On modeling optical burst switching networks with fiber delay lines: a novel approach

A. Rajabi\textsuperscript{a}, A. Khonsari\textsuperscript{*b,a}, A. Dadlani\textsuperscript{a,b}

\textsuperscript{a}Institute for Studies in Theoretical Physics and Mathematics, School of Computer Science, Tehran, Iran
\textsuperscript{b}Department of Electrical and Computer Engineering, University of Tehran, Tehran, Iran

Abstract

Optical Burst Switching (OBS) has been widely admitted as a viable candidate for the evolving all-optical network control framework of nowadays Internet core to meet the explosive demand of bandwidth constraints in multimedia and real-time applications. A number of analytical models have been proposed to characterize the behavior of OBS networks. These models, however, do not imitate the actual features of the Fiber Delay Line (FDL) components present in contemporary optical switches. Underscoring the precise behavior of FDLs leads to misleading results regarding major performance measures of the OBS networks. While FDLs behavior has been neglected in many studies, as a crude approximation, a few researches characterize FDLs as conventional buffers present in electronic switching systems. In this paper, we provide an inclusive discussion on the features of FDLs, followed by presenting a novel analytical model to realize the inimitable characteristics of FDLs. The proposed network-level model is based on the results of queuing systems with impatient customers and deterministic impatience time. The viability of the proposed model is validated through extensive simulation experiments.

Key words: Optical burst switching, fiber delay lines, queuing systems with impatient customers, performance evaluation

\textsuperscript{*}Corresponding author. Tel.: +98-21-22828687. Email addresses: alirajabi@ipm.ir (A. Rajabi), ak@ipm.ir (A. Khonsari), a.dadlani@ipm.ir (A. Dadlani)
1. Introduction

Optical Burst Switching (OBS) has been identified as the most promising technology to be employed in the next generation optical Internet. Having the potential in meeting the fast growing bandwidth demand imposed by multimedia applications such as video conferencing, Internet telephony, and digital audio is the reason for such a choice [1]. OBS integrates the benefits of Optical Circuit Switching (OCS) [2] and Optical Packet Switching (OPS) [3] in order to reduce the end-to-end network latency and the overhead involved in packet header-processing. Moreover, employing Dense Wavelength Division Multiplexing (DWDM) in OBS, which enables a transfer rate of more than 1 Tb/s on a single optical fiber, has opened up stimulating challenges to researchers and network engineers so as to seek new paradigms to efficiently utilize the bandwidth inherent in DWDM [4].

As a powerful performance evaluation tool, analytical modeling is essential to provide a platform on which robust judgments can be made on the practicability of proposals regarding various aspects of OBS. Myriad amount of work has been dedicated to providing effective analytical models in order to accurately predict the behavior of an OBS network [1][4][5][6][7][8][9][10][11]. In most of these studies, the Erlang loss system (M/M/c/c queuing system) plays a central role in modeling each OBS switch [4][7]. To capture the precise characteristics of the arrival process, switch models adopting a finite-population queuing system with no waiting room have been lately proposed [5][11]. In all these models however, the impact of Fiber Delay Lines (FDLs) have been completely neglected to relax the analysis complexity of the model. Some other studies have included FDLs by applying an M/M/c/k queuing system [1][25]. However, the behavior of FDLs in these models is approximated by that of conventional electronic buffers. As a result, these models fail to capture the unique characteristics of FDLs that distinguish them from electronic buffers. In [6], a model considering some FDL properties has been presented, yet the analytical model has been developed for a special case where optical fibers carry only two wavelengths, and no closed-form solution has been derived for the general case. Due to the significance of FDLs in enhancing performance of optical networks, presenting effective analytical models taking all FDL features into account are integral to network engineers and designers. This study has been primarily motivated by the lack of such models in the existing literature.

To the best of our knowledge, this is the first study in which a compre-
A comprehensive discussion on FDL properties is presented. The switch architecture considered in this paper is the same as in [1]. Furthermore, we discuss on the following properties that distinguish FDLs from ordinary buffers present in electronic routers:

- how an FDL of definite length can delay a burst for a limited amount of time.
- how a single FDL can be shared by more than one burst at the same time.
- how a burst can hold an FDL busy while being transmitted via a wavelength.

Unlike most of the existing models that have been proposed at either link-level or switch-level [1][2][3][11][13], we propose a novel network-level analytical model based on queuing systems with deterministic impatience [12] to capture the aforementioned characteristics of FDLs. The first two characteristics are completely covered in the model, while the impact of the third one is studied in a special case where the average burst length is less than the FDL length, both measured in units of time. The derivation of the model is straightforward and tractable. The simulation section verifies that the model presented in this paper results in much better predictions of system dynamics, such as blocking probability and network latency, than ordinary $M/M/c/k$ queuing system.

The rest of the paper is organized as follows: in Section 2, we briefly introduce the OBS framework and provide a comprehensive explanation on FDL characteristics. Section 3 introduces the assumptions and notations used throughout the model, which is derived in Section 4. Simulation results are presented in Section 5 followed by concluding remarks in Section 6.

2. Preliminaries of an OBS framework

Switches in an optical network are broadly categorized into edge and core switches [14]. Edge switches collect traffic from various upper layer users such as ATM switches and IP routers, while core switches forward this traffic in the optical domain. Based on their functionality, edge switches are further divided into two types: ingress and egress switches [9]. Figure 1 illustrates the overall view of an optical network. In what follows, we briefly describe
Figure 1: An overall view of an optical network. $I_1$ and $I_2$ are ingress switches, and $E_1$ is an egress switch. $C_1$, $C_2$, $C_3$, and $C_4$ are core switches.

the functionality of ingress, egress, and core switches. A more detailed explanation on optical switches can be found in [26].

2.1. Ingress switches

An ingress switch is where packet aggregation and burst assembly takes place. It comprises of an array of queues, where each queue is associated with a specific egress switch reachable from the ingress switch. Based upon the destination IP address contained in the packet header, every incoming packet is directed to the queue corresponding to its destined egress switch. Each queue undergoes alternative periods of aggregation and transmission. In each aggregation period, IP packets belonging to a queue are aggregated into bursts prior to transmission into the optical domain. As shown in Figure 1, ingress switch $I_2$ assembles the incoming packets from LAN/WAN 2 into bursts and sends them towards the destined egress switch $E_1$ via core switches $C_2$ and $C_3$.

Some of the most common burst assembly algorithms can be classified into timer-based [9], threshold-based [15], and mixed timer/threshold-based [16] algorithms. In the timer-based approach, a timer is set at the beginning of every new assembly cycle, determining the transmission time of the burst into the core network. After a fixed amount of time, all the packets that arrived during that time period are assembled into a burst. In the threshold-based approach, a threshold is specified to determine the generation and transmission time of a burst into the optical network. The incoming packets are stored in the prioritized queues in the ingress node, until the threshold
condition is satisfied. Once the threshold is reached, a burst is created and sent into the optical core. The timeout value for the timer-based schemes should be set carefully. If the value is chosen to be too large, the packet delay at the edge might become intolerable. On the other hand, if the value is too small, too many small-sized bursts will be generated, resulting in control overhead. While timer-based schemes might result in undesirable burst lengths, threshold-based assembly algorithms do not guarantee on the packet assembly delay. A mixed timer/threshold-based algorithm may perform better, especially with self-similar traffic, but may experience higher operational complexity.

A signaling protocol is the procedure through which a control packet reserves resources for the corresponding data burst by guiding it through a routing path. In one-way reservation [7], a control packet reserves resources along the path for the corresponding data burst without any acknowledgement from the destination node. On the contrary, in a two-way reservation, a control packet collects link and topology information instead of reserving resources for the data burst. The acknowledgement packet from the destination node to the source node reserves resources for the corresponding data burst while traversing along the reverse path. Since one-way reservation protocols are more flexible, have lower latency, and are more efficient as compared to two-way reservation protocols, they are mainly adopted in OBS networks. It is important to notice that from a modeling point of view, both control packets and data bursts streaming out of an ingress switch represent essentially the same process and can be used interchangeably in the derivation of the model, provided that the offset time between the control packet and data burst transmission remains unchanged during switching. We adopt the burst stream to explicate our model since it is more concrete to deal with a burst stream than a control packet stream.

2.2. Egress switches

The functionality of an egress switch is simply the reverse replica of an ingress switch. In other words, each burst entering an egress switch undergoes an optical to electrical conversion and is then disassembled into packets. These disassembled packets are then transmitted to the outside non-optical network based on normal IP forwarding as illustrated by the egress switch $E_1$ in Figure 1.
2.3. Core switches

A core optical switch comprises of a number of input and output interfaces, a switching fabric, and a scheduler as shown in Figure 5. Each output interface is connected to an optical link which may be augmented with a number of FDLs. The optical link carries a maximum of \( w \) wavelengths. FDLs are optical buffers that can temporarily delay an arriving burst if no free wavelength on the outgoing fiber is available for its transmission. More on FDLs is covered in subsection 2.5.

Switch scheduler is the central component of a core switch and is responsible for resource reservation, i.e., assigning a wavelength and/or an FDL to every burst. When a control packet reaches a core switch, it undergoes an optical to electrical conversion so that it can be processed in the electronic domain. After conversion, the destined egress switch, corresponding burst length, and time of burst arrival are extracted from the control packet. Based upon this information, the switch scheduler attempts to reserve the required resources for the time when the burst arrives at the switch. This type of reservation is referred to as delayed reservation (DR) [14].

In case of a successful reservation, the control packet is converted back into the optical format and forwarded to the next core switch to establish the rest of the path. Before forwarding the control packet, the burst arrival time to the next switch is calculated, and the newly calculated value is placed in the control packet so that the successive core switch has a precise estimation of burst arrival time for its scheduling process. In case of reservation failure, the control packet is simply dropped.

3. FDL versus electronic buffer

An optical link may be allocated a number of FDLs, which are optical buffers used to temporarily delay an incoming burst if all wavelengths of the output link are busy transmitting other bursts. Previous work shows that even employing a limited number of FDLs greatly improves the performance of the optical network [1]. This section emphasizes on the fundamental differences between buffers and FDLs and their impact on the analytical model.

1. Concurrent Access to Single FDL (CASF): In a queuing system, as a newly arrived job finds all servers busy, it waits inside a buffer, if any available, for one of the servers to become free. While residing inside the buffer, it does not share the space with any other incoming jobs. As a
result, in a queuing system with $D$ buffers, the maximum number of jobs awaiting service at any given time is bounded to $D$. However, a single FDL might be shared by more than one burst at the same time, as shown in Figure 2. For the sake of simplicity, assume that only one wavelength and FDL are available. In Figure 2(a), burst $B_0$ with length $L/4$ (measured in time units) enters the system at time $t_0$ and finds the wavelength busy, but the FDL free. Therefore, it enters the FDL. In Figure 2(b), the head of $B_0$ reaches the middle of the FDL at time $t_0 + L/2$. At the same time, another burst $B_1$ enters the system, finding the wavelength busy and the FDL free (since $B_0$ released the FDL $L/4$ time units earlier), as a result of which $B_1$ enters the FDL. In Figure 2(c), both $B_0$ and $B_1$ travel inside the FDL at time $t_0 + 3L/4$, implying that:

On contrary to a buffer in a queuing system which can accommodate at most one job at any given time, an FDL can be shared by more than one burst at the same time.

An immediate result is that employing a queuing system with $w$ servers and $D$ buffers to model an output interface with $w$ wavelengths and $D$ FDLs might result in an overestimation of the blocking probability. The model claims that any arriving job finding all $w$ service lines busy and $D$ buffers occupied is blocked, while an arriving burst to an output interface might not necessarily be blocked even when all wavelengths are busy and
each FDL holds at least one burst. CASF occurs frequently when the average burst length is less than that of the FDL, both measured in units of time. In this case, the head of a burst can enter an FDL while some other bursts are still being held in it (as in Figure 2(b)). Consequently, if the average burst length equals the FDL length, the impact of CASF on the model diminishes in long term. This is true especially in cases where the burst length distribution has a small variance, meaning that it slightly oscillates around the mean value. If the average burst length is less than the FDL length, the model can be adjusted to include the impact of CASF.

2. Concurrent FDL and Wavelength Holding (CFWH): In a queuing system, a job inside the system might be in either of the following two states: being served on one of the servers or waiting in one of the buffers for service release. On the other hand, a burst may be so long that even after commencing service, it still blocks the FDL, forbidding other bursts from entering it. Figure 3 illustrates the CFWH scenario. At time $t_0$, burst $B_0$ with length $4L/3$ enters the system, finds the only wavelength busy, and enters the only available FDL which is free. $L$ time units later, the head of the burst reaches the end of the FDL while still blocking the FDL. By this time, the wavelength on the optical link is freed (otherwise the burst would have never entered the FDL and had been dropped at the

![Figure 3: The Concurrent FDL and Wavelength Holding (CFWH) scenario.](image)

(a) At time $t_0$, the head of $B_0$ enters the FDL. (b) At time $t_0 + L$, the head of $B_0$ reaches the FDL and is about to start service, while its tail still remains outside the FDL, thus blocking the FDL and prohibiting other bursts from entering it. This implies that a burst can hold an FDL busy even while being transmitted over a wavelength.
instant of its arrival), and $B_0$ starts its service. From this time until time $t_0 + 4L/3$, at which $B_0$ releases the FDL, no other burst can enter it. In other words, $B_0$ holds the wavelength and blocks the FDL concurrently, implying that:

While a buffer in a queuing system is freed as soon as the job residing in it commences service, an FDL might be blocked by a burst even after its service (transmission) begins.

In contrast to CASF, CFWH might result in an underestimation of the actual blocking probability. Again, consider a queuing system with $w$ service lines and $D$ buffers. An incoming job finding less than $w + D$ jobs in the system definitely joins the system and remains in it until served. On the other hand, at an output interface with $w$ wavelengths and $D$ FDLs, a newly arrived burst might be blocked even at times when the total number of bursts at the interface is less than $w + D$, as some bursts might both, hold a wavelength and block an FDL simultaneously.

CFWH occurs frequently if the average burst length is greater than that of the FDL. Similar to CASF, if the average burst length equals the FDL length, and the burst length distribution has a small variance, then the effect of CFWH on the model becomes negligible in long term. It can even be claimed that on satisfying these two conditions, CASF and CFWH neutralize the estimation impacts of each other on the analytical model.

3. FDL Holding Time Constraint (FHTC): In a queuing system, a buffer can accommodate a job for any arbitrary amount of time. On the contrary, as the length of an FDL is limited to some constant value, it can hold a burst for only a limited amount of time, proportional to its length, by the end of which the burst must leave the FDL. On reaching an output interface, a burst either starts its transmission immediately over a free wavelength or enters an unblocked FDL if assured that by the end of its transversal inside the FDL, a free wavelength would be available for its transmission. Otherwise, it is dropped and considered as lost. In other words, the burst either balks at entering the interface and is dropped or enters the interface and is served. A similar concept exists for the term balking in queuing systems. It means that on arrival, a job decides whether to enter a system or not. But, once entering the system, it remains there until served. The condition upon which the entrance decision is made may differ from one balking system to another. In an
output interface (as a balking system), the decision is based on the \textit{waiting time until beginning of service} (WTUBS) which must be less than or equal to the FDL length.

In contrast to a balking system, in a \textit{reneging} queuing system, a newly arrived job always joins the queue, but waits only for a limited amount of time for its service to begin. If the service does not start within this limited period, the job leaves the system and is lost. In other words, joining the reneging system does not necessarily mean receiving service. The term \textit{deterministic impatience} is used to describe a reneging queue in which the WTUBS is bounded to some constant value.

Due to [17], if the entrance decision in a balking queue is based on the WTUBS, a reneging queue with the same WTUBS gives the same blocking probability and waiting time distribution of served jobs as the original balking queue, given that there are infinitely many number of buffers available.

We present a reneging-based model for an output interface which is balking in nature. However, the number of FDLs allocated to an optical link is limited to some constant value (not infinity), and hence a reneging-based model is not applicable to the balking output interface. On the other hand, we argued earlier that a single physical FDL is equivalent to \( w \) virtual FDLs, where \( w \) is the number of wavelengths on an optical fiber and is usually a large number. Consequently, even a few physical FDLs result in a large number of virtual FDLs which may be considered as infinity. As a result, a reneging queue is employed to model an output interface with the hope that the number of virtual FDLs is \textquotedblleft large enough\textquotedblright; to be thought of as infinity so that the queue acts as a \textquotedblleft balking queue\textquotedblright; in terms of blocking probability and average waiting time of served jobs. In Section 5, numerical examples are given to clarify what we mean by \textquotedblleft large enough\textquotedblright;.

4. Assumptions and notations

In this section, the notations and assumptions used in the derivation of the analytical model are presented. Tables 1 and 2 summarize, respectively, the notations and functions used in this paper.
Table 1: Notations used in the derivation of the model

<table>
<thead>
<tr>
<th>Notation</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>$I_j$</td>
<td>An arbitrary ingress switch ($1 \leq j \leq m$).</td>
</tr>
<tr>
<td>$I = I_1, \ldots, I_m$</td>
<td>The set of all ingress switches.</td>
</tr>
<tr>
<td>$E_j$</td>
<td>An arbitrary egress switch ($1 \leq j \leq n$).</td>
</tr>
<tr>
<td>$E = E_1, \ldots, E_n$</td>
<td>The set of all egress switches.</td>
</tr>
<tr>
<td>$Q_{ij}$</td>
<td>The $i^{th}$ queue in the AoQ of $I_j$ ($1 \leq i \leq n, 1 \leq j \leq m$).</td>
</tr>
<tr>
<td>$T_{ij}$</td>
<td>Random variable denoting the length of an arbitrary aggregation period of $Q_{ij}$ in terms of time.</td>
</tr>
<tr>
<td>$1/\lambda_{ij}$</td>
<td>Known constant denoting the mathematical expectation of $T_{ij}$, i.e., $E[T_{ij}] = 1/\lambda_{ij}$.</td>
</tr>
<tr>
<td>$\pi_{ji}$</td>
<td>The probability that the next coming burst from $I_j$ is destined for $E_i$ ($1 \leq i \leq n, 1 \leq j \leq m$).</td>
</tr>
<tr>
<td>$t_0^j$</td>
<td>Known constant denoting the offset time between the transmission of a control packet and its corresponding data burst from $I_j$.</td>
</tr>
<tr>
<td>$t_C$</td>
<td>Known constant denoting the switching time of an arbitrary core switch $C$, i.e., the time taken by the head of a burst to cross the switching fabric of $C$.</td>
</tr>
<tr>
<td>$1/\mu$</td>
<td>Known constant denoting the average burst length in terms of time.</td>
</tr>
<tr>
<td>$l$</td>
<td>An arbitrary link.</td>
</tr>
<tr>
<td>$f$</td>
<td>An arbitrary FDL.</td>
</tr>
<tr>
<td>$P$</td>
<td>An arbitrary path, i.e., a sequence of links connecting an ingress switch to an egress switch.</td>
</tr>
<tr>
<td>$d$</td>
<td>Known constant denoting the number of physical FDLs allocated to $l$.</td>
</tr>
<tr>
<td>$x_l$</td>
<td>Known constant denoting the length of $l$ in terms of length unit.</td>
</tr>
<tr>
<td>$t_p^l$</td>
<td>The propagation time on $l$, i.e., the time taken by the head of a burst to reach from one end of the link to the other end.</td>
</tr>
<tr>
<td>$L$</td>
<td>Known constant denoting the length of $f$ in terms of time units.</td>
</tr>
<tr>
<td>$P_l$</td>
<td>The set of all paths including $l$ as a link.</td>
</tr>
<tr>
<td>$\lambda_l$</td>
<td>Long-run rate of burst arrival at $l$.</td>
</tr>
<tr>
<td>$\delta_l$</td>
<td>Long-run rate of burst departure from $l$.</td>
</tr>
<tr>
<td>$\lambda_{lp}$</td>
<td>Long-run rate of burst arrival at $l$ through $P$ given that $P$ includes $l$.</td>
</tr>
<tr>
<td>$\delta_{lp}$</td>
<td>Long-run rate of burst departure from $l$ through $P$ given that $P$ includes $l$.</td>
</tr>
<tr>
<td>$b_l$</td>
<td>The blocking probability at $l$.</td>
</tr>
</tbody>
</table>

(continued on next page...)

11
Table 1: Notations used in the derivation of the model (continued)

<table>
<thead>
<tr>
<th>Notation</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>$q_l$</td>
<td>The average queuing delay at $l$.</td>
</tr>
<tr>
<td>$b_N$</td>
<td>The blocking probability of the network, i.e., the probability that a randomly chosen burst is blocked somewhere on its path from source to destination.</td>
</tr>
<tr>
<td>$ANL_P$</td>
<td>The average network latency of $P$, i.e., the amount of time taken by a burst traveling on $P$ until its tail is received at the destination switch from the time its corresponding control packet is sent from the source switch, given that the burst is not blocked at any intermediate core switch.</td>
</tr>
<tr>
<td>$ANL$</td>
<td>The average network latency of an arbitrary burst, given that it is not blocked at any intermediate core switch.</td>
</tr>
</tbody>
</table>

Table 2: Function definitions

<table>
<thead>
<tr>
<th>Function</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>$prev(l, P)$</td>
<td>Returns the link preceding $l$ on $P$. If $l$ is the first link of $P$, it returns NIL.</td>
</tr>
<tr>
<td>$next(l, P)$</td>
<td>Returns the link following $l$ on $P$. If $l$ is the last link of $P$, it returns NIL.</td>
</tr>
<tr>
<td>$head(l)$</td>
<td>Returns the switch, one of the output interfaces of which is connected to $l$.</td>
</tr>
<tr>
<td>$tail(l)$</td>
<td>Returns the switch, one of the input interfaces of which is connected to $l$.</td>
</tr>
<tr>
<td>$ingress(P)$</td>
<td>Returns the ingress switch connected to the first link of $P$.</td>
</tr>
<tr>
<td>$egress(P)$</td>
<td>Returns the egress switch connected to the last link of $P$.</td>
</tr>
<tr>
<td>$cores(P)$</td>
<td>Returns the set of all core switches in the path between $ingress(P)$ and $egress(P)$.</td>
</tr>
<tr>
<td>$lastlink(P)$</td>
<td>Returns the last link of $P$ connecting to an egress switch.</td>
</tr>
<tr>
<td>$index(I_j)$</td>
<td>Returns $j$.</td>
</tr>
<tr>
<td>$index(E_i)$</td>
<td>Returns $i$.</td>
</tr>
</tbody>
</table>

The assumptions made below have been widely used in the previous literature [1][6][7][8][9][10][11][13]:

1. Each egress switch in $E$ is reachable from every ingress switch in $I$.
2. The routing algorithm directs a burst through a path resulting in the minimum number of hops from source to destination.
3. $Q_{ij}$ stores IP packets destined for $E_i$.
4. $T_{ij}$ is exponentially distributed with mean $1/\lambda_{ij}$.
5. Control packet processing time at every core switch is negligible and can be ignored.
6. The wavelength of a randomly chosen burst is uniformly distributed over $w$ available wavelengths.
7. Burst length distribution is exponential with mean $1/\mu$. The exponential distribution of burst length is adopted for mathematical tractability. Furthermore, there exists some kind of dependency between the length of an aggregation period and mean burst length. The longer the aggregation period, the greater is the average burst length. This dependency is made insignificant by setting the average burst length to a constant value of $1/\mu$. However, simulation results show that if the $\lambda_{ij}$ values are chosen close to each other, such an assumption does not greatly influence the model.
8. On entering an FDL, a burst can leave it as soon as a wavelength becomes free. This is also a simplifying assumption, as it is stated in the preceding section that a burst may leave an FDL only at exit points. If a burst is traveling between two exit points and a wavelength becomes free during this period, the burst must proceed until the next exit point before being sent over the link. However, simulation results show that if the exit points are chosen to be close to each other, the behavior of an FDL with finite number of exit points can be approximated by that of an FDL which allows a burst to leave at any arbitrary point in the FDL.
9. There is no blocking at any link connecting an ingress switch to the network.
10. For any arbitrary link $l$, the arrival process at $l$ approximates to a Poisson process with rate $\lambda_l$ [19].

5. Analytical model

The model is derived in three steps. First, an arbitrary optical link is considered, and its performance metrics, such as blocking probability and average queuing delay, are obtained assuming that the arrival rate at the link is known. Next, a recursive set of equations are presented to obtain the arrival rate at an arbitrary link of the network. The blocking probability and average latency of an entire network are then derived in the third step. Finally, this section is concluded with a discussion on the impact of wavelength converters on the performance of optical networks.
5.1. Step I: Performance metrics of an arbitrary link

In this subsection, we obtain the performance metrics related to an arbitrary optical link, say \( l \). If \( \text{head}(l) \in I \), then \( b_l = 0 \) and \( q_l = 0 \) (according to assumption 10). Therefore, throughout this step, we assume that \( \text{head}(l) \notin I \).

Considering \( w \) wavelengths on \( l \) and \( d \) physical FDLs, as well as assumption 7 and Lemma 3, one may conclude that a multi-server finite-buffer queuing system with Poisson arrival and exponential service is the best choice for capturing the behavior of \( l \). However, taking the FHTC property of FDLs into consideration, we employ an \( M/M/w/N \) queue with deterministic impatience to model \( l \), where \( N \) is the total queue capacity (including service lines and buffers). Deterministic impatience is adopted since the length of an FDL is limited to constant \( L \). The better performance of this queueing system with regard to the ordinary \( M/M/w/N \) queue in terms of blocking probability and average network latency will be shown in the next section.

To obtain the number of buffers, notice that there are \( d \) physical FDLs, each equivalent to \( w \) virtual FDLs. Furthermore, a single virtual FDL can accommodate on average \( \lceil L\mu \rceil \) bursts at a given time. As a result, the total number of buffers, denoted by \( D \), can be obtained as:

\[
D = dw\lceil L\mu \rceil. \tag{1}
\]

By assuming \( L \geq 1/\mu \) (according to assumption 8), we have already eliminated the impact of CFWH on the model. Now, by defining the total number of buffers as in (3), instead of simply defining it as \( dw \), we actually take the impact of CASF into account since a single virtual FDL no longer corresponds to a single buffer, but rather to \( \lceil L\mu \rceil \) buffers. The total system capacity can be obtained from:

\[
N = w + D = w(1 + d\lceil L\mu \rceil). \tag{2}
\]

\( M/M/w/N \) queue with deterministic impatience has been thoroughly studied in \([20, 21]\), and its system size distribution for the steady state has been obtained to be:

\[
p_0^{-1} = \sum_{j=0}^{w} \frac{\rho_l^j}{j!} + \frac{\rho_l^w}{w!} \sum_{j=1}^{D} \frac{\chi_l^j}{\prod_{i=0}^{j-1}(w\mu + C_{w+j-i})},
\]

\[
(\sum_{j=0}^{w} \frac{\rho_l^j}{j!} p_0), \quad 1 \leq j \leq w, \tag{3}
\]
\[ p_{w+j} = \frac{\lambda_i^j}{w! \prod_{i=0}^{j-1} (w \mu + C_{w+j-i})} p_0, \quad 1 \leq j \leq D, \]

where \( p_j \) is the probability of \( j \) jobs in queue, \( \rho_l = \lambda_l / \mu \), and \( C_{w+j} \) is the long-term rate of time-out (reneging) when there are \( w+j \) jobs in queue and is obtained from:

\[ C_{w+j} = \frac{L^{j-1}e^{-w\mu L}}{\int_0^L t^{j-1}e^{-w\mu t} dt}. \quad (4) \]

To obtain the blocking probability at \( l \), notice that there are two types of blocking in an \( M/M/w/N \) queue with deterministic impatience. A job may balk at entering the queue if it finds all \( w \) servers and \( D \) buffers occupied by jobs, the probability of which is \( p_N \). It may also time-out after joining the queue if the waiting time until beginning of the service exceeds its patience time, which in this case is equal to \( L \). To obtain the latter probability, notice that there is no time-out when the number of jobs in queue is less than or equal to \( w \). Furthermore, the long-run rate of time-out for \( w+j \) jobs in queue is \( C_{w+j} \), and the probability of having \( w+j \) jobs is \( p_{w+j} \). As a result, the total rate of time-out is given by:

\[ \sum_{j=1}^{D} C_{w+j} p_{w+j}. \quad (5) \]

Dividing by \( \lambda_l \) gives the probability of time-out. Consequently, the blocking probability at \( l \) is derived as:

\[ b_l = p_N + \frac{1}{\lambda_l} \sum_{j=1}^{D} C_{w+j} p_{w+j}. \quad (6) \]

Next, we obtain the average queuing delay at \( l \). For this purpose, we require to know the average number of jobs, denoted by \( E[S] \), and the probability of time-out given that a job joins the queue, denoted by \( P_{T|J} \), which are given respectively as:

\[ E[S] = \sum_{j=1}^{w+D} jp_j, \quad (7) \]
and

\[ P_{T|J} = \frac{1}{\lambda_l(1-p_N)} \sum_{j=1}^{D} C_{w+j} p_{w+j}. \]  

Equation (11) can be interpreted as follows: \((1 - p_N)\) is the probability of joining the queue which when multiplied by \(P_{T|J}\) gives the probability of time-out derived earlier. Applying Little’s formula [22] yields:

\[ E[S] = \lambda_l (1 - p_N) \left( L P_{T|J} + \left( q_l + 1/\mu \right) \left( 1 - P_{T|J} \right) \right), \]  

where \(\lambda_l(1-p_N)\) and \((LP_{T|J} + (q_l + 1/\mu) (1 - P_{T|J}))\) are the effective arrival rate [22] at \(l\), and the average response time of an arbitrary job joining the queue, respectively. By response time, we mean the time period from when the job enters the queue to the time when its service completes and leaves the queue. The response time consists of two parts: if the burst times-out, its response time is \(L\) (the patience time before time-out), otherwise the response time is equal to queuing delay plus the service time. The probabilities of the two cases are given by \(P_{T|J}\) and \(1 - P_{T|J}\), respectively. Re-arranging (11) yields:

\[ q_l = \frac{E[S]}{\lambda_l (1 - p_N) \left( 1 - P_{T|J} \right)} - \frac{LP_{T|J}}{(1 - P_{T|J})} - \frac{1}{\mu}, \]  

which gives a formula to obtain the average queuing delay at \(l\).

5.2. Step II: Obtaining arrival rate at \(l\)

In the previous step, we obtained the blocking probability and average queuing delay of an arbitrary link \(l\), assuming that the arrival rate at \(l\) is \(\lambda_l\). In this step, we calculate \(\lambda_l\).

\(\lambda_l\) is the superposition of the arrival rates of the individual paths in \(P_l\) at \(l\), i.e.,

\[ \lambda_l = \sum_{P \in P_l} \lambda_l^P. \]  

The arrival rate of a path \(P\) at \(l\) equals the departure rate of \(P\) from the link preceding \(l\) on \(P\) if such a link exists. If \(l\) is the first link of \(P\) connecting an ingress switch to a core switch, then \(\lambda_l^P\) equals a fraction of
the total traffic rate generated by ingress($P$) and destined for egress($P$). Let $\varphi = index(\text{ingress}(P))$ and $\gamma = index(\text{egress}(P))$. Consequently, $\lambda_l^P$ is given by:

$$\lambda_l^P = \begin{cases} 
\delta_{\text{prec}(l,P)}^{\text{prec}(l,P)} & \text{head}(l) \notin I \\
\pi_{\varphi \gamma} \lambda_{\text{head}(l)} & \text{otherwise}
\end{cases},$$

(12)

where $\lambda_{\text{head}(l)}$ is as defined in Lemma 1. A formula for $\delta_l^P$, i.e., the departure rate from $l$ on $P$, can be easily obtained as:

$$\delta_l^P = \lambda_l^P (1 - b_l),$$

(13)

where $b_l$ is as in (9). Recall that $b_l = 0$ if $\text{head}(l) \in I$. As long as the optical network is cycle-free, (9) and (14-16) can be employed to recursively calculate the arrival rate at every arbitrary link.

5.3. Step III: Obtaining Average Network Latency and Network Blocking Probability

In this step, we derive the network blocking probability ($b_N$) as well as the average network latency ($ANL$). To obtain $b_N$, notice that $1 - b_N$ is the probability of burst delivery to egress switches and can be defined as the long-term ratio of burst delivery to egress switches to the long-term rate of burst generation at ingress switches. Therefore,

$$b_N = 1 - \frac{\sum_{\text{tail}(l) \in E} \delta_l}{\sum_{\text{head}(l) \in I} \lambda_l},$$

(14)

where $\delta_l = \lambda_l (1 - b_l)$.

To obtain $ANL$, first consider a single path, say $P$, connecting an ingress switch to an egress switch. Five components contribute to the total latency on $P$ namely, the offset time between control packet and data burst transmission, the propagation time of every link on $P$, the switching time of every core switch visited by a burst traveling on $P$, the average queuing delay of every link on $P$, and the interval from the time the head of a burst is received at an egress switch to the time its tail is received (which equals $1/\mu$ on average). With $\varphi$ as defined earlier, we consequently have:

$$ANL_P = t_0^\varphi + \sum_{l \in P} (t_l^l + q_l) + \sum_{C \in \text{cores}(P)} t_C + \frac{1}{\mu},$$

(15)
where $t_p^l = x_l / (\text{speed of light})$.

$ANL$ can be defined as a weighted summation of the average latencies of every path between a pair of ingress and egress switches where the weight of the latency of each path is the ratio of the burst delivery rate through that path to the total rate of burst delivery to egress switches, i.e.,

$$ANL = \sum_P \frac{\delta_{\text{last link}}(P)}{\sum_{l \in E} \delta_l} ANL_P.$$

(16)

5.4. Wavelength Converters (WCs)

In the derivation of the model, we have implicitly assumed that core switches benefit from the presence of wavelength converters to change the wavelength of a burst if necessary. A finite-buffer multi-server queue employed to model a link with $w$ wavelengths is valid only if a burst can be transmitted via every available wavelength and buffered inside every available FDL, which implies that its wavelength can be converted as necessary. In this subsection, we slightly tune the model to cover the case in which core switches lack the presence of WCs. In the rest of this subsection, “$\gamma$-burst” refers to a burst with wavelength $\gamma$.

As pointed out earlier, a single optical fiber with $w$ wavelengths can be viewed as $w$ virtual links, each capable of transmitting bursts of a specific wavelength. Similarly, a physical FDL is equivalent to $w$ virtual FDLs each capable of optically buffering bursts of a particular wavelength. In case of no WC, a $\gamma$-burst can only be transmitted via the virtual link and buffered inside the virtual FDLs corresponding to wavelength $\gamma$. Virtual links and FDLs corresponding to wavelengths other than $\gamma$ are completely ignored by the scheduler when it tries to reserve resources for a $\gamma$-burst. It follows that when no WC is available, bursts with different wavelengths arriving at a link do not compete for the available resources, but the contention rather exists only among the bursts with the same wavelength. Consequently, the arrival stream of bursts at a link could be decoupled into $w$ non-overlapping streams, one for each wavelength, where each stream is allocated with a single virtual link and $d$ virtual FDLs (since there are $d$ physical FDLs, the total number of virtual FDLs allocated to each stream is $d$). Figure 10 illustrates the models of an arbitrary link, say $l$, with and without WCs.

As can be observed in Figure 10(a), the single queue with $w$ wavelengths and $dw$ FDLs is modified to $w$ queues, each associated with one wavelength and $d$ FDLs. Due to the uniform distribution of a burst (assumption 6)
Figure 4: (a) The model of an optical link with wavelength converters (WCs). There is a single stream of bursts with rate $\lambda_l$. Each arriving burst may utilize every $dw$ FDL and every $w$ wavelength made available. (b) The model of an optical link without WCs. There are $w$ distinguished stream of bursts, each having a specific wavelength and rate $\lambda_l/w$. Each stream can utilize only one wavelength and $d$ FDLs allotted to the wavelength of that stream.

as well as the decomposition property of Poisson process [22], the stream of bursts into each queue in Figure 10(b) is also Poisson with rate $\lambda_l/w$. As the $w$ queues in Figure 10(b) are equivalent and independent of each other, one may conclude that the blocking probability and average queuing delay at $l$ is the same as those of one of the queues in Figure 10(b) (an $M/M/1/N$ queuing system with deterministic impatience, arrival rate of $\lambda_l/w$, and $N = 1+d[L\mu]$). Such a queue is a special case of the queue studied in step I of the model derivation. Consequently, its blocking probability and average queuing delay can be obtained from (9) and (13), respectively. Once the performance metrics of $l$ are obtained this way, one can proceed with steps II and III as before.

We end this section by arguing on how the presence of WCs enhance the performance of optical networks by reducing the blocking probability. It is

19
well-known in queuing theory that a queuing system in which the buffers and servers are shared between arrivals results in lesser blocking probability than a queue in which buffers are divided between the servers, and each arrival is assigned to one server and may utilize buffers exclusively allocated to that server. A special case of the above assertion has been deeply studied in [18]. As an optical fiber equipped with WCs can be modeled with a queue of the former type, it is justified why WCs reduce the blocking probability. In the next section, we verify the validity of the claim made above.

6. Numerical results

We constructed a discrete event-driven simulator in the Ptolemy environment [23] to verify the accuracy of the proposed model. An optical network consisting of 3 ingress, 8 core, and 3 egress switches was adopted as the case study. The topology of the network is as shown in Figure 11. The average burst length is taken to be 3 time units in all scenarios. All ingress switches generate traffic with the same rate in every simulation experiment. All optical fibers carry the same number of wavelengths and are assigned the same number of FDLs. \( t_p^l \) and \( t_C \) are assumed to be negligible compared to other parameters contributing to the total network latency. Each simulation scenario was made to run until its steady state, i.e., until no further increase in simulation time altered the collected statistics appreciably. On average, 8 batches of bursts, each consisting of 1,000,000 bursts generated by ingress switches, was sufficient to reach the steady-state. Statistical data gathered during the first batch was discarded to avoid distortion due to the initial warm-up conditions.

In the remainder of this section, we consider three analysis namely, \( A_1 \), \( A_2 \), and \( A_3 \). In \( A_1 \), the blocking probability and average queuing delay at an optical fiber are obtained from (9) and (13), respectively, and the number of FDLs assigned to each optical fiber is calculated from (4). Consequently, \( A_1 \) takes both, CASF and FHTC into consideration. In \( A_2 \), the blocking probability and average queuing delay are obtained from an ordinary \( M/M/w/N \) queuing system, and the number of FDLs assigned to each fiber is \( dw \). \( A_2 \) ignores both, CASF and FHTC, yet it has been widely used to model an optical fiber in the existing literature. \( A_3 \) is the same as \( A_1 \), except that the number of FDLs assigned to each optical fiber is \( dw \) (rather than \( dw\lfloor L\mu \rfloor \)), i.e., \( A_3 \) takes FHTC into account; however, it ignores CASF. In what follows,
we show that $A_1$ (the approach suggested in this paper) results in better performance predictions of system dynamics than $A_2$ and $A_3$.

Let $\zeta$ be the performance measure of interest and $\zeta(.)$ denote the function which returns the value of $\zeta$ obtained from its argument (for example, if $\zeta$ is the blocking probability, $\zeta(A_1)$ is the value of blocking probability obtained from $A_1$). The relative error of $A_1$ with respect to the value obtained from simulation is defined as:

$$R(A_i) = \frac{|\zeta(A_i) - \zeta(S)|}{\zeta(S)} \times 100,$$

where $\zeta(S)$ is the value obtained from simulation and $|x|$ is the absolute value of $x$. $R(A_i)$ is the function we use to measure the effectiveness of $A_i$. The lower the value of $R(A_i)$, the more effective is the analysis $A_i$.

6.1. The equivalence of balking and reneging queuing systems

Towards the end of Section 2, we argued that a reneging queuing system cannot be employed to model an optical fiber which is balking in nature unless the number of buffers is large enough to be considered as infinity. To show that a finite-buffer reneging queue is a good approximation of a balking queue, we simulated both. Figures 12 and 13 depict respectively, the blocking probability and average queuing delay of both queues as a function of the number of buffers for $\lambda = 2$, $1/\mu = 3$, $w = 4$, and $L = 3$, where
\(\lambda, \mu, w,\) and \(L\) are the average arrival rate, average service rate, number of servers, and the waiting time until beginning of service, respectively. As can be observed, for \(D \geq 10\), both balking and reneging queues result in the same blocking probability and average queuing delay of served jobs. Moreover, for \(D < 10\), the difference between Figures 12 and 13 is negligible. For example, in the worst case \((D = 6)\), the blocking probability of the reneging queue is 0.351, which differs in only 1.2% with respect to that of the balking queue which is 0.3467. Figures 12 and 13 imply that for traffic intensity \(\rho \leq 1.5\), finite-buffer balking and reneging queues behave alike, and the reneging-based model presented in this paper holds well.

6.2. Blocking probability and average network latency as a function of traffic intensity (a comparison of M/M/w/N and M/M/w/N with deterministic impatience)

Figure 14 illustrates the blocking probability as a function of traffic generation rate for \(w = 4, d = 4,\) and \(L = 3\). Diamonds and squares represent the analysis results of \(A_1\) and \(A_2\), respectively, while triangles denote the values obtained from simulation. Since \(L = 1/\mu = 3\), \(A_1\) and \(A_3\) are essentially the same, and hence only \(A_1\) is considered. As can be observed, the simulation results are in good agreement with those of \(A_1\). For example, in the worst case when \(\lambda = 2\), \(R(A_1)\) is only 7.74, while \(R(A_2)\) is 100. The increase in relative error is noticeable as one adopts \(A_2\) instead of \(A_1\) to model an optical

![Figure 6](image)

**Figure 6:** Comparison of balking and reneging queuing systems: blocking probability versus number of buffers.
Figure 7: Comparison of balking and reneging queuing systems: average queuing delay of served jobs versus number of buffers.

fiber.

Figure 15 depicts the average network latency as a function of traffic generation rate. Again, it can be seen that $A_1$ results in better predictions than $A_2$. For $\lambda = 2$, the relative error of $A_1$ is approximately 8.5, while that of $A_2$ is 19.17. Based on the values adopted for base offset time ($= 10$) and average burst length ($= 3$), the curve related to $A_2$ implies that burst traveling through the network experience no queuing delay at intermediate switches which is in contradiction with the results obtained from simulation. The difference between the values obtained from $A_1$ and simulation is due to the assumptions made in the derivation of the model.

6.3. The role of number of wavelengths on blocking probability and average network latency

Figures 16 and 17 depict respectively, the blocking probability and average network latency as a function of the number of wavelengths when $\lambda_{I_j} = 2$ ($1 \leq j \leq m$), $d = 2$, and $L = 3$. Although the number of physical FDLs ($d$) is constant, by increasing the number of wavelengths, the number of virtual FDLs increases as well. In Figure 16, for $w = 4$, $R(A_1)$ is 7.79 and $R(A_2)$ is 77.55, while in Figure 17, these values are 7.8 and 13.72, respectively. Thus, even in this case, $A_1$ results in better estimations in both scenarios.

It should be noticed that increasing the number of wavelengths has a great impact on reducing the blocking probability and queuing delay. By
Figure 8: Blocking probability as a function of traffic rate.

Figure 9: Average network latency as a function of traffic rate.

doubling the number of wavelengths from 4 to 8, the blocking probability decreases by approximately 88 percent, and by tripling this number, the blocking probability becomes zero. Network queuing delay decreases as well by a factor of 69 percent as the number of wavelengths increases from 4 to 8. The network queuing delay is calculated by the difference between the values in Figure 17 and the sum of the base offset time and the average burst length (which is 13). The final point regarding Figures 16 and 17 is that by increasing the number of wavelengths, the analysis results of $A_1$ and $A_2$...
converge towards each other. This is due to the fact that by increasing the wavelength count, incoming bursts are less likely to be buffered inside FDLs and are rather transferred instantly via optical fibers. $A_1$ is essentially the impatient version of $A_2$. Notice that in this case, the number of buffers is the same in both approaches since $L = 1/\mu$ in both cases. As an incoming burst often finds a free wavelength when the number of wavelengths is large, it rarely becomes impatient. Thus, the difference between $A_1$ and $A_2$ fades away as $w$ increases.

6.4. The role of number of FDLs on blocking probability and average network latency

In Figure 18, the influence of the number of physical FDLs on the network blocking probability is illustrated. In this case, $\lambda_I = 2$ ($1 \leq j \leq m$), $w = 2$, and $L = 3$. As shown in Figure 18, the difference between the blocking probability values obtained from $A_1$ and $A_2$ initially increases with increase in the number of FDLs (when the number of FDLs increases from 2 to 5) and then remains constant for the rest of the simulation (when the number of FDLs increases from 6 to 11). In the case of three FDLs, $R(A_1)$ is 5.3, while $R(A_2)$ is 33.9. However, with increase in number of FDLs from 3 to 10, $R(A_1)$ remains the same, while $R(A_2)$ rises to 100. The relative error obtained for the latter case is high, which shows the appropriateness of the adopted $A_1$ approach. It can be seen that with increase in the number of FDLs, the blocking probability remains constant, concluding that the

![Figure 10: Blocking probability as a function of the number of wavelengths.](image-url)
Figure 11: Average network latency as a function of the number of wavelengths.

blocking probability is merely affected by the slight increase in the number of physical FDLs. In the following two subsections, we shall show that for better FDL utilization, one should consider increasing the FDL length or the number of exit points on an FDL.

Figure 19 presents the average latency behavior of the optical network with respect to the number of physical FDLs for the same set of parametric values as in Figure 18. In order to show the independence between the average network latency and the number of existing physical FDLs, we calculated the average latency experienced by the entire network for various values of FDL counts, ranging from 2 to 11. The resulting trends show that the network latency is not affected by the FDL count throughout the simulation. The better performance of $A_1$ is justified by calculating the relative errors for each of the two methods. In the case of seven FDLs, the values obtained for $R(A_1)$ and $R(A_2)$ are 12.04 and 24.69, respectively. Thus, it can be inferred that based on the trends in Figures 18 and 19, the FDL count alleviates neither the total network blocking probability nor the average network latency.

6.5. The role of FDL length on blocking probability and average network latency

In Figures 20 and 21, we study the impact of CASF. For the sake of clarity, analysis results of $A_1$ and $A_3$ are only considered. Figure 20 plots the blocking probability against the FDL length for $\lambda_{ij} = 1$ (1 ≤ j ≤ m),
Figure 12: Blocking probability as a function of the number of physical FDLs.

Figure 13: Average network latency as a function of the number of physical FDLs.

For small FDL lengths, $A_1$ and $A_3$ behave similarly. However, as the FDL length increases from 3 to 12, $A_1$ estimates the blocking probability more accurately than $A_3$. As an example, for $L = 8$, $R(A_1)$ is 7, while $R(A_3)$ is 24. The same conclusion can be made from Figure 21 which depicts the average network latency in terms of FDL length. At first glance, it may seem that $A_3$ predicts the network latency better than $A_1$. However, simulations show that by increasing the FDL length from 3 to 20, the average network latency increases monotonically. This is logical as with
increase in FDL length, bursts can be delayed longer inside FDLs before being transmitted via optical fibers, which in turn increases the queuing delay (and subsequently the network latency). \( A_1 \) agrees with the simulation, i.e., it increases monotonically as the FDL length increases. On the other hand, for \( L \geq 12 \), \( A_2 \) does not change appreciably as \( L \) increases. Although not shown in Figure 21 due to space limitations, for larger values of \( L \), \( R(A_1) < R(A_3) \).

It can be noticed that as the length of an FDL grows, \( A_3 \) approaches to \( A_2 \) since the latter is an asymptotic case of \( A_3 \) in which patience time tends to infinity. Finally, notice that on contrary of increasing the number of physical FDLs, which had little impact on enhancing the performance of optical networks, increasing the FDL length can prove effective in improving performance. By tripling the length of an FDL from 3 to 12, the blocking probability diminishes by a factor of 37 percent. The FDL length can be virtually increased by employing FDLs with feedback architecture [4].

7. Conclusion

Recent studies have convincingly demonstrated that FDLs in Internet optical core switches play a critical role in enhancing network performance by obtaining enough time in order to not overload a potential congested downstream switch and henceforth avoid contention. Not taking FDLs behavior into account leads to the inaccuracy of the calculated performance measures of the network model and thus leads to incongruity with a real network.

![Figure 14: Blocking probability as a function of FDL length.](image-url)
Figure 15: Average network latency as a function of FDL length.

Previous analytical results of OBS networks subject to crude investigation of FDLs are inadequate to capture the realistic behavior of optical switches. This paper has presented a comprehensive discussion of FDL features and their difference with conventional electronic buffers. Then a new analytical model for OBS networks based on queuing systems with deterministic impatience time has been proposed so as to capture the important features of FDLs. By including wavelength converters (WCs) in the model, we also have shown the invaluable role of WCs in further reducing the network blocking probability and thus decreasing the average network latency of messages in the network. Simulation results have revealed that the proposed model provides more accurate prediction of the performance measures compared to the $M/M/c/k$ model of the past studies. As a future direction of this study, we aim at extending the model so as to support real-world multimedia traffic distributions.
References


