ELSEVIER

Performance Evaluation 33 (1998) 113–136

Performance analysis of a photonic single-hop ATM switch architecture, with tunable transmitters and fixed frequency receivers

Martin W. McKinnon [a,1], George N. Rouskas [b,*], Harry G. Perros [b]

[a] *GTRI, Georgia Institute of Technology, Atlanta, GA 30332, USA*

[b] *Department of Computer Science, North Carolina State University, Raleigh, NC 27695-7534, USA*

Received 14 February 1997; received in revised form 11 December 1997

Abstract

We consider a photonic asynchronous transfer mode (ATM) switch based on the single-hop wavelength division multiplexing (WDM) architecture with tunable transmitters and fixed frequency receivers. The switch operates under a schedule that masks the transceiver tuning latency. We analyze approximately a queueing model of the switch in order to obtain the queue-length distribution and the cell-loss probability at the input and output ports. The analysis is carried out assuming two-state Markov modulated Bernoulli process (MMBP) sources that capture the notion of burstiness and correlation, two important characteristics of ATM traffic, and non-uniform destination probabilities. We present results which establish that the performance of the switch is a complex function of a number of system parameters, including the load balancing and scheduling algorithms, the number of available channels, and the buffer capacity. We also show that the behavior of the switch in terms of cell-loss probability as these parameters are varied cannot be predicted without an accurate analysis. Our work makes it possible to study the interactions among the system parameters, and to predict, explain, and fine tune the performance of the switch. © 1998 Elsevier Science B.V. All rights reserved.

Keywords: Optical networks; Photonic ATM switch architecture; Markov modulated Bernoulli process (MMBP); Wavelength division multiplexing (WDM); Discrete-time queueing networks

1. Introduction

One of the issues in evolving today's asynchronous transfer mode (ATM) networks is that of developing switch architectures that can effectively switch cells at very high data rates (currently, data rates on the order of a few tens of Gigabits per second per port are envisioned). Over the last decade, a great deal of research has been devoted to the design of fast cell switches suitable to a broadband integrated services environment; surveys of some of these architectures may be found in [1,2]. Mainstream research and development activities in the area of broadband switching are focused exclusively on electronics-based

[*] Corresponding author. Tel.: 919-515-3860; fax: 919-515-7925; e-mail: rouskas@csc.ncsu.edu.

[1] E-mail: bmck@comlab.gtri.gatech.edu.

PII S0166-5316(98)00003-0

technologies which have attained a high level of maturity. On the other hand, the deployment of optics is limited to mere point-to-point transmission where the technology has proven successful in a short time span.

Given the continued rapid progress in lightwave technology (including the demonstration of fast tunable transceivers [3,4], the development of erbium-doped fiber amplifiers [5], and guided-wave optical switching [6]), and the anticipated total dominance of optical fiber in the wired network, the issue of deeper penetration of optics naturally arises. Given the potential of optical solutions to cell switching, the possibility of employing photonics to implement switching functions hitherto reserved for electronics is currently being explored (see [7] and references thereof). However, there remain at least two major technical challenges to be overcome before one can contemplate the design of *all-optical* switches. First, there is the difficulty of "controlling light by light", and secondly, the technologies for implementing buffering in the optical domain are not yet mature enough. Consequently, the most likely scenarios for near-term photonic cell switching will involve an optical switching fabric with electronic control and buffering.

It has long been recognized that wavelength division multiplexing (WDM) will be instrumental in bridging the gap between the speed of electronics and the virtually unlimited bandwidth available within the optical medium. The wavelength domain adds a significant new degree of freedom to network design, allowing new network concepts to be developed. With a few exceptions (e.g., [8–11]), however, most broadcast WDM architectures that have appeared in the literature require a large number of wavelengths and/or very fast tunable transceivers [12,13]. Furthermore, the performance analysis of these architectures has been typically carried out assuming uniform traffic and memoryless arrival processes (see most of the above references, as well as [14,15]). However, it has been shown that, in order to study correctly the performance of a switch, one needs to use traffic models that capture the notion of burstiness and correlation, and which permit non-uniform output port destinations [16,17].

In this paper we revisit the well-known and widely studied single-hop broadcast-and-select WDM architecture [18]. Unlike previous work, however, we develop a queueing-based decomposition algorithm to study the performance of a single-hop ATM switch architecture operating under schedules that mask the transceiver tuning latency [10]. The analysis is carried out using arrival models that capture the important characteristics of ATM type of traffic, and non-uniform destination probabilities. Our work makes it possible to capture the complex interaction among the various system parameters such as the arrival processes, the number of available channels, and the scheduling and load balancing algorithms. To the best of our knowledge, such a comprehensive performance analysis of a single-hop WDM architecture has not been done before.

Section 2 presents our system model and provides some background information. The performance analysis of the switch is presented in Sections 3 and 4, numerical results are given in Section 5, and we conclude the paper in Section 6.

2. The ATM switch under study

2.1. The switch architecture

We consider an optical switch architecture with N input ports and N output ports interconnected through a broadcast passive star (the switch fabric) that can support $C \leq N$ wavelengths $\lambda_1, \ldots, \lambda_C$ (see Fig. 1). Each input port is equipped with a laser that enables it to inject signals into the optical medium. Similarly,

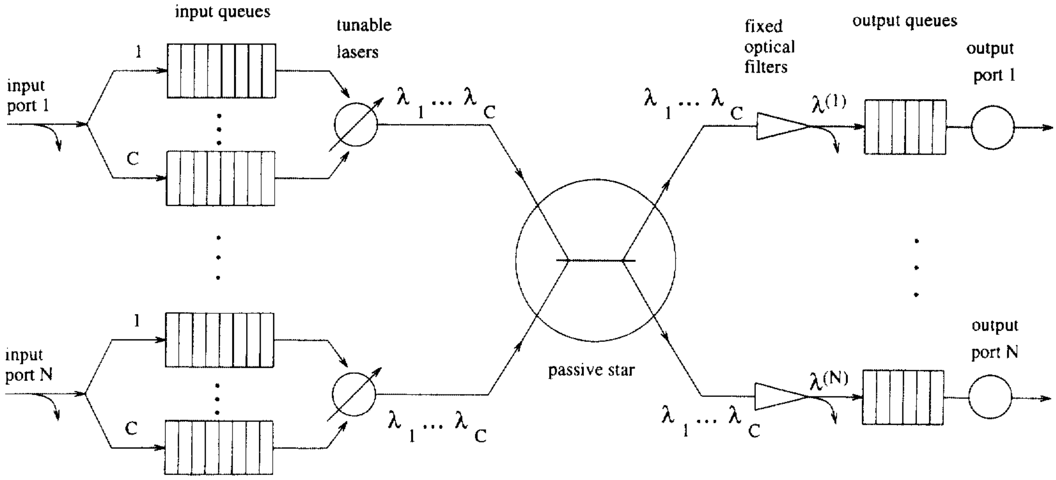


Fig. 1. Queueing model of a switch architecture with N ports and C wavelengths.

each output port is capable of receiving optical signals through an optical filter. The laser at each input port is assumed to be tunable over all available wavelengths. The optical filters, on the other hand, are fixed to a given wavelength. Let $\lambda(j)$ denote the receive wavelength of output port j. Since $C \leq N$, a set $\mathcal{R}_c$ of output ports may be sharing a single receive wavelength λ_c:

$$\mathcal{R}_c = \{j : \lambda(j) = \lambda_c\}, \quad c = 1, \ldots, C. \tag{1}$$

Sets $\mathcal{R}_c$ will typically be obtained by running a load balancing algorithm [19].

The switch operates in a slotted mode. Since there are N ports but $C \leq N$ channels, each channel must run at a rate N/C times faster than the rate of the input links (N/C need not be an integer). The rate of an output link is equal to the rate of an input link. Thus, we distinguish between *arrival* slots (which correspond to the ATM cell transmission time at the input–output link rate) and *service* slots (which are equal to the cell transmission time at the channel rate within the switch). Obviously, the duration of a service slot is equal to C/N times that of an arrival slot. Without loss of generality, we assume that all input links are synchronized at arrival slot boundaries; similarly for output links. On the other hand, all C channels internal to the switch are synchronized at service slot boundaries.

The switch employs electronic queueing at both the input and output ports, as Fig. 1 illustrates. Cells arrive at an input port i and are buffered at a finite capacity queue, if the queue is not full. Otherwise, they are dropped. As Fig. 1 indicates, the buffer space at each input port is assumed to be partitioned into C independent queues. Each queue c at input port i contains cells destined for the output ports which listen to a particular wavelength λ_c, $c = 1, \ldots, C$. This arrangement eliminates the head-of-line problem, and permits an input port to send a number of cells back-to-back when tuned to a certain wavelength. We let $B_{ic}^{(\mathrm{in})}$ denote the capacity of the queue at input port i corresponding to wavelength λ_c.

Cells buffered at an input port are transmitted on an FIFO basis onto the optical medium by the port's laser. This transmission takes place on an appropriate service slot which guarantees that the cell will be correctly received by its destination output port. Upon arriving at the output port, the cell is once again placed in a finite capacity buffer. Let $B_j^{(\mathrm{out})}$ denote the buffer capacity of output port j. Cells arriving at an output port to find a full buffer are lost. Cells in an output buffer are also served on an FIFO basis.

Interest in such a photonic switch architecture arises for several reasons:

- it is highly modular, allowing the switch to grow relatively easily by adding ports and wavelengths;
- it is scalable, since the number of wavelengths need not be equal to the number of ports, and since the data rate within the switch needs only be N/C times the rate of the input–output links;
- its hardware requirements, in terms of the number of transceivers per port, is minimum;
- it can be reconfigured [19] to adapt to changing traffic patterns or to overcome failures of ports or transceivers; and,
- it does not require extremely fast tunable transmitters (as explained below), and thus can be built using *currently available* tunable optical devices.

2.2. Transmission schedules

One of the potentially difficult issues that arises in a WDM environment is that of coordinating the various transmitters/receivers. Some form of coordination is necessary because (a) a transmitter and a receiver must both be tuned to the same channel for the duration of a cell's transmission, and (b) a simultaneous transmission by one or more input ports on the same channel will result in a *collision*. The issue of coordination is further complicated by the fact that tunable transceivers need a non-negligible amount of time to switch between wavelengths. For the Gigabit per second rates envisioned here, and for 53 byte ATM cells, the tuning latency of state-of-the-art tunable lasers or filters can be as long as several times the size of a service slot [3]. Consequently, approaches that require each tunable transmitter to send a single cell and then switch to a new wavelength, will suffer a high tuning overhead and will result in a very low throughput.

In a recent paper [10], it was shown that careful scheduling can mask the effects of arbitrarily long tuning latencies, making it possible to build high-throughput photonic ATM switches using *currently available* lightwave technology. The key idea is to have each tunable transmitter send a *block* of cells on a wavelength before switching to another one. The main result of Rouskas and Sivaraman [10] was a set of new algorithms for constructing near-optimal (and, under certain conditions, optimal) schedules for transmitting a set of traffic demands $\{a_{ic}\}$. Quantity a_{ic} represents the number of cells to be transmitted by input port i onto channel λ_c per frame. The schedules are such that no collisions occur. Furthermore, they are easy to implement in a high speed environment, since the order in which the various input ports transmit is the same for all channels [10].

Quantity a_{ic}, $i = 1, \ldots, N$, $c = 1, \ldots, C$, can be seen as the number of service slots per frame allocated to input port i, so that the port can satisfy the required quality of service of its incoming traffic intended for wavelength λ_c. By fixing a_{ic}, we indirectly allocate a certain amount of the bandwidth of wavelength λ_c to port i. This bandwidth could be equal to the effective bandwidth [20] of the total traffic carried by input port i on wavelength λ_c. In general, the estimation of the quantities a_{ic}, $i = 1, \ldots, N$, $c = 1, \ldots, C$, is part of the call admission algorithm, and it is beyond the scope of this paper. We note that as the traffic varies, a_{ic} may vary as well. In this paper, we assume that quantities a_{ic} are fixed, since this variation will more likely take place over larger scales in time.

We assume that transmissions by the input ports onto wavelength λ_c follow a schedule as shown in Fig. 2. This schedule repeats over time. Each frame of the schedule consists of M arrival slots. Within each frame, input port i is assigned a_{ic} *contiguous* service slots for transmitting cells on channel λ_c. These a_{ic} slots are followed by a *gap* of $g_{ic} \geq 0$ slots during which no port can transmit on λ_c. This gap may be necessary to ensure that input port $i + 1$ has sufficient time to tune from wavelength λ_{c-1} to λ_c before it starts transmission. The algorithms in [10] are such that the number of slots in most of the gaps is equal to either

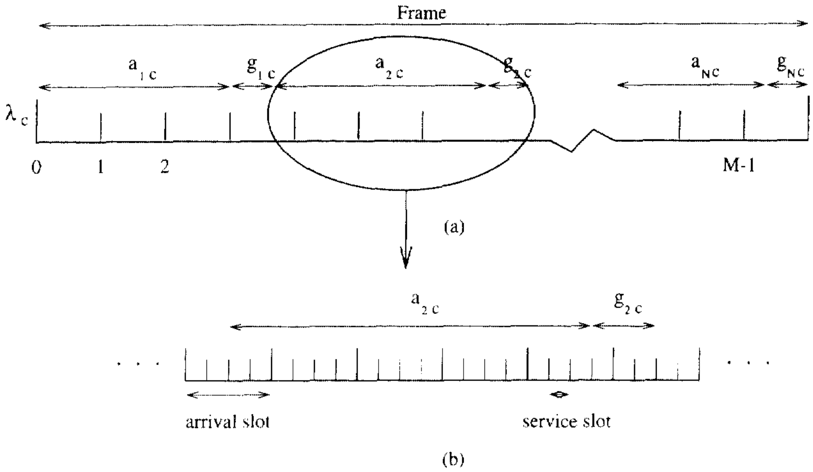


Fig. 2. (a) Schedule for channel λ_c. (b) The detail corresponding to input port 2.

zero or a small integer. Thus, the length of the schedule is very close to the lower bound $\max_i\{\sum_{c=1}^{C} a_{ic}\}$. Note that in Fig. 2 we have assumed that an arrival slot is an integer multiple of service slots. This may not be true in general, and it is not a necessary assumption for our model. Observe also that, although the frame begins and ends on *arrival* slot boundaries, the beginning or end of transmissions by a port does not necessarily coincide with the beginning or end of an *arrival* slot (although it is, obviously, synchronized with *service* slots).

Since the schedule is based on per-port traffic information which is available at the switch itself, it can be computed once and become available to all ports via shared memory (each input port needs access to a different part of the schedule; similarly for output ports). We also emphasize that, in a WDM environment with a large tuning latency, per-port throughput may be low. This observation provides the motivation for using the schedules in [10] which achieve a high aggregate throughput even in the presence of large latencies. Finally, under low loads, the length of the schedule may be determined by the transmitter tuning requirements, meaning that many slots may be empty. Fortunately, we have shown [9] in a similar environment that having empty slots does not affect the throughput, precisely because the traffic load is low.

2.3. *Traffic model*

The arrival process to each input port of the switch is characterized by a two-state Markov modulated Bernoulli process (MMBP), hereafter referred to as 2-MMBP. A 2-MMBP is a Bernoulli process whose arrival rate varies according to a two-state Markov chain. It captures the notion of burstiness and the correlation of successive interarrival times, two important characteristics of ATM type of traffic. For details on the properties of the 2-MMBP, the reader is referred to [21]. (We note that the algorithm for analyzing the switch was developed so that it can be readily extended to MMBPs with more than two states.)

We assume that the arrival process to port i, $i = 1, \ldots, N$, is given by a 2-MMBP characterized by the transition probability matrix $\mathbf{Q}_i$, and by $\mathbf{A}_i$ as follows:

$$\mathbf{Q}_i = \begin{bmatrix} q_i^{(00)} & q_i^{(01)} \\ q_i^{(10)} & q_i^{(11)} \end{bmatrix} \quad \text{and} \quad \mathbf{A}_i = \begin{bmatrix} \alpha_i^{(0)} & 0 \\ 0 & \alpha_i^{(1)} \end{bmatrix}. \tag{2}$$

In (2), $q_i^{(kl)}$, $k, l = 0, 1$, is the probability that the 2-MMBP will make a transition to state l given that it is currently at state k. Obviously, $q_i^{(k0)} + q_i^{(k1)} = 1$, $k = 0, 1$. Also, $\alpha_i^{(0)}$ ($\alpha_i^{(1)}$) is the probability that an arrival will occur in a slot at state 0 (1). Transitions between states of the 2-MMBP occur only at the boundaries of *arrival* slots. We assume that the arrival process to each input port is given by a different 2-MMBP.

Let r_{ij} denote the probability that a cell arriving to input port i will have j as its destination output port. We will refer to $\{r_{ij}\}$ as the *routing* probabilities; this description implies that the routing probabilities can be input port dependent and non-uniformly distributed. The destination probabilities of successive cells are not correlated. That is, in an input port, the destination of one cell does not affect the destination of the cell behind it. This assumption is reasonable when the switch is used as part of a backbone network. Given these assumptions, the probability that a cell arriving to port i will have to be transmitted on wavelength λ_c is

$$r_{ic} = \sum_{j \in \mathcal{R}_c} r_{ij}, \quad i = 1, \ldots, N. \tag{3}$$

3. Queueing analysis

In this section, we analyze the queueing network shown in Fig. 1, which represents the tunable-transmitter, fixed-receiver switch under study. The arrival process to each input port is assumed to be a 2-MMBP, and the access of the input ports to the wavelengths is governed by the schedule described in Section 2.2. We analyze this queueing network in order to obtain the queue-length distribution in an input or output port, from which performance measures such as the cell-loss probability and the cell delay can be obtained. Although we do not present a delay analysis in this paper, an approximate expression for the delay distribution to traverse the switch can be found in [22].

3.1. *Input side analysis*

In this section, we obtain the queue-length distribution of an input queue. We first sketch an exact decomposition of the corresponding queueing network which, however, is not scalable to large systems. Then, we present in detail an approximation method which, as we will show later, gives accurate results.

3.1.1. Exact queueing analysis

We first observe that we can analyze the input side of the switch by decomposing it into N sub-systems, each corresponding to an input port, and analyzing each sub-system in isolation. Because of the fact that (a) the arrival processes to the various input queues are independent, (b) the way the schedule is constructed (i.e., that different inputs transmit to the same wavelength at different times), and (c) the operation of the input ports is independent of the operation of output ports, this decomposition is exact. Furthermore, we can analyze the sub-system corresponding to input port i by defining a $(C + 2)$-dimensional stochastic process $(x, y_1, \ldots, y_C, z)$, where:

- x represents the arrival slot number within a frame ($x = 0, 1, \ldots, M - 1$),
- y_c indicates the number of cells in the input queue servicing λ_c ($y_c = 0, 1, \ldots, B_{ic}^{(\mathrm{in})}$; $c = 1, \ldots, C$), and
- z indicates the state of the 2-MMBP describing the arrival process to this port, i.e., $z = 0, 1$.

It is easy to verify that this process defines a Markov chain, and thus, the steady-state joint occupancy distribution of the C queues of input port i can be obtained. Unfortunately, the state space of the Markov

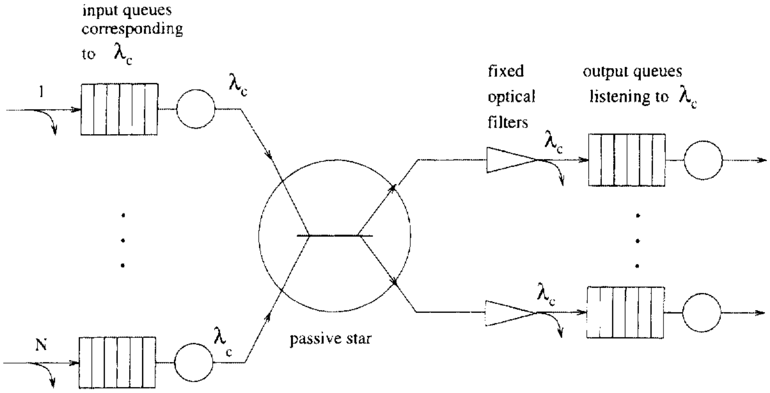

input queues corresponding to λ_c
1
N
λ_c
passive star
fixed optical filters
output queues listening to λ_c

Fig. 3. Queueing sub-network for wavelength λ_c.

chain grows in size as $O(2M\prod_{c=1}^{C} B_{ic}^{(\mathrm{in})})$. As a result, this analysis can only be applied to trivial systems. In the next section, we describe an approximate decomposition that can be applied to large systems.

3.1.2. Approximate queueing analysis

Our main approximation is to assume that arrivals to each queue of a given input port are independent and are generated by the original 2-MMBP (which characterizes the arrival process to the input port) appropriately thinned using the routing probabilities r_{ic}.

Assuming independence of arrivals among the queues of each input port, the original queueing network can now be decomposed into C sub-networks, one per wavelength, as in Fig. 3. For each wavelength λ_c, the corresponding sub-network consists of N input queues, and all the output queues that listen to wavelength λ_c. Each input queue of the sub-network is the queue associated with wavelength λ_c in each input port of the switch. That is, the ith input queue of this sub-network is the cth queue of input port i. Since throughout this section we only consider the sub-network corresponding to λ_c, we will simply refer to this queue as "input queue i". These input queues will transmit to the output queues of the sub-network over wavelength λ_c. In view of this decomposition, it suffices to analyze a single sub-network, since the same analysis can be applied to all other sub-networks.

Consider now the sub-network for wavelength λ_c. We will analyze this sub-network by decomposing it into individual input and output ports. As discussed in the previous section, each input queue i of the sub-network is only served for a_{ic} consecutive service slots per frame. During that time, no other input port is served. Input queue i is not served in the remaining slots of the frame. In view of this point, there is no dependence among the input queues of the sub-network, and consequently each one can be analyzed in isolation in order to obtain its queue-length distribution.

From the queueing point of view, the queueing network shown in Fig. 3 can be seen as a polling system in discrete time. Despite the fact that polling systems have been extensively analyzed, we note that very little work has been done within the context of discrete time (see, e.g., [23]). In addition, this particular problem differs from the typical polling system since we consider output queues, which are not typically analyzed in polling systems.

3.1.3. The queue-length distribution of an input queue

Consider input queue i of the sub-network for λ_c in isolation. This input queue receives exactly a_{ic} service slots on wavelength λ_c, as shown in Fig. 4(a). The block of a_{ic} service slots may not be aligned with

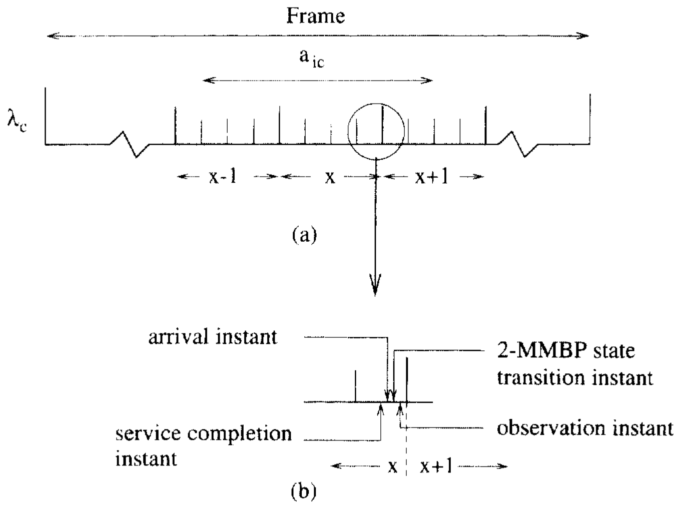


Fig. 4. (a) Service period of input port i on channel λ_c, and (b) detail showing the relationship among service completion, arrival, 2-MMBP state transition, and observation instants within a service and an arrival slot.

the boundaries of the arrival slots. For instance, in the example shown in Fig. 4(a), the block of a_{ic} service slots begins at the second service slot of arrival slot $x-1$, and it ends at the end of the second service slot in arrival slot $x+1$. Here, $x-1$, x, and $x+1$ represent the arrival slot number within a frame.

For each arrival slot, define $v_{ic}(x)$ as the number of service slots allocated to input queue i that lie within arrival slot x.[2] Then, in the example in Fig. 4(a), we have: $v_{ic}(x-1)=3$, $v_{ic}(x)=4$, $v_{ic}(x+1)=2$, and $v_{ic}(x')=0$ for all other x'. Obviously we have

$$\sum_{x=0}^{M-1} v_{ic}(x) = a_{ic}. \tag{4}$$

We analyze input queue i by constructing its underlying Markov chain embedded at arrival slot boundaries. The order of events is as follows. The service (i.e., transmission) completion of a cell occurs at an instant just before the end of a service slot. An arrival may occur at an instant just before the end of an arrival slot, but after the service completion instant of a service slot whose end is aligned with the end of an arrival slot. The 2-MMBP describing the arrival process to the queue makes a state transition immediately after the arrival instant. Finally, the Markov chain is observed at the boundary of each arrival slot, *after* the state transition by the 2-MMBP. The order of these events is shown in Fig. 4(b).

The state of the input queue is described by the tuple (x, y, z), where:

- x represents the arrival slot number within a frame ($x = 0, 1, \ldots, M-1$),
- y indicates the number of cells in the input queue ($y = 0, 1, \ldots, B_{ic}^{(\mathrm{in})}$), and
- z indicates the state of the 2-MMBP describing the arrival process to this queue, i.e., $z = 0, 1$.

[2] In Fig. 4, we assume that each arrival slot contains an integral number of service slots. If this is not the case, $v_{ic}(x)$ is defined as the number of service slots that are concluded within arrival slot x (i.e., if there is a service slot that lies partially within arrival slots x and $x+1$, it will be counted in $v_{ic}(x+1)$).

Table 1
Transition probabilities out of state (x, y, z) of the Markov chain

Current state	Next state	Transition probability
(x, y, z)	$(x \oplus 1, \max\{0, y - v_{ic}(x \oplus 1)\}, z')$	$q_i^{(zz')}(1 - \alpha_i^{(z)} r_{ic})$
(x, y, z)	$(x \oplus 1, \min\{B_{ic}^{(\text{in})}, \max\{0, y - v_{ic}(x \oplus 1)\} + 1\}, z')$	$q_i^{(zz')} \alpha_i^{(z)} r_{ic}$

It is straightforward to verify that, as the state of the queue evolves in time, it defines a Markov chain. Let $\oplus$ denote modulo-M addition, where M is the number of arrival slots per frame. Then the transition probabilities out of state (x, y, z) are given in Table 1. Note that the next state after (x, y, z) always has an arrival slot number equal to $x \oplus 1$. In the first row of Table 1 we assume that the 2-MMBP makes a transition from state z to state z' (from (2), this event has a probability $q_i^{(zz')}$ of occurring), and that no cell arrives to this queue during the current slot (from (2) and (3), this occurs with probability $1 - \alpha_i^{(z)} r_{ic}$). Since at most $v_{ic}(x \oplus 1)$ cells are serviced during arrival slot $x \oplus 1$, and since no cell arrives, the queue length at the end of the slot is equal to $\max\{0, y - v_{ic}(x \oplus 1)\}$. In the second row of Table 1 we assume that the 2-MMBP makes a transition from state z to state z' and a cell arrives to the queue. This arriving cell cannot be serviced during this slot, and has to be added to the queue. Finally, the expression for the new queue length ensures that it will not exceed the capacity $B_{ic}^{(\text{in})}$ of the input queue.

We observe that the probability transition matrix of this Markov chain has the following block form:

$$\mathbf{S}_{ic} = \left[\begin{array}{cccccc} 0 & \mathbf{R}_{ic}(0) & 0 & 0 & \cdots & 0 \\ 0 & 0 & \mathbf{R}_{ic}(1) & 0 & \cdots & 0 \\ 0 & 0 & 0 & \mathbf{R}_{ic}(2) & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \mathbf{R}_{ic}(M-2) \\ \mathbf{R}_{ic}(M-1) & 0 & 0 & 0 & \cdots & 0 \end{array}\right] \begin{array}{l} 0 \\ 1 \\ 2 \\ \vdots \\ M-2 \\ M-1 \end{array} \tag{5}$$

This block form is due to the fact that at each transition instant (i.e., at each arrival slot boundary), the random variable x changes to $x \oplus 1$. Changes in the other two random variables, y and z, of the state of the queue are governed by the matrices $\mathbf{R}_{ic}(x)$. There are M different $\mathbf{R}_{ic}$ matrices, one for each arrival slot x in the frame. Let us define matrices $\mathbf{X}_{ic}$ and $\mathbf{Y}_{ic}$ as follows:

$$\mathbf{X}_{ic} = r_{ic}\, \mathbf{A}_i\, \mathbf{Q}_i \quad \text{and} \quad \mathbf{Y}_{ic} = (\mathbf{I} - r_{ic}\, \mathbf{A}_i)\, \mathbf{Q}_i, \tag{6}$$

where $\mathbf{I}$ is the identity matrix. Then the transition matrix $\mathbf{R}_{ic}(x)$ associated with arrival slot x can be written as

$$\mathbf{R}_{ic}(x) = \left[\begin{array}{ccccccccc} \mathbf{Y}_{ic} & \mathbf{X}_{ic} & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \mathbf{Y}_{ic} & \mathbf{X}_{ic} & 0 & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & \mathbf{Y}_{ic} & \mathbf{X}_{ic} & 0 & 0 & 0 & 0 & \cdots & 0 \\ 0 & 0 & \mathbf{Y}_{ic} & \mathbf{X}_{ic} & 0 & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & \mathbf{Y}_{ic} & \mathbf{X}_{ic} & 0 & \cdots & 0 \end{array}\right] \begin{array}{c} 0 \\ \vdots \\ v_{ic}(x \oplus 1) \\ v_{ic}(x \oplus 1)+1 \\ v_{ic}(x \oplus 1)+2 \\ \vdots \\ B_{ic}^{(\text{in})} \end{array} \tag{7}$$

The structure of matrix $\mathbf{R}_{ic}(x)$ given in (7) can be explained as follows. Suppose that the number of cells y in the queue at the end of slot x is at most $v_{ic}(x \oplus 1)$. Since up to $v_{ic}(x \oplus 1)$ cells can be served within slot $x \oplus 1$, the number in the queue at the end of that slot will be 1 or 0, depending on whether an arrival occurred or not. This is indicated by the transitions in rows 0 through $v_{ic}(x \oplus 1)$ of matrix $\mathbf{R}_{ic}(x)$. However, if at the end of slot x we have $y > v_{ic}(x \oplus 1)$, then the number in the queue at the next transition will be $y - v_{ic}(x \oplus 1)$ (plus one if an arrival occurred). This is indicated by the transitions in rows $v_{ic}(x \oplus 1) + 1$ through B_{ic} of $\mathbf{R}_{ic}(x)$. Of course, y cannot exceed the queue capacity $B_{ic}^{(\text{in})}$. Since the number of service slots $v_{ic}(x \oplus 1)$ depends on the particular slot $x \oplus 1$ within the frame, $\mathbf{R}_{ic}(x)$ is a function of x.

Matrix $\mathbf{R}_{ic}(x)$ is slightly different when $v_{ic}(x \oplus 1) = 0$. This is because, in this case, if the state of the input queue is such that $y = B_{ic}^{(\text{in})}$, a new arrival will be discarded. So when $y = B_{ic}^{(\text{in})}$, the 2-MMBP is allowed to make a transition, but regardless of whether or not an arrival is generated, the number of cells in the queue will remain equal to $B_{ic}^{(\text{in})}$. Thus, the last row of $\mathbf{R}_{ic}(x)$ will be $[0\ 0\ \cdots\ 0\ \mathbf{Q}_i]$.

It is now straightforward to verify that the Markov chain with transition matrix $\mathbf{S}_{ic}$ is irreducible, and therefore a steady-state distribution exists. Transition matrix $\mathbf{S}_{ic}$ defines a *p-cyclic* Markov chain [24], and therefore it can be solved using any of the techniques for *p-cyclic* Markov chains in [24, Ch. 7]. We have used the LU decomposition method in [24] to obtain the steady-state probability $\pi_{ic}(x, y, z)$ that at the end of arrival slot x, the 2-MMBP is in state z and the input queue has y cells. The steady-state probability that the queue has y cells at the end of slot x, independent of the state of the 2-MMBP is

$$\pi_{ic}(x, y) = \sum_{z=0,1} \pi_{ic}(x, y, z). \tag{8}$$

Finally, we note that all the results obtained in this section can be readily extended to MMBP-type arrival processes with more than two states. For this, it would suffice to appropriately modify matrices $\mathbf{X}_{ic}$ and $\mathbf{Y}_{ic}$.

3.2. Output side analysis

We now obtain the queue-length distribution of an output queue. Our analysis follows steps similar to the input side case.

3.2.1. Exact queueing analysis

Let us suppose that the (exact or approximate) queue-length distribution of the input queues is known. Given that transmissions on different channels are independent, and that output queue receivers operate on one wavelength, the output side of the switch may be decomposed into C independent sub-systems, one per wavelength. Let us consider the sub-network corresponding to channel λ_c, and let k_c be the number of output ports sharing this channel. We can then define a $(k_c + 1)$-dimensional Markov chain, where:

- x indicates the arrival slot number within the frame ($x = 0, 1, \ldots, M - 1$), and
- w_n $\forall n = 1, \ldots, k_c$ indicates the number of cells at the nth output queue which shares λ_c ($w_n = 0, 1, \ldots, B_j^{(\text{out})}$).

The transitions out of state $(x, w_1, \ldots, w_{k_c})$ can be computed given the schedule and the queue-length distribution of the input ports. However, for realistic switch dimensions, this method will lead to a state space explosion since the total number of states is of the order of $M \times \prod_{j \in \mathcal{R}_c} B_j^{(\text{out})}$. We now proceed to describe an approximation method that can be used for large systems.

3.2.2. Approximate queueing analysis

Consider the sub-network for wavelength λ_c, and observe that the arrival process to the output queues sharing λ_c is the combination of the departure processes from the input queues corresponding to λ_c. An interesting aspect of the departure process from the input queues is that for each frame, during the sub-period a_{ic} we only have departures from the ith input queue. This period is then followed by a gap g_{ic} during which no departure occurs. This cycle repeats for the next input queue. Thus, in order to characterize the overall departure process offered as the arrival process to these output queues, it suffices to characterize the departure process from each input queue, and then combine them. (We note that this overall departure process is quite different from the typical superposition of a number of departure processes into a single stream, where, at each slot, more than one cell may be departing.)

However, the arrival processes to the output queues listening on λ_c are not independent. Specifically, if j and j' are two output ports on λ_c, and there is a transmission from input port i to output port j in a given service slot, then there can be no arrival to output port j' in the same service slot. As in the input side case, we will nevertheless make the assumption that these arrival processes are indeed independent, and that each is an appropriately thinned (based on the routing probabilities) version of the departure process from the input queues. Note that this is an approximation only when there are multiple output ports with receivers fixed on channel λ_c.

3.2.3. The queue-length distribution of an output queue

As in the previous section, we obtain the queue-length distribution of output port j at arrival slot boundaries. Recall that an arrival slot to an input queue is equal to a departure slot from an output queue. Also, arrival and departure slots are synchronized. Therefore, during an arrival slot x, a cell may be transmitted to the outgoing link from the output queue. However, during slot x, there may be several arrivals to the output queue from the input queues.

Let (x, w) be the state associated with output port j, where:

- x indicates the arrival slot number within the frame ($x = 0, 1, \ldots, M-1$), and
- w indicates the number of cells at the output queue ($w = 0, 1, \ldots, B_j^{(\text{out})}$).

We assume the following order of events. A cell will begin to depart from the output queue at an instant immediately after the beginning of an arrival slot and the departure will be completed just before the end of the slot. A cell from an input port arrives at an instant just before the end of a service slot, but before the end-of-departure instant of an arrival slot whose end is aligned with the end of the service slot. Finally, the state of the queue is observed just before the end of an arrival slot and after the arrival associated with the last service slot has occurred (see Fig. 5(b)).

Let $u_j(x)$ be the number of service slots of any input queue on wavelength λ_c within arrival slot x. We have that

$$u_j(x) = \sum_{i=1}^{N} v_{ic}(x), \tag{9}$$

where $v_{ic}(x)$ is as defined in (4). Quantity $u_j(x)$ represents the maximum number of cells that may arrive to output port j within slot x. In the example of Fig. 5(a) where we show the arrival slots during which cells from input ports i and $i+1$ may arrive to output port j, we have $u_j(x-1) = v_{ic}(x-1) = 4$, $u_j(x) = v_{ic}(x) + v_{i+1,c}(x) = 1 + 2 = 3$, and $u_j(x+1) = v_{i+1,c}(x+1) = 4$.

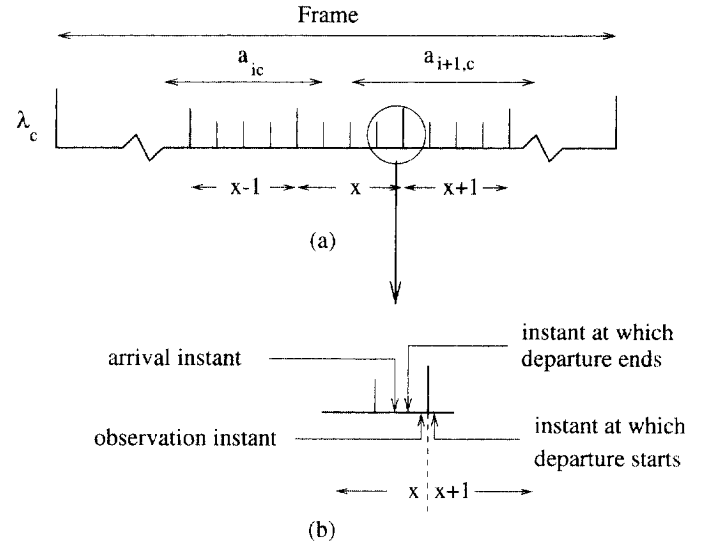


Fig. 5. (a) Arrivals to output port j from input ports i and $i+1$. (b) The detail showing the relationship of departure, arrival, and observation instants.

Table 2
Transition probabilities out of state (x, w) of the Markov chain

Current state	Next state	Transition probability
$(x, 0)$	$(x \oplus 1, \min\{B_j^{(\mathrm{out})}, s\}), 0 \le s \le u_j(x \oplus 1)$	$\sum_{s_1+\cdots+s_N=s} \prod_{i=1}^{N} L_i(s_i \mid x)$
$(x, w), w > 0$	$(x \oplus 1, \min\{B_j^{(\mathrm{out})}, w+s\} - 1), 0 \le s \le u_j(x \oplus 1)$	$\sum_{s_1+\cdots+s_N=s} \prod_{i=1}^{N} L_i(s_i \mid x)$

Observe now that (a) at each state transition x advances by one (modulo-M), (b) exactly one cell departs from the queue as long as the queue is not empty, (c) a number $s \le u_j(x \oplus 1)$ of cells may be transmitted from the input ports to output port j within arrival slot $x \oplus 1$, and that (d) the queue capacity is $B_j^{(\mathrm{out})}$. Then the transition probabilities out of state (x, w) for this Markov chain are given in Table 2.

In Table 2, $L_i(s_i \mid x)$ is the probability that input port i transmits s_i cells to output port j given that the system is at the end of arrival slot x (in other words, it is the probability that s_i cells are transmitted within slot $x \oplus 1$). [3] To obtain $L_i(s_i \mid x)$, define r'_{ij} as the conditional probability that a cell is destined for output port j, given that the cell is destined to be transmitted on λ_c, the receive wavelength of output port j:

$$r'_{ij} = \frac{r_{ij}}{\sum_{k \in \mathcal{R}_c} r_{ik}} = \frac{r_{ij}}{r_{ic}}. \tag{10}$$

This "thinning" of the arrival processes using the r'_{ij} routing probabilities discounts the correlation between arrival streams and is the crux of the approximation for the output side of the switch. The error introduced by this approximation will be discussed later in this work.

[3] Since in most cases only one or two input ports will transmit to the same channel within an arrival slot (refer also to Fig. 2), the summation and product in the expression in the last column of Table 2 do not necessarily run over all N values of i, only over one or two values of i. Thus, this expression can be computed very fast, not in exponential time as implied by the general form presented in the table.

Define $\pi_{ic}(y \mid x)$ as the conditional probability of having y cells at the ith input queue given that the system is observed at the end of slot x:

$$\pi_{ic}(y \mid x) = \frac{\pi_{ic}(x, y)}{\pi_{ic}(x)} = M\,\pi_{ic}(x, y). \tag{11}$$

Then, for $r'_{ij} < 1$, the probability $L_i(s_i \mid x)$ is given by

$$L_i(s_i \mid x) = \begin{cases} \displaystyle\sum_{y=s_i}^{B_{ic}^{(\text{in})}} \pi_{ic}(y \mid x) \binom{\min\{y, v_{ic}(x \oplus 1)\}}{s_i} (r'_{ij})^{s_i} (1 - r'_{ij})^{\min\{y, v_{ic}(x \oplus 1)\} - s_i}, \\ \qquad s_i \le v_{ic}(x \oplus 1), \\ 0, \quad \text{otherwise.} \end{cases} \tag{12}$$

Expression (12) can be explained by noting that input port i will transmit s_i cells to output port j during arrival slot $x \oplus 1$ if (a) $v_{ic}(x \oplus 1) \ge s_i$, (b) input port i has $y \ge s_i$ cells in its queue for λ_c at the beginning of the slot (equivalently, at the end of slot x), and (c) exactly s_i of $\min\{y, v_{ic}(x \oplus 1)\}$ cells that will be transmitted by this queue in this arrival slot are for output j.

If $r'_{ij} = 1$, in which case j is the only port listening on wavelength λ_c, the expression for $L_i(s_i \mid x)$ must be modified as follows:

$$L_i(s_i \mid x) = \begin{cases} \pi_{ic}(s_i \mid x), & 0 \le s_i < v_{ic}(x \oplus 1), \\ \displaystyle\sum_{y=s_i}^{B_{ic}^{(\text{in})}} \pi_{ic}(y \mid x), & s_i = v_{ic}(x \oplus 1), \\ 0, & \text{otherwise.} \end{cases} \tag{13}$$

Expressions (12) and (13) are based on the assumption that $v_{ic}(x \oplus 1) < B_{ic}^{(\text{in})}$ which we believe is a reasonable one. In the general case, quantity $v_{ic}(x \oplus 1)$ in both expressions must be replaced by $\min\{v_{ic}(x \oplus 1), B_{ic}^{(\text{in})}\}$.

The transition matrix $\mathbf{T}_j$ of the Markov chain defined by the evolution of the state (x, w) of output queue j has the following form, which is similar to that of matrix $\mathbf{S}_{ic}$ given by (5):

$$\mathbf{T}_j = \left[\begin{array}{cccccc} 0 & \mathbf{U}_j(0) & 0 & 0 & \cdots & 0 \\ 0 & 0 & \mathbf{U}_j(1) & 0 & \cdots & 0 \\ 0 & 0 & 0 & \mathbf{U}_j(2) & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \cdots & \mathbf{U}_j(M-2) \\ \mathbf{U}_j(M-1) & 0 & 0 & 0 & \cdots & 0 \end{array}\right] \begin{array}{c} 0 \\ 1 \\ 2 \\ \vdots \\ M-2 \\ M-1 \end{array} \tag{14}$$

$\mathbf{U}_j(x)$ is a $(B_j^{(\text{out})} + 1) \times (B_j^{(\text{out})} + 1)$ matrix that governs changes in random variable w of the state of the output queue. The elements of this matrix can be determined using Table 2 and expressions (12) or (13). Since $L_i(s_i \mid x)$ depends on $v_{ic}(x)$ and $u_j(x)$, $\mathbf{U}_j(x)$ also depends on x, the slot number within the frame.

We observe that $\mathbf{T}_j$ also defines a *p-cyclic* Markov chain. We have used the LU decomposition method as prescribed in [24] to obtain $\pi_j(x, w)$, the steady-state probability that output queue j has w cells at the end of slot x.

3.3. Summary of the decomposition algorithm

Below we summarize our approach to analyzing the sub-network of Fig. 3 corresponding to wavelength λ_c. We assume that quantities $\{a_{ic}\}$ and the corresponding schedule (see [10]) are given.

1. For each arrival slot x, use the schedule and expressions (4) and (9) to compute the quantities $v_{ic}(x)$ and $u_j(x)$, $i = 1, \ldots, N$, $j{:}\, \lambda(j) = \lambda_c$.
2. For each input queue i, construct the transition probability matrix $\mathbf{S}_{ic}$ from (2), (3), (5), (6), and (7). Solve this matrix and use (8) to obtain the steady-state probability $\pi_{ic}(x, y)$ that input queue i has y cells at the end of the xth slot of the frame.
3. For each output port $j \in \mathcal{R}_c$, use $\pi_{ic}(x, y)$ derived in Step 2, and (12) and (13) to construct the transition matrix $\mathbf{T}_j$ given by (14). Solve the matrix as in Step 2 to obtain $\pi_j(x, w)$, the steady-state probability that port j has w cells in its queue at the end of slot x.

Note that the complexity of this approach is dominated by Step 2. For each of the N input queues we have to solve a matrix of dimensions $[2M(B_{ic}^{(\text{in})} + 1)] \times [2M(B_{ic}^{(\text{in})} + 1)]$, where M is the length of the schedule (in arrival slots) and $B_{ic}^{(\text{in})}$ is the capacity of the respective queue. (Inverting a $K \times K$ matrix takes time $O(K^3)$, although we can take advantage of the fact that the matrix is sparse to solve for the queue-length distributions at a significantly faster rate.) Thus, in the worst case, the overall complexity of our algorithm is $O(NM^3B^3)$, where $B = \max_i\{B_{ic}^{(\text{in})}\}$.

4. Cell-loss probability

We now use the queue-length distributions for the input and output ports, $\pi_{ic}(x, y)$ and $\pi_j(x, w)$, respectively, derived in the previous section, to obtain the cell-loss probability at the input and output ports.

4.1. The cell-loss probability at an input port

Let Ω_{ic} be the cell-loss probability at the cth queue of input port i, i.e., the probability that a cell arriving to that queue will be lost. Ω_{ic} can be expressed as

$$\Omega_{ic} = \frac{E[\text{number of cells lost per frame at queue } c \text{ of port } i]}{E[\text{number of arrivals per frame at queue } c \text{ of port } i]}. \tag{15}$$

Obtaining the expectation in the denominator is easy. From (2) and [21], the steady-state arrival probability for the arrival process to this queue is

$$\gamma_i = \frac{q_i^{(10)}\alpha_i^{(0)} + q_i^{(01)}\alpha_i^{(1)}}{q_i^{(01)} + q_i^{(10)}}. \tag{16}$$

Then

$$E[\text{number of arrivals per frame at queue } c \text{ of port } i] = M\,\gamma_i\,r_{ic}. \tag{17}$$

To obtain the expectation in the numerator, let us refer to Fig. 4(b) which shows the service completion, arrival, and observation instants within slot x. We observe that, due to the fact that at most one cell may arrive in slot x, if the number $v_{ic}(x)$ of slots during which this queue is serviced within arrival slot x is not zero (i.e., $v_{ic}(x) > 0$), no arriving cell will be lost. Even if the cth queue at input port i is full at the beginning of slot x, $v_{ic}(x) \geq 1$ cells will be serviced during this slot, and the order of service completion and arrival instants in Fig. 4(b) guarantees that an arriving cell will be accepted. On the other hand, if $v_{ic}(x) = 0$ for slot x, then an arriving cell will be discarded if and only if the queue is full at the beginning of x (equivalently, at the end of the slot before x). Since the 2-MMBP can be in one of two states, we have that

$$\begin{aligned} &E[\text{number of cells lost per frame at queue } c \text{ of port } i] \\ &\quad = \sum_{x:v_{ic}(x)=0} \sum_{z=0}^{1} \alpha_i^{(z)} r_{ic} \pi_{ic}(B_{ic}, z \mid x \ominus 1). \end{aligned} \tag{18}$$

In (18), $\ominus$ denotes regular subtraction with the exception that, if $x = 0$, then $x \ominus 1 = M - 1$, and the summation runs over all x for which $v_{ic}(x) = 0$. Using (15), (17) and (18), and the fact that $\pi_{ic}(x) = 1/M$ for all x, we obtain an expression for Ω_{ic} as follows:

$$\Omega_{ic} = \frac{\sum_{x:v_{ic}(x)=0} \sum_{z=0}^{1} \alpha_i^{(z)} \pi_{ic}(x \ominus 1, B_{ic}^{(\text{in})}, z)}{\gamma_i}. \tag{19}$$

4.2. The cell-loss probability at an output port

The cell-loss probability at an output port is more complicated to calculate, since we may have multiple cell arrivals to the given output port within a single arrival slot (refer to Fig. 5(a)). Let us define $\Omega_j(n \mid x)$ as the conditional probability that n cells will be lost at output queue j given that the current arrival slot is x. An output port will lose n cells in slot x if (a) the port had w, $0 \leq w \leq B_j^{(\text{out})}$, cells at the beginning of slot x, and (b) exactly $B_j^{(\text{out})} - w + n$ cells arrived during slot x. We can then write

$$\Omega_j(n \mid x) = \sum_{w=0}^{B_j^{(\text{out})}} \pi_j(w \mid x \ominus 1) \Pr[B_j^{(\text{out})} - w + n \text{ cells arrive to } j \mid x], \tag{20}$$

where $\pi_j(w \mid x \ominus 1) = M\pi(x \ominus 1, w)$ similar to (11). The last probability in (20) can be obtained using (12) or (13), as in Table 2:

$$\Pr[s \text{ cells arrive to output port } j \mid x] = \sum_{s_1+\cdots+s_N=s} \prod_{i=1}^{N} L_i(s_i \mid x \ominus 1). \tag{21}$$

Note that at most $u_j(x)$ cells may arrive (and get lost) in arrival slot x. Using (20), we can then compute the expected number of cells lost in slot x as

$$E[\text{number of cells lost at } j \mid x] = \sum_{n=1}^{u_j(x)} n\Omega_j(n \mid x). \tag{22}$$

The expected number of arrivals to port j in slot x can be computed using (21):

$$E[\text{number of arrivals to } j \mid x] = \sum_{s=1}^{u_j(x)} s \Pr[s \text{ cells arrive to } j \mid x]. \tag{23}$$

Finally, the probability Ω_j that an arriving cell to output port j will be lost regardless of the arrival slot x can be found as follows:

$$\Omega_j = \frac{E[\text{number of lost cells in a frame}]}{E[\text{number of arrivals in a frame}]} = \frac{\sum_{x=0}^{M-1} E[\text{number of lost cells at } j \mid x]}{\sum_{x=0}^{M-1} E[\text{number of arrivals to } j \mid x]}. \tag{24}$$

5. Numerical results

We now apply our analysis to a switch with $N = 16$ ports. The arrival process to each of the ports of the switch is described by a different 2-MMBP. In Fig. 6, we plot two important parameters of each of the 16 2-MMBPs we have used: the mean interarrival time in slots (γ_i^{-1} in (16)), and the squared coefficient of variation of the interarrival time given in [21]. As can be seen, the arrival processes exhibit a wide range of behavior in terms of these two parameters. The routing probabilities we used are

$$r_{ij} = \begin{cases} 0.10, & i = 1, \ldots, 16, \quad j = 1, \\ 0.06, & i = 1, \ldots, 16, \quad j = 2, \ldots, 16. \end{cases} \tag{25}$$

That is, output port 1 is a hot spot, receiving 10% of the total traffic, while the remaining traffic is evenly distributed to the other 15 ports. The total rate at which cells are generated by users of the network is 1.98 cells per arrival slot. Most of the traffic is generated at port 1, as the rate of new cells generated at this port is 0.583 cells per arrival slot. The cell generation rate decreases monotonically for ports 2–16. For load balancing purposes, we have allocated one of the C channels exclusively to port 1, since this port receives a considerable fraction of the total traffic. The remaining $C - 1$ channels are shared by the other 15 output

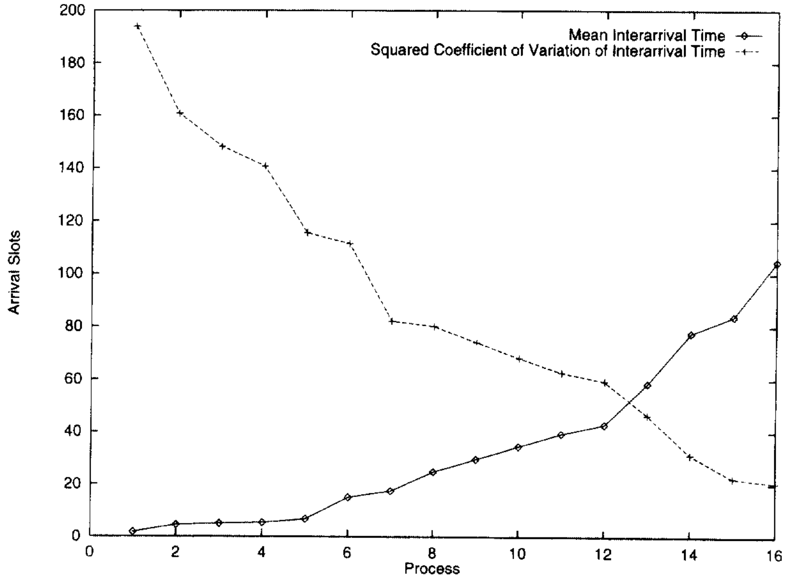


Fig. 6. Mean arrival rate and squared coefficient of variation of the interarrival time for the arrival processes to the 16 input ports of the switch.

Table 3
Channel sharing for $C = 4, 6, 8$

	$C=4$	$C=6$	$C=8$
$\mathcal{R}_1$	$\{1\}$	$\{1\}$	$\{1\}$
$\mathcal{R}_2$	$\{2, 5, 8, 11, 14\}$	$\{2, 7, 12\}$	$\{2, 9, 16\}$
$\mathcal{R}_3$	$\{3, 6, 9, 12, 15\}$	$\{3, 8, 13\}$	$\{3, 10\}$
$\mathcal{R}_4$	$\{4, 7, 10, 13, 16\}$	$\{4, 9, 14\}$	$\{4, 11\}$
$\mathcal{R}_5$		$\{5, 10, 15\}$	$\{5, 12\}$
$\mathcal{R}_6$		$\{6, 11, 16\}$	$\{6, 13\}$
$\mathcal{R}_7$			$\{7, 14\}$
$\mathcal{R}_8$			$\{8, 15\}$

ports. The allocation of the output ports to the remaining wavelengths was performed in a round-robin fashion, and is given in Table 3 for $C = 4, 6, 8$.

The quantities a_{ic} of the schedule, i.e., the number of cells to be transmitted by port i onto channel λ_c per frame (refer to Section 2.2 and Fig. 2) were fixed to be as close to (but no less than) 0.5 arrival slots as possible. Recall that, while the length of an arrival slot is independent of C and is taken as our unit of time, the length of a service slot depends on the number of channels. In cases in which 0.5 arrival slots is not an integral number of service slots, the value a_{ic} is rounded up to the next integer to ensure that every queue is granted at least 0.5 arrival slots of service during each frame [4] (i.e., $a_{ic} = \lceil N/2C \rceil \forall i, c$). In constructing the schedules, we have assumed that the time it takes a laser to tune from one channel to another is equal to one arrival slot. [5] Finally, for all the results we present in this section we have let all input and output queues have the same buffer capacity B (i.e., $B_{ic}^{(\text{in})} = B_j^{(\text{out})} = B$) to reduce the number of parameters that need to be controlled.

In Fig. 7 we show the part of the schedule corresponding to channel λ_1 for three different values of the number of channels $C = 4, 6$, and 8; the parts of the schedules for other channels are very similar. The schedules will help explain the performance results to be presented shortly. Since the number of ports $N = 16$, for $C = 4$ each arrival slot is exactly four service slots long. Each input port is allocated 0.5 arrival slots, or 2 service slots for transmissions on each channel, as Fig. 7(a), illustrates. For $C = 4$ the switch is *bandwidth limited* [10], i.e., the length of the schedule is determined by the bandwidth requirements on each channel ($= 16 \times 0.5 = 8$ arrival slots), not the transmission and tuning requirements of each input port ($= 4 \times 0.5 + 4 \times 1 = 6$ arrival slots). The schedule for $C = 6$ in Fig. 7(b) is an example where there is a non-integral number of service slots within each arrival slot. More precisely, one arrival slot contains $N/C = \frac{16}{6}$, or $2\frac{2}{3}$ service slots. Each input port is assigned two service slots ($a_{ic} = 2$) for transmissions on each channel, since one service slot is less than 0.5 arrival slots. For $C = 6$, the switch is again bandwidth limited, and the total schedule length becomes $16 \times 2 = 32$ service slots, or 12 arrival slots.

[4] Other schemes for allocating a_{ic} have been implemented, including setting a_{ic} proportional to r_{ic}, setting a_{ic} proportional to $\max_z\{\alpha_{ic}^{(z)}\}$, and setting a_{ic} to the effective bandwidth [20] of port i's total traffic carried on channel λ_c. Although the cell-loss probability results do depend on the actual values of a_{ic}, the overall conclusions drawn regarding our analysis are very similar. Thus, we have decided to include only the simplest case here.

[5] Again, due to the synchronous nature of this switch, if one arrival slot is not an integral number of service slots, the number of service slots for which a transmitter cannot transmit is rounded up to the next integer, thereby setting the required time for tuning to some value slightly greater than one arrival slot. As a result, the tuning time is always $\lceil N/C \rceil$ service slots.

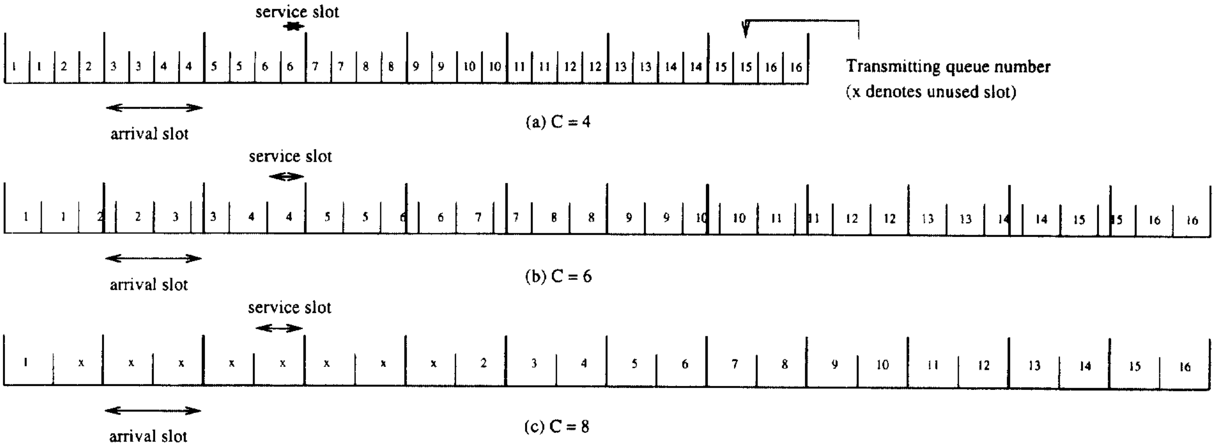


Fig. 7. Transmission schedules for λ_1 and $C = 4, 6, 8$ (the unit of time is fixed across the schedules and is equal to an arrival slot).

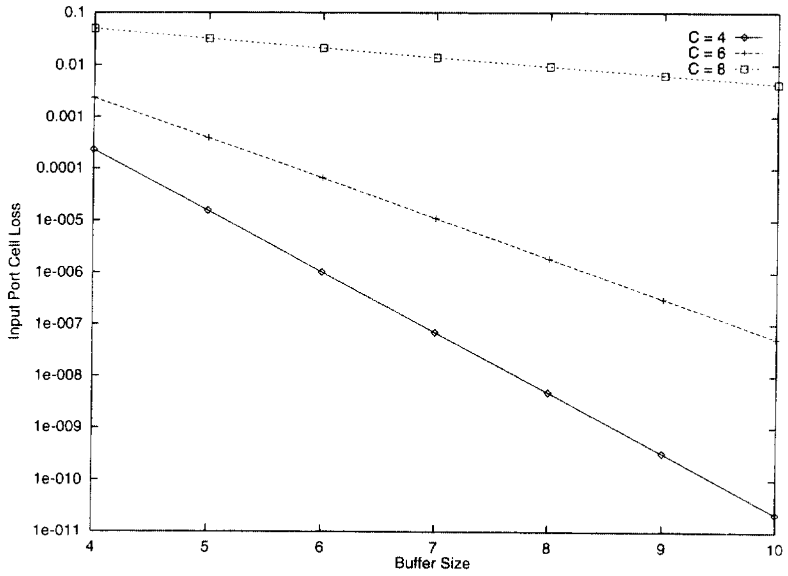


Fig. 8. Input queue cell-loss probability $\Omega_{1,1}$ for $C = 4, 6, 8$ as a function of buffer size.

Finally, when $C = 8$, $a_{ic} = 1$ service slot = 0.5 arrival slots, and the corresponding schedule is shown in Fig. 7(c). However, in this case the switch is *tuning limited* [10], i.e., the port transmission and tuning requirements determine the schedule length. Since each input port has to transmit for 0.5 arrival slots on each channel, and to tune to each of the 8 channels (recall that the tuning time is one arrival slot), the total schedule length is $8 \times 0.5 + 8 \times 1 = 12$ arrival slots. But the transmissions on each channel only take $16 \times 0.5 = 8$ arrival slots; the remaining 4 arrival slots in Fig. 7(c) are not used.

Figs. 8–11 show the cell-loss probability (CLP) at four different input queues as a function of the buffer size B for $C = 4, 6, 8$. We only show results for two input ports, namely, the port with the highest traffic intensity (port 1) in Figs. 8 and 10, and a representative intermediate port (port 8) in Figs. 9 and 11. We also

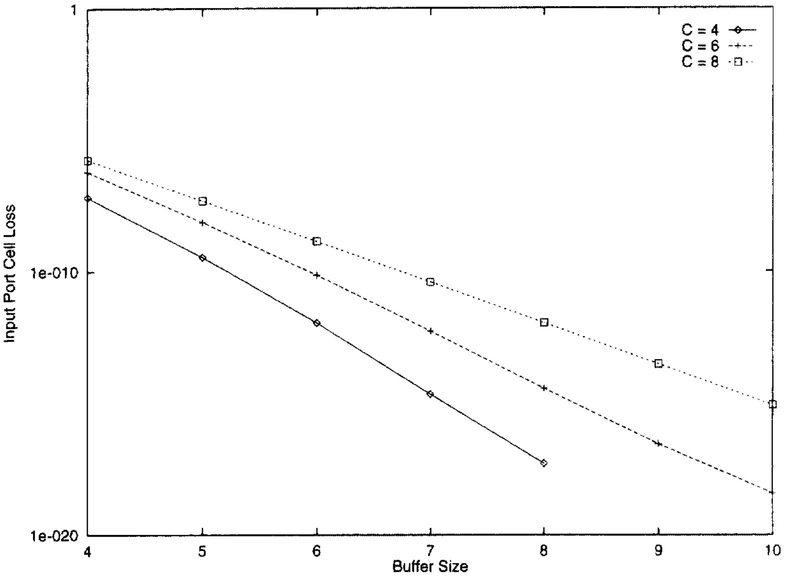


Fig. 9. Input queue cell-loss probability $\Omega_{8,1}$ for $C = 4, 6, 8$ as a function of buffer size.

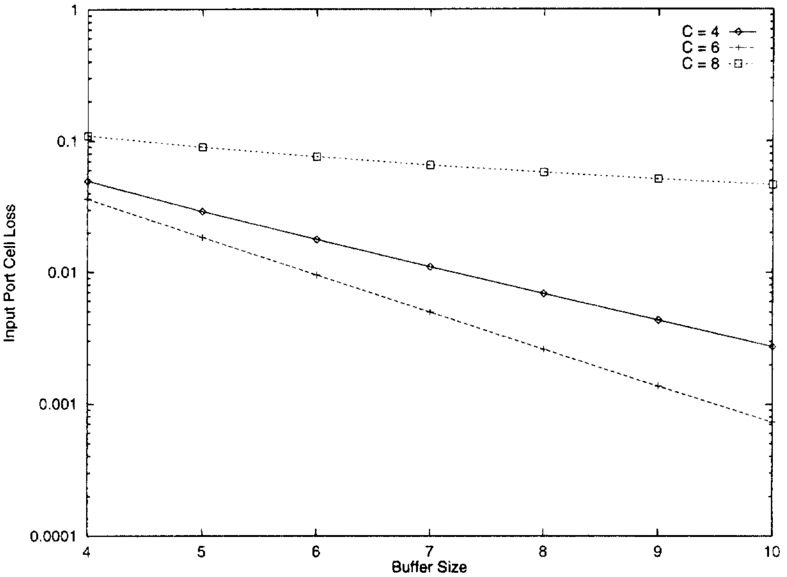


Fig. 10. Input queue cell-loss probability $\Omega_{1,2}$ for $C = 4, 6, 8$ as a function of buffer size.

consider only input queues 1 and 2 (out of C) at each port. Queue 1 at each port is for traffic to be carried on wavelength λ_1, which is dedicated to output port 1 (the "hot spot"). Thus, the amount of traffic received by this queue *does not change as we vary the number of channels*, since the first channel is dedicated to output port 1. Queue 2 at each port is for traffic to be carried on wavelength λ_2. The amount of traffic received by this queue will decrease as the number of channels increases, since channel λ_2 will be shared by fewer

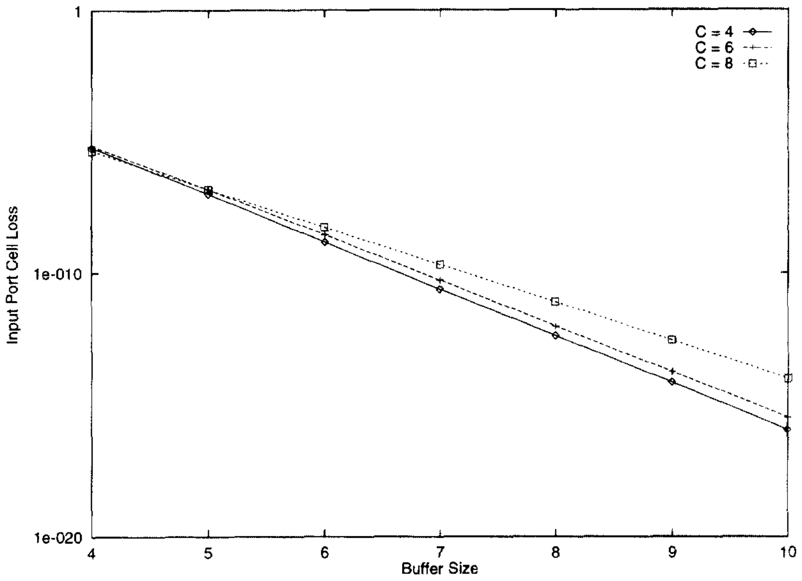


Fig. 11. Input queue cell-loss probability $\Omega_{8,2}$ for $C = 4, 6, 8$ as a function of buffer size.

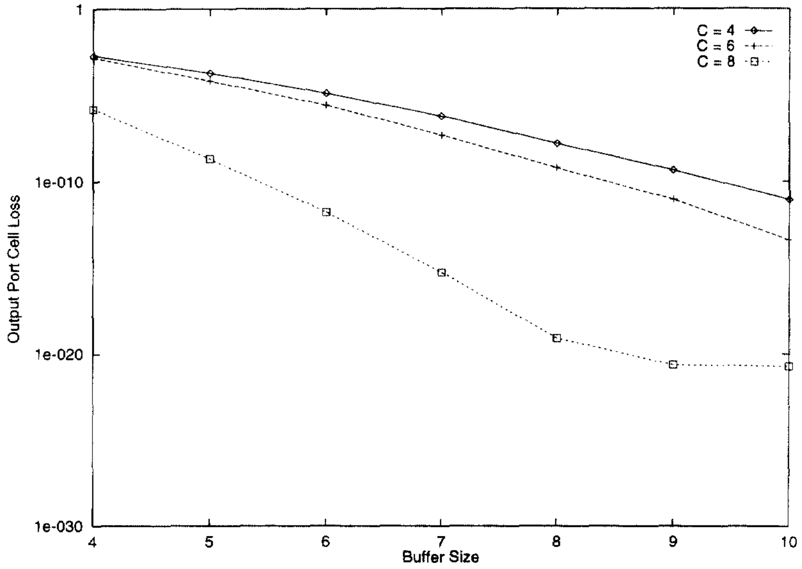


Fig. 12. Output queue cell-loss probability Ω_1 for $C = 4, 6, 8$ as a function of buffer size.

output ports. The behavior of queue 2 is representative of the behavior of the other $C - 2$ queues, 3 through C.

Fig. 8 plots the CLP $\Omega_{1,1}$ (i.e., the CLP at input queue 1 of port 1) as a function of the buffer size B for $C = 4, 6, 8$. As expected, the CLP decreases as the buffer size increases. For a given buffer size, however, the CLP changes dramatically and counter to intuition, as the number C of channels is varied. Specifically, the CLP increases with C; i.e., adding more channels results in worse performance. When B is 10, there

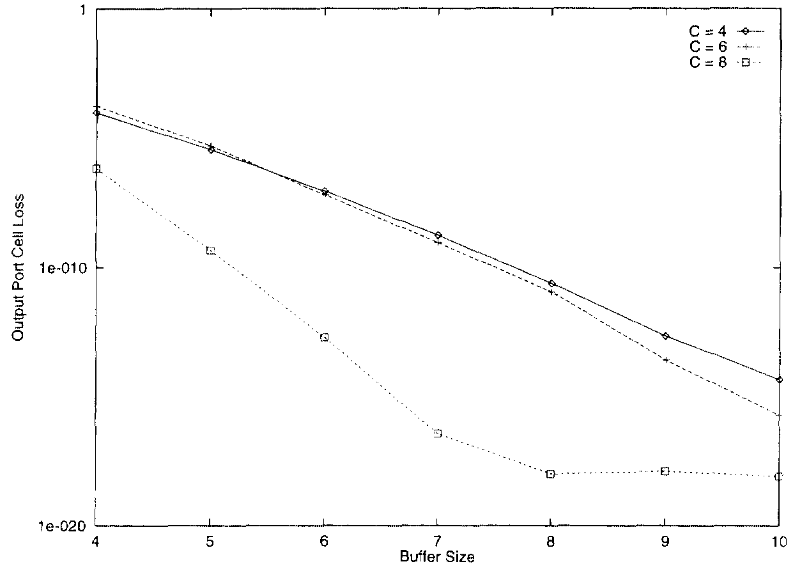


Fig. 13. Output queue cell-loss probability Ω_8 for $C = 4, 6, 8$ as a function of buffer size.

is roughly nine orders of magnitude difference between the CLP for $C = 4$ and $C = 8$, and three orders of magnitude difference between $C = 4$ and $C = 6$. As we discussed above, the traffic load of this queue does not change with C; the queue receives the traffic for destination 1, which is always 10% of the total traffic generated at port 1 (see (25)). What does change as C varies is the service rate of the queue, and this change can help explain the results in Fig. 8. Referring to Fig. 7, we note that when $C = 4$, each frame of the schedule is $M = 8$ arrival slots long, and $a_{1,1} = 2$. Hence, at most 8 cells may arrive to this queue during a frame while as many as 2 cells will be serviced. When $C = 6$, $M = 12$ and $a_{1,1} = 2$, indicating a decrease in the service rate of the queue. Similarly, for $C = 8$, $M = 12$ and $a_{1,1} = 1$, a further decrease in available service per frame for this queue. This decrease is the reason behind the sharp increase in CLP with C in Fig. 8. Very similar behavior is observed in Fig. 9 where we plot $\Omega_{8,1}$, the CLP at input queue 1 of port 8. The main difference between Figs. 8 and 9 is in the absolute values of CLP. The very small CLP numbers for $\Omega_{8,1}$ are due to the fact that the amount of traffic entering queue 1 of port 8 (0.004 cells per arrival slot) is significantly smaller than the traffic entering the same queue of port 1 (0.058 cells per arrival slot – recall that the traffic sources were chosen so that the cell generation rate decreases as the port index increases). In fact, for buffer sizes $B = 9$ and $B = 10$ and $C = 4$ our analysis gave CLP values that are essentially zero; these values are not plotted in Fig. 9 because we believe that they are the result of round-off errors.

Figs. 10 and 11 plot the CLP at input queue 2 of ports 1 and 8, respectively, against the buffer size. From (25) and Table 3 we note that the traffic received by this queue decreases from 30% of the overall switch traffic when $C = 4$ to 18% when $C = 6$ or 8; this decrease is due to the fact that 5 output ports share wavelength λ_2 when $C = 4$, but only 3 output ports share it when $C = 6$ or 8. Thus, the CLP behavior at this queue will depend not only on the change in the service rate as C varies, but also on the change in the amount of traffic received due to the addition of new channels. In Fig. 10, and for a given buffer size, the CLP decreases as C increases from 4 to 6 (compare to Fig. 8). In this case, the decrease in the traffic arrival rate (from an average rate of 0.175 to 0.105 cells per arrival slot) more than offsets the decrease in the service rate that we discussed above. On the other hand, the CLP values for $C = 6$ are higher than those for $C = 4$ in Fig. 11 (input queue 2 of port 8) due to the fact that the decrease in the offered load (from 0.012 to 0.007 cells per arrival slot) is not substantial enough to offset the decrease in the service rate; still, this increase is less severe than the one in Fig. 9 where there was no decrease in the arrival rate. As C increases to 8 there is no change in the offered traffic for either queue; as expected, the CLP rises with the decrease in the service rate.

Finally, Figs. 12 and 13 plot the CLP at output queues 1 and 8, respectively. Output queue 8 is representative of queues 2–16 in that it receives 6% of the total switch traffic (see (25)). Again, the CLP decreases with increasing buffer size. Also, the lower values of CLP in Fig. 13 compared to Fig. 12 reflect the fact that only 6% of the total traffic is destined to output queue 8, as opposed to 10% for the hot spot queue 1. What is surprising in Figs. 12 and 13, however, is that, for a given buffer size, the CLP decreases as the number C of channels increases. This behavior is in sharp contrast to the one we observed in the input side case, and can be explained as follows. First, higher losses at the input queues for larger values of C means that fewer cells will make it to the output queues, thus losses will be lower at the latter. But the dominant factor in the CLP behavior in Figs. 12 and 13 is the change in the service rate of the output queues as C varies (refer to Fig. 7). For $C = 4$, as many as 32 cells may arrive to each output queue within a frame, and 8 cells may be served. When $C = 6$ the number of potential arrivals in a frame remains at 32, but the frame is 12 arrival slots long, meaning that up to 12 cells may be served, leading to a drop in the CLP. Finally, for $C = 8$ the number of cells served in a frame is the same as in $C = 6$, but the maximum number of cells that may arrive becomes only 16, explaining the dramatic drop in the CLP.

In order to validate the accuracy of the approximation algorithm, we ran several experiments involving different switch configurations. The switch sizes varied from two ports and one wavelength to 16 ports and 10 wavelengths. We observed that the smallest relative error [6] for cell-loss occurred at an output port which was allocated to a dedicated wavelength. The relative error observed in this case was approximately 1×10^{-3}. The relative error increased as the number of ports sharing a single wavelength increased. The worst relative error observed was 5×10^{-2}. We observed a similar behavior for the cell-loss at the input ports.

6. Concluding remarks

In this paper we introduced a model for a photonic single-hop ATM switch architecture. The model consists of a queueing network of input and output queues, and a schedule that masks the transceiver tuning latency. We developed a decomposition algorithm to obtain the queue-length distributions at the input and output queues of the switch. We also obtained analytic expressions for the cell-loss probability at the various queues. Finally, we presented a study case to illustrate the significance of our work in predicting and explaining the performance of the switch in terms of the cell-loss probability.

Overall, the results presented in this paper indicate that the performance of an optical WDM switch can exhibit behavior that is counter to intuition, and which may not be predictable without an accurate analysis. The performance curves shown also establish that the cell-loss probability in such an environment depends strongly on the interaction among the scheduling and load balancing algorithms, the routing probabilities,

[6] Relative error is taken here to mean the ratio of the absolute difference between the mean simulated value and the value calculated by our algorithm divided by the algorithm's value.

and the number of available channels. Our work has made it possible to investigate the behavior of optical ATM switches under more realistic assumptions regarding the traffic sources and the system parameters (e.g., finite buffer capacities) than was possible before, and it represents a first step towards a more thorough understanding of ATM switch performance in a WDM environment. Our analysis also suggests that simple slot allocation schemes similar to the ones used for our study case are not successful in utilizing the additional capacity provided by an increase in the number of channels. The specification and evaluation of more efficient slot allocation schemes should be explored in future research.

References

[1] F.A. Tobagi, Fast packet switch architectures for broadband integrated services digital networks, Proc. IEEE. (January 1990) 133–167.

[2] H. Ahmadi, W.E. Denzel, A survey of modern high-performance switching techniques, IEEE J. Selected Areas Comm. (September 1989) 1227–1237.

[3] P.E. Green, Fiber Optic Networks, Prentice-Hall, Englewood Cliffs, NJ, 1993.

[4] A. Sneh, K.M. Johnson, High-speed tunable liquid crystal optical filter for WDM systems, in: Proc. IEEE/LEOS Summer Topical Meetings on Optical Networks and their Enabling Technologies, Lake Tahoe, CA, July 1994, pp. 59–60.

[5] K. Nakagawa, S. Nishi, K. Aida, E. Yoneda, Trunk and distribution network application of erbium-doped fiber amplifier, J. Lightwave Technol. LT-9 (1991).

[6] L. Thylen, G. Karlsson, O. Nilsson, Switching technologies for future guided wave optical networks: Potentials and limitations of photonics and electronics, IEEE Comm. Magazine 34 (2) (1996) 106–113.

[7] A. Jaszczyk, H.T. Mouftah, Photonic fast packet switching, IEEE Comm. Magazine (1993) 58–65.

[8] G.N. Rouskas, M.H. Ammar, Analysis and optimization of transmission schedules for single-hop WDM networks, IEEE/ACM Trans. Networking 3 (2) (1995) 211–221.

[9] V. Sivaraman, G.N. Rouskas, HiPeR-ℓ: A *H*igh Performance Reservation protocol with ℓook-ahead for broadcast WDM networks, in: Proc. INFOCOM '97, IEEE Press, New York, 1997, pp. 1272–1279.

[10] G.N. Rouskas, V. Sivaraman, Packet scheduling in broadcast WDM networks with arbitrary transceiver tuning latencies, IEEE/ACM Trans. Networking 5 (3) (1997) 359–370.

[11] D.A. Levine, I.F. Akyildiz, PROTON: A media access control protocol for optical networks with star topology, IEEE/ACM Trans. Networking 3 (2) (1995) 158–168.

[12] T.T. Lee, M.S. Goodman, E. Arthurs, A broadband optical multicast switch, in: Proc. ISS '90, 1990.

[13] P.A. Humblet, R. Ramaswami, K.N. Sivarajan, An efficient communication protocol for high-speed packet-switched multichannel networks, IEEE J. Selected Areas Comm. 11 (4) (1993) 568–578.

[14] I.M.I. Habbab, M. Kavehrad, C.-E.W. Sundberg, Protocols for very high-speed optical fiber local area networks using a passive star topology, J. Lightwave Technol. LT-5 (12) (1987) 1782–1793.

[15] J. Jue, M. Borella, B. Mukherjee, Performance analysis of the Rainbow WDM optical network prototype, IEEE J. Selected Areas in Comm. 14 (5) (1996) 945–951.

[16] V. Paxson, S. Floyd, Wide area traffic: The failure of poisson modeling, IEEE/ACM Trans. Networking 3 (3) (1995) 226–244.

[17] G. Pujolle, H.G. Perros, Queueing systems for modelling ATM networks, in: Int. Conf. on the Performance of Distributed Systems and Integrated Communication Networks, Kyoto, Japan, September 1991, pp. 10–12.

[18] B. Mukherjee, WDM-based local lightwave networks Part I: Single-hop systems, IEEE Network Magazine (1992) 12–27.

[19] I. Baldine, G.N. Rouskas, Dynamic load balancing in broadcast WDM networks with tuning latencies, in: Proc. INFOCOM '98, to appear.

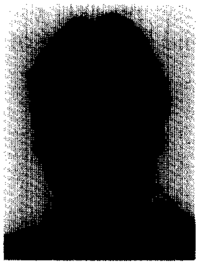
[20] H.G. Perros, K.M. Elsayed, Call admission control schemes: A review, IEEE Comm. Magazine 34 (11) (1996) 82–91.

[21] D. Park, H.G. Perros, H. Yamashita, Approximate analysis of discrete-time tandem queueing networks with bursty and correleated input traffic and customer loss, Oper. Res. Lett. 15 (1994) 95–104.

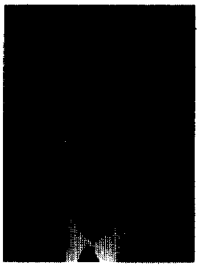
[22] M.W. McKinnon, Performance analysis of photonic ATM switch architectures, Ph.D. Thesis, North Carolina State University, Raleigh, NC, December 1997.

[23] A.O. Zaghloul, H.G. Perros, Approximate analysis of a discrete-time polling system with bursty arrivals, in: Perros, Pujolle, Takahashi (Eds.), Modelling and Performance Evaluation of ATM Technology, 1993.

[24] W. Stewart, Numerical Solutions of Markov Chains, Princeton University Press, Princeton, NJ, 1994.

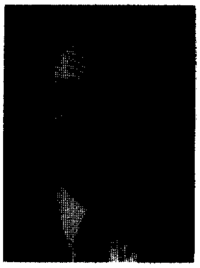


Martin W. McKinnon received his doctoral degree from North Carolina State University in Computer Engineering in 1997, with his research focusing on the design and performance analysis of photonic device architectures. He has been employed by The MITRE Corporation, Nortel (formerly Bell Northern Research), and other telecommunication-related organizations. He is currently with the Communication and Networking Division of the Georgia Tech Research Institute. His current research is in photonic architectures, active networking, and queueing theory. He is a member of IEEE, ACM and ORSA.



George N. Rouskas received the Diploma in Electrical Engineering from the National Technical University of Athens (NTUA), Athens, Greece, in 1989, and the M.S. and Ph.D. degrees in Computer Science from the College of Computing, Georgia Institute of Technology, Atlanta, GA, in 1991 and 1994, respectively. He joined the Department of Computer Science, North Carolina State University, in August 1994, as an Assistant Professor. His research interests include high-speed and lightwave network architectures, multipoint-to-multipoint communication, and performance evaluation.

He is a recipient of a 1997 NSF Faculty Early Career Development (CAREER) Award. He also received the *1995 Outstanding New Teacher Award* from the Department of Computer Science, North Carolina State University and the *1994 Graduate Research Assistant Award* from the College of Computing, Georgia Tech. He is a member of the IEEE, the ACM and of the Technical Chamber of Greece.



Harry G. Perros received the B.Sc. degree in Mathematics in 1970 from Athens University, Greece, the M.Sc. degree in Operational Research with Computing from Leeds University, UK, in 1971, and the Ph.D. degree in Operations Research from Trinity College, Dublin, Ireland, in 1975. From 1976 to 1982 he was a Assistant Professor in the Department of Quantitative Methods, University of Illinois at Chicago. In 1979 he spent a sabbatical term at INRIA, Rocquencourt, France, In 1982 he joined the Department of Computer Science, North Carolina State University, as an Associate Professor, and since 1988 he is a Professor. During the academic year 1988–89 he was on a sabbatical leave of absence first at BNR, Research Triangle Park, North Carolina, and subsequently at the University of Paris 6, France. Also, during the academic year 1995–96 he was on a sabbatical leave of absence at NORTEL, Research Triangle Park, North Carolina.

He has published extensively in the area of performance modeling of computer and communication systems, and has organized several national and international conferences. He also published a monograph entitled, "Queueing networks with blocking: exact and approximate solutions", Oxford Press. He is the chairman of the IFIP W.G. 6.3 on the Performance of Communication Systems. His current research interests are in the areas of optical networks and their performance, and software performance evaluation.