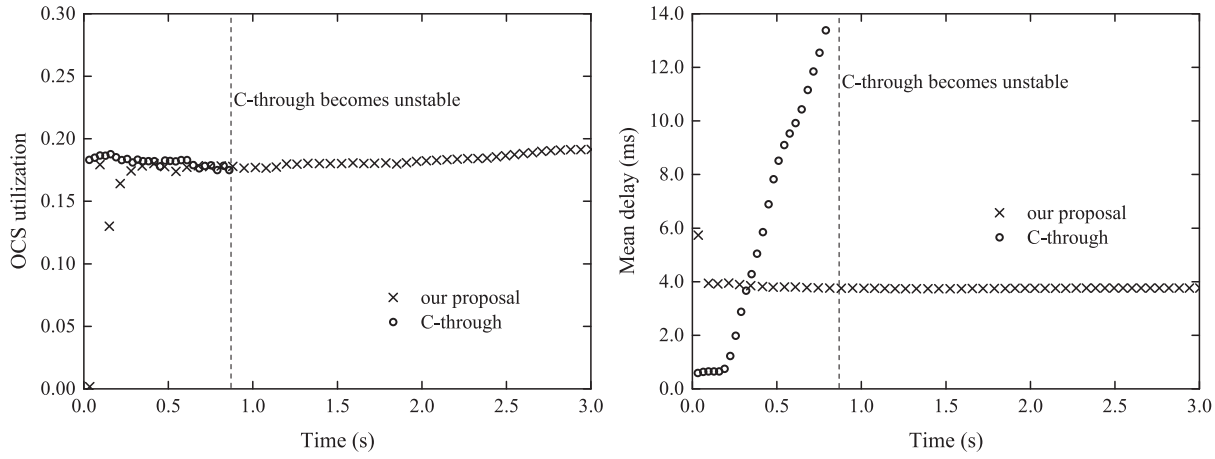
5.2 Case Studies

To study the performance of our strategy, we carry out system-level simulation. In additional to the mean delay, we evaluate OCS utilization to check if our strategy is cost effective. Herein, the OCS utilization is defined as the ratio of the total traffic offloaded by the OCSs to the total capacity of the OCSs. Also, we simulate the maximal buffer occupancy of ToR switches and estimate the bandwidth overhead paid for message exchanges between the ToR switches and the central controller, to verify if our strategy is technically feasible in practice. We further compare our strategy with C-through in terms of the OCS utilization and the mean delay, to demonstrate the effectiveness of our strategy.

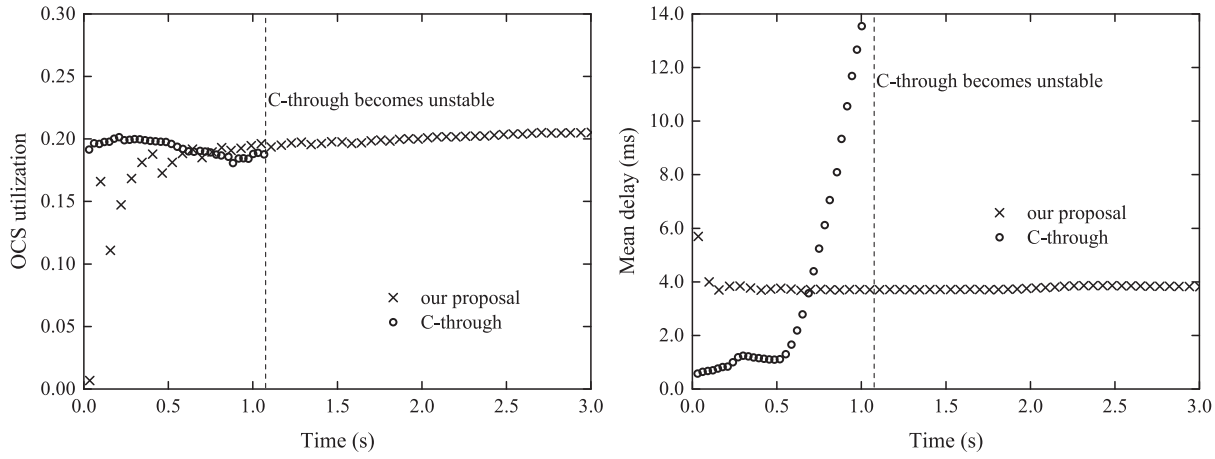
In the simulation, we assume that the packet arrival process of each ToR pair is a two-state Markov-modulated Poisson process (MMPP) [32], where the ratio of the ON period to the OFF period is 1:60. We select this ratio to mimic the traffic pattern measured in [7]. The packet size obeys an exponential distribution with parameter 800 Bytes, and the packet inter-arrival time during the ON (OFF) period is an exponential random variable with mean 0.071 μs (38.4 μs). As a result, the data rates of the ON period and the OFF period are 90 Gb/s and 0.17 Gb/s, respectively. We set that the optical service rate is $\mu_1 = 100$ Gb/s, which has already been available in current data centers and the electrical service rate is $\mu_0 = 1.64$ Gb/s. We consider the case where the lightpath set-up delay is 1 ms and the delay requirement is 4 ms, which is same as the average read (or write) latency in Amazon data centers [33]. According to (25), the buffer threshold $L = 88.96$ MB. Also, we suppose that the data center installs enough OCSs to ensure the probability that the request has to wait in the buffer is very small (e.g., 0.2 percent).

5.2.1 Network With 80 ToR Switches

We first study the case where there are 80 ToR switches in the network. We use 6 80 $\times$ 80 OCSs with 100 Gb/s line rate, which can provide a capacity of 600 Gb/s for each ToR switch. On the other hand, a ToR pair has the burst traffic with probability $1/61$, and thus the average number of the elephant flows from a ToR switch is $79/61 \approx 1.3$. Thus, the average data rate of the elephant flows generated by a ToR switch is 116.6 Gb/s. This indicates the OCS utilization is roughly 19.4 percent. Fig. 14 plots the OCS utilization of our strategy changing with time. The OCS utilization converges to 19.2 percent when $b = 0.5$ and 20.5 percent when $b = 1$.



(a) $b = 0.5$



(b) $b = 1$

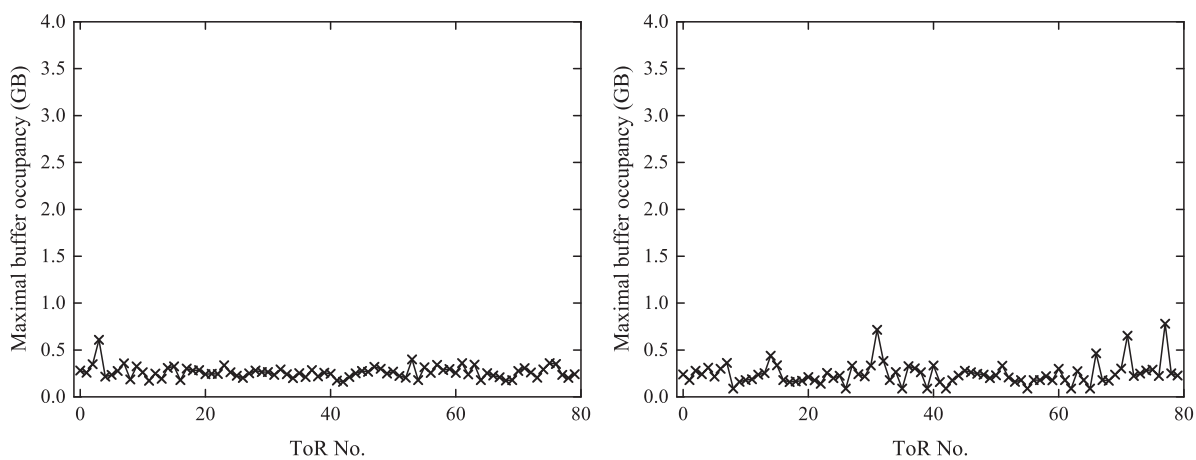
Fig. 14. Performance comparison when the number of ToR switches is 80.

Accordingly, Fig. 14 shows that the mean delay converges to 3.75 ms when $b = 0.5$ and 3.8 ms when $b = 1$, slightly smaller than the delay requirement 4 ms. This confirms that our rule not only can provide delay guaranteed but also is not sensitive to the traffic burstiness.

As a comparison, we also simulate the performance of C-through. The original C-through reported in [10] employs only one OCS in the network, where the number of sever racks is small. However, multiple OCSs are indispensable in a network with 80 ToR switches, since the average data rate of the elephant flows generated by a ToR switch is 116.6 GB/s. We thus have to extend C-through as follows. When an OCS becomes free, the controller creates a traffic matrix according to the buffer occupancies of the ToR pairs that are not connected via the lighpaths. The controller calculates an MWM, and sets up lightpaths for up to N ToR pairs via this OCS, where N is the total number of ToR switches in the data center. In the simulation, we also use 6 OCSs to facilitate the comparison. As Fig. 14 shows, the mean delay of the extended C-through increases unbounded no matter what the burstiness b is. In particular, the OCS utilization of the extended C-through equals to 17.5 percent before it becomes unstable when $b = 0.5$ and 18.7 percent when $b = 1$. This is mainly attributed to the fact that C-through is a synchronous scheduling strategy, as Section 1.1 mentions. Fig. 14 clearly shows that our scheme outperforms the extended C-through.

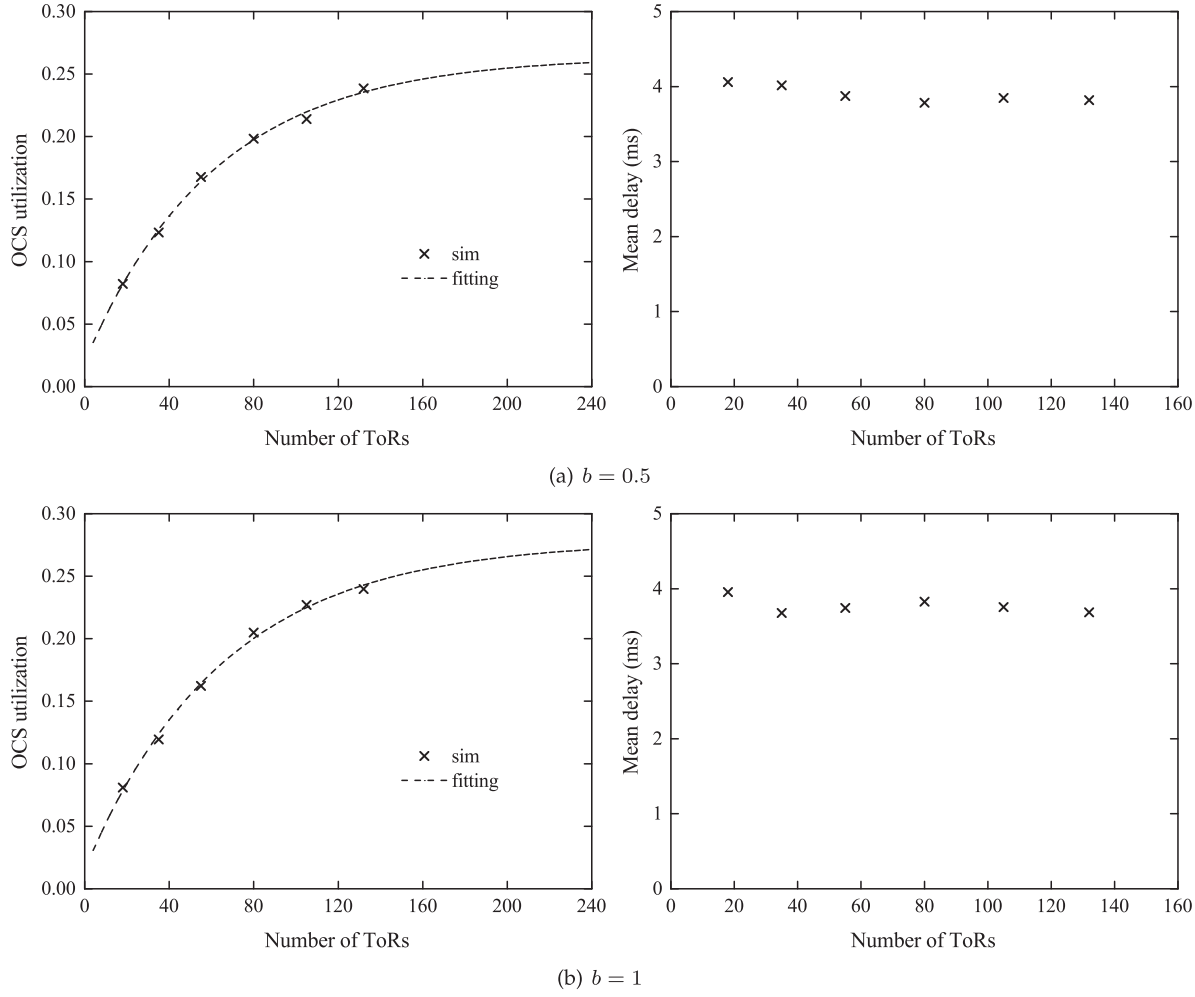
We also study the buffer required by a ToR switch to implement our strategy. We take the ToR switch by Huawei, CloudEngine 6870 [26], as an example. This kind of ToR switch possesses a 4-GB buffer, which is statistically shared by all the ports. As we show in Fig. 14, the mean delay of the traffic of each ToR pair is ~ 3.75 ms, which means that the mean queue length is ~ 0.77 MB. Also, a ToR switch will communicate with 79 other ToR switches. Since all the 79 queues statistically share the same buffer, the average occupancy level of the buffer will converge to $0.77 \times 79 \approx 60.8$ MB, which is much less than 4 GB. To check the situation in the worst case, we measure the maximal buffer occupancy of different ToR switches during the simulation in Fig. 15, which confirms that a 4 GB buffer is far more than enough.

We further estimate the bandwidth overhead paid by our strategy for message exchanges between all the ToR switches and the central controller. Suppose the messages exchanged between the ToR switches and the central controller are formatted based on the OpenFlow protocol, which means the size of the message needed to install a flow entry is 72 Bytes [18]. According to (24), the lightpath set-up frequency in this case is equal to $f = 0.184s^{-1}$, which implies that the bandwidth overhead is only 83.7 KB/s. In addition, the central controller receives about 1163 requests per second, and thus one NOX controller is enough for the central controller, because the number of requests that can be processed by one



(a) $b = 0.5$ (b) $b = 1$

Fig. 15. Maximal buffer occupancy of different ToR switches.



(a) $b = 0.5$

(b) $b = 1$

Fig. 16. System performance versus number of ToR switches in the network.

NOX controller is 30000 per second [16]. We conclude that our optical traffic offloading strategy only imposes a small processing burden to the data center.

5.2.2 Network With Different Number of ToR Switches

Fig. 16 further studies the performance of our scheme under different number of ToR switches. Given the number of ToR switches, we determine the number of OCSs using the Jacobaeus method presented in [34], such that the probability that the lightpath set-up request has to wait in the buffer is less than 0.2 percent. In addition to that the mean delay is almost smaller than 4 ms, it is very interesting to see that the OCS utilization increases with the number of ToR switches. Such an improvement of the OCS utilization is achieved by

the effect of statistical bandwidth multiplexing when the number of ToR pairs is large. We do not simulate the network with a large number of ToR switches, since such a simulation is extremely time consuming. Instead, we simply fit the curve in Fig. 16 to estimate the OCS utilization of our scheme when the number of ToR switches is large. Our fitting shows that the OCS utilization of our scheme could reach $> 26\%$ when the number of ToR switches is 240. Remember that 240 is the port count of the fast optical crossbar demonstrated in [14]. An OCS with 240 ports can connect with 240 ToRs.

There are two potential ways for our strategy to further improve the OCS utilization. The first one is to reconfigure the OCSs when newly arrived requests cannot be satisfied. Clearly, OCS reconfigurations can accommodate more requests at the same time and thus improve the OCS utilization. However, OCS reconfigurations will interrupt the existing lightpaths and increase the volume of message exchanges, which will not only introduce additional delay to the traffic carried by affected lightpaths but also add the system operation overhead. The second way is to set up a two-hop connection passing through an intermediate ToR switch for the request when the network cannot establish a lightpath for the source to the destination. This method can also improve the utilization. The advantage is that it does not interrupt the existing lightpaths, and the disadvantage is that it may degrade the delay performance since some elephant flows will experience one more optical-electrical-optical conversion. From this discussion, we can see that there may exist a trade-off between the OCS utilization and the delay performance. To find out which one can make a better compromise needs more researches in the future.

We note that the asynchronous optical traffic offloading strategy (or the idea of Mahout) can also be applied to optical data center networks, such as OSA [35], WaveCube [36], and MegaSwitch [37]. In these networks, a small fraction of optical resources form a fixed network to accommodate the delay-sensitive traffic, such as mice flows, while the remaining resources are dynamically reconfigured to serve the bulky and delay-tolerant traffic, such as elephant flows. The idea of resource partition in these optical networks is quite similar to that in hybrid optical/electrical data center networks. Thus, the asynchronous optical traffic offloading strategy can be applied to these optical networks as follows. Initially, the ToR switch pair transmits the traffic via the fixed network. Once the buffer of this ToR switch pair exceeds the threshold, the system builds up a lightpath or a multi-hop connection in the reconfigurable network to clean up the traffic. When the buffer becomes empty, the system will tear down the lightpath or the connection, and route the traffic back to the fixed network.

6 Conclusion

In this paper, we propose a threshold-based optical traffic offloading strategy to deal with the burst traffic in hybrid optical/electrical networks. In this strategy, each ToR switch maintains a buffer and a threshold of the queue length for the traffic to each destination ToR switch. The ToR switch sets up a lightpath to offload the traffic when the queue length exceeds the threshold, and tears down the lightpath when the backlog is cleaned up. We develop a fluid-flow model to analyze the optical traffic offloading process. Our analytical result shows that there is a trade-off between the delay performance and the system operation overhead, from which we provide a buffer-threshold selection rule for the optical traffic offloading scheme. Using a case study, we demonstrate that the proposed optical traffic offloading strategy with the buffer-threshold selection rule is suitable for applications in current commercial data center networks. Our case studies demonstrate that the proposed optical traffic offloading strategy with the buffer-threshold selection rule can not only be applied to different kinds of commercial data center networks but also outperform C-through.

Acknowledgments

This work was supported by the National Science Foundation of China under Grant 61671286.

References

[1] T. Benson, A. Akella, and D. A. Maltz, "Network traffic characteristics of data centers in the wild," in *Proc. 10th ACM SIGCOMM Conf. Internet Meas.*, 2010, pp. 267–280.
[2] S. Kandula *et al.*, "The nature of data center traffic: Measurements & analysis," in *Proc. 9th ACM SIGCOMM Conf. Internet Meas.*, 2009, pp. 202–208.
[3] P. Costa *et al.*, "R2C2: A network stack for rack-scale computers," *SIGCOMM Comput. Commun. Rev.*, vol. 45, pp. 551–564, Oct. 2015.
[4] T. Benson, A. Anand, A. Akella, and M. Zhang, "Understanding data center traffic characteristics," in *Proc. 1st ACM SIGCOMM Workshop Res. Enterprise Netw.*, 2009, pp. 65–72.
[5] A. Greenberg *et al.*, "VL2: A scalable and flexible data center network," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 39, pp. 51–62, Oct. 2009.
[6] R. Kapoor *et al.*, "Bullet trains: A study of NIC burst behavior at microsecond timescales," in *Proc. 9th ACM Conf. Emerg. Netw. Experiments Technol.*, 2013, pp. 133–138.
[7] M. Alizadeh *et al.*, "Data center TCP (DCTCP)," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 41, no. 4, pp. 63–74, 2010.
[8] H. Wu, Z. Feng, C. Guo, and Y. Zhang, "ICTCP: Incast congestion control for TCP in data-center networks," *IEEE/ACM Trans. Netw.*, vol. 21, no. 2, pp. 345–358, Apr. 2013.
[9] N. Farrington *et al.*, "Helios: A hybrid electrical/optical switch architecture for modular data centers," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 41, no. 4, pp. 339–350, 2010.
[10] G. Wang *et al.*, "c-Through: Part-time optics in data centers," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 40, no. 4, pp. 327–338, 2010.
[11] H. H. Bazzaz *et al.*, "Switching the optical divide: Fundamental challenges for hybrid electrical/optical datacenter networks," in *Proc. 2nd ACM Symp. Cloud Comput.*, 2011, pp. 30:1–30:8.
[12] H. Liu *et al.*, "Scheduling techniques for hybrid circuit/packet networks," in *Proc. 11th ACM Conf. Emerg. Netw. Experiments Technol.*, 2015, pp. 41:1–41:13.
[13] S. Han, T. J. Seok, N. Quack, B.-W. Yoo, and M. C. Wu, "Large-scale silicon photonic switches with movable directional couplers," *Optica*, vol. 2, no. 4, pp. 370–375, 2015.
[14] T. J. Seok, K. Kwon, J. Henriksson, J. Luo, and M. C. Wu, "240×240 wafer-scale silicon photonic switches," in *Proc. Optical Fiber Commun. Conf.*, 2019, pp. Th1E–5.
[15] M. Noormohammadpour and C. S. Raghavendra, "Datacenter traffic control: Understanding techniques and tradeoffs," *IEEE Commun. Surv. Tut.*, vol. 20, no. 2, pp. 1492–1525, Dec. 2018.
[16] A. Tavakoli *et al.*, "Applying NOX to the datacenter," in *Proc. 8th ACM Workshop Hot Topics Netw.*, 2009.
[17] M. Ghobadi *et al.*, "ProjecToR: Agile reconfigurable data center interconnect," in *Proc. ACM SIGCOMM Conf.*, 2016, pp. 216–229.
[18] M. Al-Fares *et al.*, "Hedera: Dynamic flow scheduling for data center networks," in *Proc. 7th USENIX Conf. Netw. Syst. Des. Implementation*, 2010.
[19] S. B. Venkatakrishnan, M. Alizadeh, and P. Viswanath, "Costly circuits, submodular schedules and approximate carathéodory theorems," in *Proc. ACM SIGMETRICS Conf.*, 2016, pp. 75–88.

[20] T. Mori *et al.*, "Identifying elephant flows through periodically sampled packets," in *Proc. 4th ACM SIGCOMM Conf. Internet Meas.*, 2004, pp. 115–120.
[21] P. Phaal, S. Panchen, and N. McKee, "InMon corporation's sFlow: A method for monitoring traffic in switched and routed networks," RFC 3176, 2001, [Online]. Available: https://www.rfc-editor.org/info/rfc3176, doi: 10.17487/RFC3176.
[22] B. Claise, "Cisco systems netflow services export version 9," RFC 3954, 2004, [Online]. Available: https://www.rfc-editor.org/info/rfc3954, doi: 10.17487/RFC3954.
[23] C. Bi, X. Luo, T. Ye, and Y. Jin, "On precision and scalability of elephant flow detection in data center with SDN," in *Proc. IEEE GLOBECOM Workshops*, 2013, pp. 1227–1232.
[24] A. R. Curtis *et al.*, "DevoFlow: Scaling flow management for high-performance networks," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 41, no. 4, pp. 254–265, 2011.
[25] A. R. Curtis, W. Kim, and P. Yalagandula, "Mahout: Low-overhead datacenter traffic management using end-host-based elephant detection," in *Proc. IEEE INFOCOM*, 2011, pp. 1629–1637.
[26] "CloudEngine 6870 series data center switches data sheet." *Data Sheet*, HUAWEI, 2018. [Online] Available: https://www.router-switch.com/media/upload/product-pdf/huawei-ce6870-series-switches-datasheet.pdf
[27] E. Blanton and M. Allman, "On the impact of bursting on TCP performance," in , 2005, pp. 1–12.
[28] A. Rao *et al.*, "Network characteristics of video streaming traffic," in *Proc. 7th Conf. Emerg. Netw. Experiments Technologies*, 2011, pp. 25:1–25:12.
[29] M. Alizadeh *et al.*, "Less is more: Trading a little bandwidth for ultra-low latency in the data center," in *Proc. 9th USENIX Conf. Netw. Syst. Des. Implementation*, 2012, pp. 253–266.
[30] H. Jiang and C. Dovrolis, "Source-level IP packet bursts: Causes and effects," in *Proc. 3rd ACM SIGCOMM Conf. Internet Measu.*, 2003, pp. 301–306.
[31] L. Wang, T. Ye, and T. T. Lee, "A parallel route assignment algorithm for fault-tolerant Clos networks in OTN switches," *IEEE Trans. Parallel Distrib. Syst.*, vol. 30, no. 5, pp. 977–989, May 2019.
[32] W. Fischer and K. Meier-Hellstern, "The Markov-modulated Poisson process (MMPP) cookbook," *Perform. Eval.*, vol. 18, no. 2, pp. 149–171, 1993.
[33] G. DeCandia *et al.*, "Dynamo: Amazon's highly available key-value store," *ACM SIGOPS Oper. Syst. Rev.*, vol. 41, pp. 205–220, Dec. 2007.
[34] J. Y. Hui, *Switching and Traffic Theory for Integrated Broadband Networks*. Boston, USA: Kluwer Academic Publishers, 1990.
[35] K. Chen *et al.*,"OSA: An optical switching architecture for data center networks with unprecedented flexibility," *IEEE/ACM Trans. Netw.*, vol. 22, no. 2, pp. 498–511, Apr. 2014.
[36] K. Chen *et al.*, "WaveCube: A scalable, fault-tolerant, high-performance optical data center architecture," in *Proc. IEEE Conf. Comput. Commun.*, 2015, pp. 1903–1911.
[37] L. Chen *et al.*, "Enabling wide-spread communications on optical fabric with megaswitch," in *Proc. Conf. Netw. Syst. Des. Implementation*, 2017, pp. 577–593.

Tong Ye (Member, IEEE) received the BS and MS degrees from the University of Electronic Science and Technology of China, Chengdu, China, in 1998 and 2001, respectively, and the PhD degree in electronics engineering from Shanghai Jiao Tong University, Shanghai, China, in 2005. He was with The Chinese University of Hong Kong for one and a half year as a postdoctoral research fellow. He is currently an associate professor with the State Key Laboratory of Advanced Optical Communication Systems and Networks, Shanghai Jiao Tong University. His research interests include the design of optical network architectures, optical network systems and subsystems, and silicon-ring-based optical signal processing.

Jianke Li received the BS degree from Wuhan University, Wuhan, China, in 2016, and the MS degree from Shanghai Jiao Tong University, Shanghai, China, in 2019. He is currently a software development engineer in Pinduoduo inc., Shanghai, China.

Xiaodan Pan received the BS degree in information engineering from Xi'an Jiao Tong University, Xi'an, China, in 2014. She is currently working toward the PhD degree with the State Key Laboratory of Advanced Optical Communication Systems and Network, Shanghai Jiao Tong University, Shanghai, China. Her major research interest includes energy efficient ethernet.

Tony T. Lee (Fellow, IEEE) received the BS degree in electrical engineering from National Cheng Kung University, Taiwan, and the MS and PhD degrees in electrical engineering from the Polytechnic Institute of NYU, Brooklyn, NY. He was with AT&T Bell Laboratories, Holmdel, NJ, from 1977 to 1983. He was with Telcordia Technologies, Morristown, NJ, from 1983 to 1993. From 1991 to 1993, he was a professor of electrical engineering with the Polytechnic Institute of NYU. From 1993 to 2013, he was a chair professor with the Information Engineering Department, The Chinese University of Hong Kong. From 2013 to 2018, he was a Zhiyuan chair professor with the Electronics Engineering Department, Shanghai Jiao Tong University, and an emeritus professor of information engineering with the Chinese University of Hong Kong. He is currently a professor with the School of Science and Engineering, The Chinese University of Hong Kong (Shenzhen). He is a fellow of the HKIE. He has received many awards, including the 1989 Leonard G. Abraham Prize Paper Award from the IEEE Communication Society, the 1999 Outstanding Paper Award from the IEICE of Japan, and the 1999 National Natural Science Award from China. He has served as an editor for the *IEEE Transactions on Communications* and an area editor for the *Journal of Communication Network*.