ELSEVIER

Performance Evaluation 19 (1994) 317–340

PERFORMANCE EVALUATION
An International Journal

Analysis of a finite buffer queue with different scheduling and push-out schemes

Ian F. Akyildiz [a,*], Xian Cheng [b]

[a] School of Electrical and Computer Engineering, Georgia Institute of Technology Atlanta, GA 30332, USA
[b] AT&T Bell Labs, Room 3C-508A, 101 Crawfords Corner Road, Holmdel, NJ 07733, USA

(Received 1 July 1992; revised 16 February 1993)

Abstract

A queueing system M_1, M_2/G_1, $G_2/1/N$ with different scheduling and push-out scheme is analyzed in this paper. This work is motivated by the study of the performance of an output link of ATM switches with traffic of two classes with different priorities. However, the queueing model developed in this paper is more general than that of the output link of ATM switches with two-class priority traffic. General service time distributions are allowed for classes 1 and 2 and a general service discipline function, $\alpha_1(i, j)$, is introduced where $\alpha_1(i, j)$ is the probability that a class 1 packet will be served, given that there are i class 1 and j class 2 packets waiting for service. An exact solution is obtained for the loss probabilities for classes 1 and 2, the queue length distribution and the mean waiting time for class 1. The queue length distribution and the mean waiting time for class 2 are calculated approximately. It is shown that the approximation is an upper bound and the error due to the approximation is very small when the loss probability of class 2 is small (e.g., less than 0.01).

Key words: High speed networks; ATM networks; Performance evaluation; Finite buffer queue; Priorities

1. Introduction

We analyze an M_1, M_2/G_1, $G_2/1/N$ queue with different scheduling and push-out schemes. Our work is primarily motivated by the study of the performance of an output link of ATM switches with traffic of two classes with different priorities. The future Broadband Integrated Services Digital Network (B-ISDN) will provide an integrated access that will support a wide variety of applications for its customers in a flexible and cost-effective manner. The transfer mode chosen by the CCITT [1,2] for B-ISDN is called the ATM. ATM is a high bandwidth, low-delay, packet-like switching and multiplexing technique. ATM can switch all types of

* Corresponding author, Tel. +1 404 894 5141, fax +1 404 853 9140, Email: ian@armani.gatech.edu. The work of Akyldiz was supported in part by National Science Foundation (NSF) under Grant No. CCR-90-11981.

SSDI 0166-5316(93)E0030-9

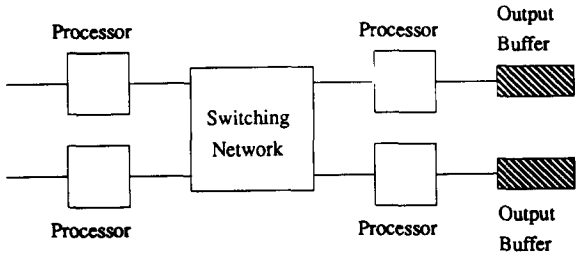


Fig. 1. An ATM switch.

traffic, ranging from low-bit rate to high rate traffic, in a packet format of fixed length called *cell* using a simplified end-to-end protocol. Various different media such as voice, data, video and graphics can be accommodated in an ATM network. Each service component of a multimedia system requires its own grade of service (GOS). For example, voice packets are more sensitive to delay than data packets. A data packet requires a higher level of protection against loss than a voice packet. Therefore, the network should be designed and controlled to satisfy these greatly differing performance requirements. Various service and buffer control mechanisms have been proposed, ranging from the dedicated buffer access for each traffic class to the shared buffer with or without push-out scheme [4,6].

Consider an ATM switch consisting of a fast switching network, processors and buffers as shown in Fig. 1. The processors are responsible for managing the buffers, among other things. Cell delay and loss may occur when cells pass through the switch and the output buffer. If we assume that there are two traffic classes in the ATM switch, the output buffer can be modeled as a finite buffer two-class queue as shown in Fig. 2. The server in Fig. 2 represents the trunk for the transmission of cells out of the output buffer. Our work is motivated by the study of the performance of the queueing model given in Fig. 2.

There is only a small number of published studies on the push-out priority schemes. Doshi and Heffes [3] have described and analyzed an overload control algorithm using the push-out scheme with replacement strategy FIFO for the M/M/1/N queue. Sumita and Ozawa [4] have derived conservation laws for systems using a push-out scheme. They have also proposed a mixed head-of-line service discipline for the push-out scheme in which, when the server becomes idle, the server will serve class 1 packets first with a probability α or class 2 packets first with a $1-\alpha$. They obtain the mean waiting times for packet classes 1 and 2. Their result shows that the two mean waiting times are subject to a linear restriction. Furthermore,

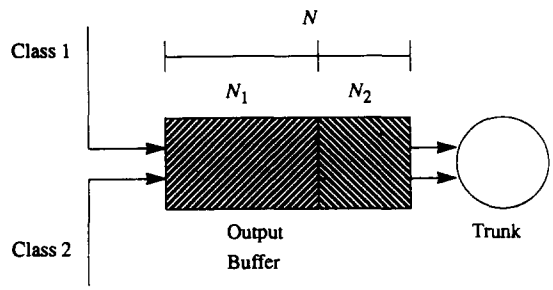


Fig. 2. A queueing model.

Hebuterne and Gravey [6] have evaluated the loss probabilities of a similar system assuming a Poisson arrival process, a deterministic service time and the replacement strategy FIFO. Their solution is not applicable to a general service time distribution. They observe a tagged low priority packet from joining until leaving the system and derive the probabilities that this packet will either be served or discarded from the system. Kroner [7] presents a method to compute the loss probabilities of an M_1, $M_2/G/1/N$ push-out system with FIFO service discipline. He considers three different space priority mechanisms, namely, push-out scheme, partial buffer sharing, and the scheme with a separate route for each traffic class, and determines the push-out scheme as the best scheme in terms of loss probabilities. A finite-buffer priority queue M_1, M_2/G_1, $G_2/1/N$ is analyzed in [8]. However, no push-out scheme and buffer space division are considered in [8]. Saito [10] analyzes an $MMPP1 + MMPP2/G/1/K$ queue with a push-out scheme. Recently, some related studies also appeared in [12–14].

In this paper we present an exact method to compute loss probabilities, the distribution of the number of class 1 packets in the system and the mean waiting time of a class 1 packet. An approximate solution is given for the computation of the mean waiting time for class 2 packets. Our model in the paper differs from the other analyzed push-out models in that we allow general service time distributions for classes 1 and 2, a general service discipline and a divided buffer management scheme. This paper is organized as follows. In Section 2 we describe the model. In Section 3 we outline the calculation of loss probabilities. In Section 4 we present a method for computing the steady state probabilities of the number of class 1 packets and number of class 2 packets at a service beginning time in the system. In Section 5 we detail the computation for the average number of losses of packet during a service time, which has been used in Section 3. In Sections 6 and 7 we derive an exact mean waiting time computation for class 1 and an approximate mean waiting time computation for class 2, respectively. We give numerical examples in Section 8 with some discussion about the results. Finally, in Section 9 we conclude the paper.

2. Model description

We consider an M_1, M_2/G_1, $G_2/1/N$ with additional features in service discipline and buffer management as will be explained shortly in this section. Since we allow general, variable length service times, we will call a customer a packet instead of a cell called in ATM networks. Two classes of packets are denoted by class 1 and class 2. The arrival process for class s ($s = 1, 2$) is Poisson with rate λ_s ($s = 1, 2$). (Note that we do not consider the bursty traffic here.) The service time of a class s ($s = 1, 2$) can be a random variable with a general probability distribution. Let $b_s(x)$ and $\bar{b}_s$ denote, respectively, the probability density function and the mean of the service time of a class s packet ($s = 1, 2$). Service times and arrival processes are independent of each other.

There are N number of total buffer spaces in the system, where N is finite and can be divided as $N = N_1 + N_2$. The number of class 1 packets waiting for service cannot be more than $N_1 - 1$. (Total number of class 1 packets in the system may be N_1 if the one currently in service is class 1.) An arrival of class 1 packet can join the system by taking an unoccupied buffer space, if it finds that there are less than $N_1 - 1$ class 1 packets waiting for service and there is

an unoccupied buffer space in the system upon its arrival. An arrival of class 1 is lost if there are, upon its arrival, $N_1 - 1$ class 1 packets waiting for service in the system, even though there is an unoccupied buffer space in the system. In the contrast, an arrival of class 2 can take an unoccupied buffer space anywhere in the system upon its arrival as long as there is one. It is lost, otherwise. However, an arrival of class 1 can join the system by replacing (pushing out) a waiting class 2 packet in the system if it finds that there are less than $N_1 - 1$ class 1 packets waiting for service and that there is no unoccupied buffer space in the system upon its arrival. The class 2 packet being pushed out is lost.

The service discipline is specified by $\alpha_s(i, j)$, $s = 1, 2$, where $\alpha_1(i, j)$ is the probability that class 1 packet will be served when there are i class 1 and j class 2 packets in the system at the beginning of the service. $\alpha_2(i, j)$ can be similarly defined. Obviously, $\alpha_2(i, j) = 1 - \alpha_1(i, j)$ when $i + j > 0$. $\alpha_s(0, 0)$ $(s = 1, 2)$ is undefined. We also assume that the server will not be idle as long as there is some packet in the system waiting for service. Equivalently, this is to say that $\alpha_1(i, 0) = 1$, $i > 0$, and $\alpha_2(0, j) = 1$, $j > 0$.

Using the α we can model several different scheduling disciplines in the system.

a) *Head of Line (HOL) Scheduling*

$$\alpha_1(i, j) = 1, \quad \text{if } i > 0,$$

b) *Shortest Line First (SLF) Scheduling*

$$\alpha_1(i\,j) = \begin{cases} 1 & \text{if } i \leqslant j \\ 0 & \text{if } i > j. \end{cases}$$

c) *Longest Line First (LLF) Scheduling*

$$\alpha_1(i, j) = \begin{cases} 1 & \text{if } i \geqslant j \\ 0 & \text{if } i < j. \end{cases}$$

d) *Random (RS) Scheduling*

$$\alpha_1(i, j) = p \quad \text{if } i > 0 \text{ and } j > 0,$$

i.e., the server will serve class 1 with probability p and class 2 with probability $1 - p$. We should point out that although α_1 is general, it has to be a function of the (i, j), numbers of packets of two classes, and therefore, it cannot exactly model schemes that depend more than (i, j). For example, FIFO and LIFO. Note that from the loss probabilities point of view, it does not matter in which order the packets of the same class are served and which class 2 packet will be pushed out.

3. Loss probabilities

Packet losses occur only when the server is busy. A packet can be lost if either there is no space available in the buffer upon its arrival or it is pushed out from the buffer while waiting for service. Let l_{12} be the loss probability for a packet of either class 1 or class 2 and s_{12} be the ratio of packets lost during a long period of time over packets served in the same period time. It could be true that numbers of packets lost during a service time (the time used to serve a

packet) may not be independent. Nevertheless, an average of the losses should exist and it is equal to s_{12}. Consider a time period T when the system reaches the steady state. On the average there are $(\lambda_1 + \lambda_2)T$ arrivals from classes 1 and 2 in T, and $(1 - l_{12})$ fraction of them is served. Therefore, the average number of total packet losses in T is $(\lambda_1 + \lambda_2)T(1 - l_{12})s_{12}$. By definition, l_{12} is the ratio of the average number of total losses in T to the average number of total arrivals in T. Thus,

$$l_{12} = \frac{(\lambda_1 + \lambda_2)T(1 - l_{12})s_{12}}{(\lambda_1 + \lambda_2)T} = (1 - l_{12})s_{12}. \tag{1}$$

From Eq. (1) we get

$$l_{12} = \frac{s_{12}}{1 + s_{12}}. \tag{2}$$

Obviously, s_{12} is also equal to the average number of class 1 or class 2 packets lost during a service time. The computation of s_{12} will be shown later (Section 5) from this point of view.

Similarly, let l_1 be the loss probability of a class 1 packet and s_1 be the average number of class 1 packets lost per packet served. We have

$$l_1 = \frac{(\lambda_1 + \lambda_2)T(1 - l_{12})s_1}{\lambda_1 T} = \frac{(\lambda_1 + \lambda_2)(1 - l_{12})s_1}{\lambda_1}. \tag{3}$$

Finally, let l_2 be the loss probability of a class 2 packet. Using

$$(\lambda_1 + \lambda_2)l_{12} = \lambda_1 l_1 + \lambda_2 l_2, \tag{4}$$

we obtain

$$l_2 = \frac{(\lambda_1 + \lambda_2)l_{12} - \lambda_1 l_1}{\lambda_2}. \tag{5}$$

s_1 will be computed in Section 5 from the point of view of the average number of class 1 packets lost during a service time.

4. Steady state probabilities

The average number of packet losses during a service time can be computed by conditioning on the number of class 1 and the number of class 2 packets in the system at the beginning of the service time. In this section, we will compute the probabilistic distribution of the numbers of class 1 and class 2 packets in the system at the beginning of a service time. We proceed as follows. First, the distribution of the numbers of packets left in the system at a packet's departure time is computed, and then the distribution of the numbers of packets at the beginning of a service time is derived from the departure time distribution.

Let (i, j) denote that there are i class 1 packets and j class 2 packets in the queue at a packet's departure time. Since we restrict our view at a packet's departure time, (i, j) constitutes a Markov chain (imbedded Markov chain), where $0 \leqslant i < N_1$, $j \geqslant 0$ and $i + j \leqslant N - 1$.

Let $p(i, j)$, $0 \leqslant i < N_1$, $j \geqslant 0$ and $i + j \leqslant N - 1$, be the steady state probability that the system is in state (i, j) at a packet's departure time and $P_{(i,j);(k,l)}$ be the one step transition probability from state (i, j) to state (k, l). Clearly, $P_{(i,j);(k,l)}$ is a function of arrival rates and service time. The Markov chain is totally determined by $P_{(i,j);(k,l)}$. To facilitate the expression for $P_{(i,j);(k,l)}$, the following definitions are introduced.

Definition 1. $I(n, \lambda, b(x))$ is the probability that there are exactly n Poisson arrivals with arrival rate λ during a service time of which the probability density function (pdf) is $b(x)$.

$$I(n, \lambda, b(x)) = \int_0^\infty \frac{(\lambda x)^n}{n!} e^{-\lambda x} b(x)\, dx$$

$$= \frac{\lambda^n}{n!} \int_0^\infty x^n\, e^{-\lambda x} b(x)\, dx \tag{6}$$

Definition 2. $I(\geqslant n, \lambda, b(x))$ is the probability that there are at least n Poisson arrivals with rate λ during a service time of which the pdf is $b(x)$.

$$I(\geqslant n, \lambda, b(x)) = 1 - \sum_{i=0}^{n-1} I(i, \lambda, b(x)). \tag{7}$$

Definition 3. $II(n_1, n_2, \lambda_1, \lambda_2, b(x))$ is the probability that there are exactly n_1 and n_2 arrivals from Poisson processes with arrival rates λ_1 and λ_2, respectively, during a service time of which the pdf is $b(x)$.

$$II(n_1, n_2, \lambda_1, \lambda_2, b(x))$$

$$= \int_0^\infty \frac{(\lambda_1 x)^{n_1}}{n_1!} \exp(-\lambda_1 x) \frac{(\lambda_2 x)^{n_2}}{n_2!} \exp(-\lambda_2 x) b(x)\, dx$$

$$= \frac{\lambda_1^{n_1} \lambda_2^{n_2} (n_1 + n_2)!}{(\lambda_1 + \lambda_2)^{n_1 + n_2} n_1! n_2!} I(n_1 + n_2, \lambda_1 + \lambda_2, b(x)). \tag{8}$$

Definition 4. $II(\geqslant n_1, n_2, \lambda_1, \lambda_2, b(x))$ is the probability that there are at least n_1 and exactly n_2 arrivals from Poisson processes with arrival rates λ_1 and λ_2, respectively, during a service time of which the pdf is $b(x)$.

$$II(\geqslant n_1, n_2, \lambda_1, \lambda_2, b(x))$$

$$= I(n_2, \lambda_2, b(x)) - \sum_{i=0}^{n_1 - 1} II(i, n_2, \lambda_1, \lambda_2, b(x)). \tag{9}$$

The following two probabilities are similarly defined. They are

$$II(n_1, \geqslant n_2, \lambda_1, \lambda_2, b(x))$$

$$= II(\geqslant n_2, n_1, \lambda_2, \lambda_1, b(x)) \tag{10}$$

and

$$
\begin{aligned}
&II(\geqslant n_1,\ \geqslant n_2,\ \lambda_1,\ \lambda_2,\ b(x)) \\
&= \int_0^\infty \sum_{i=n_1}^{\infty} \sum_{j=n_2}^{\infty} \frac{(\lambda_1 x)^i}{i!} \exp(-\lambda_1 x) \frac{(\lambda_2 x)^j}{j!} \exp(-\lambda_2 x) b(x)\, \mathrm{d}x \\
&= I(\geqslant n_2,\ \lambda_2,\ b(x)) - \sum_{i=0}^{n_1-1} II(i,\ \geqslant n_2,\ \lambda_1,\ \lambda_2,\ b(x)).
\end{aligned} \tag{11}
$$

If $n_2 < 0$ in the above definitions, the computation should not count the arrival process related to n_2. For example, $II(n_1, n_2, \lambda_1, \lambda_2, b(x)) = I(n_1, \lambda_1, b(x))$ if $n_2 < 0$.

Now we are ready to compute $P_{(i,j);(k,l)}$.

Case 1. For $i = 0$ and $j = 0$: the class of the packet that will be served next depends on the class from which the next packet comes. Since both arrival processes are Poisson, with probability $\lambda_1/(\lambda_1 + \lambda_2)$, the next packet comes from class 1 and with probability $\lambda_2/(\lambda_1 + \lambda_2)$ from class 2. Therefore, we have:

$$
P_{(0,0);(k,l)} = \begin{cases}
\left[\dfrac{\lambda_1}{\lambda_1+\lambda_2} II(k, l, \lambda_1, \lambda_2, b_1(x)) + \dfrac{\lambda_2}{\lambda_1+\lambda_2} II(k, l, \lambda_1, \lambda_2, b_2(x))\right] \\
\quad \text{if } k < N_1 - 1 \text{ and } k + l < N - 1 \\
\left[\dfrac{\lambda_1}{\lambda_1+\lambda_2} II(k, \geqslant l, \lambda_1, \lambda_2, b_1(x)) + \dfrac{\lambda_2}{\lambda_1+\lambda_2} II(k, \geqslant l, \lambda_1, \lambda_2, b_2(x))\right] \\
\quad \text{if } k < N_1 - 1 \text{ and } k + l = N - 1 \\
\left[\dfrac{\lambda_1}{\lambda_1+\lambda_2} II(\geqslant k, l, \lambda_1, \lambda_2, b_1(x)) + \dfrac{\lambda_2}{\lambda_1+\lambda_2} II(\geqslant k, l, \lambda_1, \lambda_2, b_2(x))\right] \\
\quad \text{if } k = N_1 - 1 \text{ and } k + l < N - 1 \\
\left[\dfrac{\lambda_1}{\lambda_1+\lambda_2} II(\geqslant k, \geqslant l, \lambda_1, \lambda_2, b_1(x)) + \dfrac{\lambda_2}{\lambda_1+\lambda_2} II(\geqslant k, \geqslant l, \lambda_1, \lambda_2, b_2(x))\right] \\
\quad \text{if } k = N_1 - 1 \text{ and } k + N - 1.
\end{cases} \tag{12}
$$

Case 2. For $i = 0$ and $j > 0$: a packet of class 2 will be served. Let $\Delta_1 = k$ and $\Delta_2 = l - (j - 1)$. Δ_1 and Δ_2 indicate, respectively, numbers of changes of classes 1 and 2 packets during a service time. Δ_2 may become negative if $j - 1 > N_2$ and $\Delta_1 > N - j$. Therefore we have:

$$
P_{(0,j);(k,l)} = \begin{cases}
0 & \text{if } \Delta_2 < 0 \text{ and } k + l < N - 1 \\
II(\Delta_1, \Delta_2, \lambda_1, \lambda_2, b_2(x)) & \text{if } k < N_1 - 1 \text{ and } k + l < N - 1 \\
II(\Delta_1, \geqslant \Delta_2, \lambda_1, \lambda_2, b_2(x)) & \text{if } k < N_1 - 1 \text{ and } k + l = N - 1 \\
II(\geqslant \Delta_1, \Delta_2, \lambda_1, \lambda_2, b_2(x)) & \text{if } k = N_1 - 1 \text{ and } k + l < N - 1 \\
II(\geqslant \Delta_1, \geqslant \Delta_2, \lambda_1, \lambda_2, b_2(x)) & \text{if } k = N_1 - 1 \text{ and } k + l = N - 1.
\end{cases} \tag{13}
$$

Case 3. For $i>0$ and $j=0$: a packet of class 1 will be served. Let $\Delta_1 = k-(i-1)$ and $\Delta_2 = l$. Δ_1 and Δ_2 have the same interpretation as that in case 2. Note that Δ_1 cannot be negative. We have

$$P_{(i,0);(k,l)} = \begin{cases} 0 & \text{if } \Delta_1 < 0 \\ II(\Delta_1, \Delta_2, \lambda_1, \lambda_2, b_1(x)) & \text{if } k < N_1 - 1 \text{ and } k+l < N-1 \\ II(\Delta_1, \geqslant \Delta_2, \lambda_1, \lambda_2, b_1(x)) & \text{if } k < N_1 - 1 \text{ and } k+l = N-1 \\ II(\geqslant \Delta_1, \Delta_2, \lambda_1, \lambda_2, b_1(x)) & \text{if } k = N_1 - 1 \text{ and } k+l < N-1 \\ II(\geqslant \Delta_1, \geqslant \Delta_2, \lambda_1, \lambda_2, b_1(x)) & \text{if } k = N_1 - 1 \text{ and } k+l = N-1. \end{cases} \tag{14}$$

Case 4. For $i>0$ and $j>0$: a packet of class s ($s=1, 2$) will be served with probability $\alpha_s(i, j)$. Let $\Delta_1^1 = k-(i-1)$, $\Delta_2^1 = l-j$, $\Delta_1^2 = k-i$, and $\Delta_2^2 = l-(j-1)$. Δ_1^1 and Δ_2^1 indicate, respectively, the numbers of changes of classes 1 and 2 packets in a class 1 packet service time, and Δ_1^2 and Δ_2^2 have the similar interpretation except that in a class 2 packet service time. Note that it is impossible to have $\Delta_1^1 < 0$ or $\Delta_2^1 < 0$ and $k+l < N-1$ when a class 1 packet is being served and $\Delta_1^2 < 0$ or $\Delta_2^2 < 0$ and $k+l < N-1$ when a class 2 packet is being served. Let

$$\delta(\Delta_1, \Delta_2) = \begin{cases} 0 & \text{if } \Delta_1 < 0 \\ & \quad \text{or } \Delta_2 < 0 \text{ and } k+l < N-1 \\ 1 & \text{otherwise.} \end{cases} \tag{15}$$

Thus,

$$P_{(i,j);(k,l)} = \begin{cases} \alpha_1(i,j)\delta(\Delta_1^1, \Delta_2^1) II(\Delta_1^1, \Delta_2^1, \lambda_1, \lambda_2, b_1(x)) + \alpha_2(i,j)\delta(\Delta_1^2, \Delta_2^2) II(\Delta_1^2, \Delta_2^2, \lambda_1, \lambda_2, b_2(x)) \\ \quad \text{if } k < N_1 - 1 \text{ and } k+l < N-1 \\ \alpha_1(i,j)\delta(\Delta_1^1, \Delta_2^1) II(\Delta_1^1, \geqslant \Delta_2^1, \lambda_1, \lambda_2, b_1(x)) + \alpha_2(i,j)\delta(\Delta_1^2, \Delta_2^2) II(\Delta_1^2, \geqslant \Delta_2^2, \lambda_1, \lambda_2, b_2(x)) \\ \quad \text{if } k < N_1 - 1 \text{ and } k+l = N-1 \\ \alpha_1(i,j)\delta(\Delta_1^1, \Delta_2^1) II(\geqslant \Delta_1^1, \Delta_2^1, \lambda_1, \lambda_2, b_1(x)) + \alpha_2(i,j)\delta(\Delta_1^2, \Delta_2^2) II(\geqslant \delta_1^2, \Delta_2^2, \lambda_1, \lambda_2, b_2(x)) \\ \quad \text{if } k = N_1 - 1 \text{ and } k+l < N-1 \\ \alpha_1(i,j)\delta(\Delta_1^1, \Delta_2^1) II(\geqslant \Delta_1^1, \geqslant \Delta_2^1, \lambda_1, \lambda_2, b_1(x)) + \alpha_2(i,j)\delta(\Delta_1^2, \Delta_2^2) II(\geqslant \Delta_1^2, \geqslant \Delta_2^2, \lambda_1, \lambda_2, b_2(x)) \\ \quad \text{if } k = N_1 - 1 \text{ and } k+l = N-1. \end{cases} \tag{16}$$

This completes the computation for $P_{(i,j);(k,l)}$, where $0 \leqslant i,\ k < N_1;\ j, l \geqslant 0$ and $i+j,\ k+l < N$.

The steady state probabilities, $p(i, j)$, should observe the law of conservation:

$$p(i,j) = \sum_{\text{all } (k,l)} p(k,l) P_{(k,l),(i,j)} \text{ for } 0 \leqslant i \leqslant N_1;\ j \geqslant 0;\ i+j \leqslant N-1, \tag{17}$$

and also

$$\sum_{\text{all } (i,j)} p(i,j) = 1. \tag{18}$$

We can compute the values of $p(i, j)$ by solving Eqs. (17) and (18) numerically, which involves

$(N_1(2N - N1 + 1)/2$ independent linear equations. We used the svd method in our numerical experiment and no numerically nonstable case was encountered.

Let $q(i, j)$ be the probability that there are i class 1 packets and j class 2 packets at the beginning of a service time. Except for the first packet, the beginning of a service is preceded by the departure of the last packet served. There is then a one-to-one correspondence between a packet's departure and the beginning of the service of the next packet. If there is some packet left in the system at a packet's departure time, then the beginning of the service time for the next packet coincides with the departure time and they should observe the same packets left in the system. However, if there is no packet left at a departure time, there will be either (1, 0) or (0, 1) packet in the system at the beginning of the next service time depending on from which class the next packet comes. It is clear now that the possible state (i, j) at the beginning of a service is $i + j \geqslant 1$ and $i + j \leqslant N - 1$ for $N \geqslant 2$. So $q(i, j)$ is computed as follows:

1. For $N = 1$, the only possible states are (1, 0) and (0, 1):

$$q(1, 0) = \frac{\lambda_1}{\lambda_1 + \lambda_2},$$

and

$$q(0, 1) = \frac{\lambda_2}{\lambda_1 + \lambda_2}.$$

2. For $N \geqslant 2$:

$$q(i, j) = \begin{cases} \dfrac{\lambda_1}{\lambda_1 + \lambda_2} p(0, 0) & \text{if } i = 1 \text{ and } j = 0 \text{ and } N_1 = 1 \\ p(1, 0) + \dfrac{\lambda_1}{\lambda_1 + \lambda_2} p(0, 0) & \text{if } i = 1 \text{ and } j = 0 \text{ and } N_1 > 1 \\ p(0, 1) + \dfrac{\lambda_2}{\lambda_1 + \lambda_2} p(0, 0) & \text{if } i = 0 \text{ and } j = 1 \\ p(i, j) & \text{if } i + j \geqslant 2 \text{ and } i + j \leqslant N - 1. \end{cases} \tag{19}$$

5. Average number of losses during a service time

We define $L(n, \lambda, B(x))$ as the average number of arrivals after the first n arrivals of a Poisson arrival process with rate λ during a service time whose pdf is $b(x)$, i.e. $L(n, \lambda, b(x))$ is the average number of arrivals counted after the first n arrivals during a service time. By definition

$$\begin{aligned} &L(n, \lambda, b(x)) \\ &\quad = \sum_{k=n+1}^{\infty} \int_0^{\infty} (k - n) \frac{(\lambda x)^k}{k!} e^{-\lambda x} b(x)\, dx \\ &\quad = \lambda \bar{b} - \sum_{k=1}^{n} k I(k, \lambda, b(x)) - n I(\geqslant (n + 1), \lambda, b(x)), \end{aligned} \tag{20}$$

where $\bar{b}$ is the mean of $b(x)$. Suppose there are (i, j), $i > 0$ and $j > 0$, packets in the system (including the one which is going to receive service) at the beginning of a service time. Consider the number of class 1 packets that may be lost during the service time. If a class 1 packet is served, then the first $(N_1 - i)$ arrivals of class 1 packets during the service time can join the system. After that, all arrivals are lost due to the fact that there are $(N_1 - 1)$ class 1 packets waiting in the queue. Therefore, the average number of class 1 packets lost during a service time which begins with i class 1 packets and j class 2 packets is equal to the average number of class 1 packets arrived after the first $(N_1 - i)$ class 1 arrivals if the packet in service is class 1 or is equal to the average number of class 1 packets arrived after the first $(N_1 - (i+1))$ class 1 arrivals if the packet in service is class 2. Therefore,

$$s_1 = \sum_{i>0, j\geqslant 0} q(i, j)\alpha_1(i, j)L(N_1 - i, \lambda_1, b_1(x)) + \sum_{j>0, i\geqslant 0} q(i, j)\alpha_2(i, j)L(N_1 - i - 1, \lambda_1, b_2(x)). \tag{21}$$

The idea for computing s_{12}, the average number of losses of packets of either class during a service time is similar but more complicated. Again, suppose there are (i, j) packets in the system at the beginning of a service time. First, let us consider the case where the next packet to be served is a class 1 packet. Let $t = \min\{N_1 - i, N - i - j\}$ and $y = \max\{0, N_2 - j\}$. t can be thought as the maximum number of class 1 arrivals during the service time which result in no packets being lost or pushed out, and y is the number of unoccupied buffer spaces that only class 2 packets can take. $(N - i - j)$ is the total number of unoccupied buffer spaces at the beginning of the service time. Assuming that there are k and l arrivals from classes 1 and 2, respectively, during a service time beginning at state (i, j), the number of total losses of packets of the two classes during the service time is

$$s_{12}|_{(i,j);(k,l)} = \begin{cases} l - (N - i - j - k) & \text{if } k \leqslant t \text{ and } l > N - i - j - k \\ k - t & \text{if } k > t \text{ and } l \leqslant y \\ k - t + l - y & \text{if } k > t \text{ and } l > y \\ 0 & \text{otherwise.} \end{cases} \tag{22}$$

Therefore, the average number of packets of the two classes lost during a class 1 service time beginning at (i, j) is

$$\begin{aligned} s_{12}|_{(i,j)} &= \int_0^\infty \sum_{k=0}^{t} \sum_{l=N-i-j-k+1}^{\infty} \frac{(\lambda_1 x)^k}{k!}\exp(-\lambda_1 x)\frac{(\lambda_2 x)^l}{l!} \\ &\quad \times \exp(-\lambda_2 x)(l - (N - i - j - k))b_1(x)\,dx + \int_0^\infty \sum_{k=t+1}^{\infty} \sum_{l=0}^{y} \frac{(\lambda_1 x)^k}{k!} \\ &\quad \times \exp(-\lambda_1 x)\frac{(\lambda_2 x)^l}{l!}\exp(-\lambda_2 x)(k - t)b_1(x)\,dx \\ &\quad + \int_0^\infty \sum_{k=t+1}^{\infty} \sum_{l=y+1}^{\infty} \frac{(\lambda_1 x)^k}{k!}\exp(-\lambda_1 x)\frac{(\lambda_2 x)^l}{l!}\exp(-\lambda_2 x)(k - t + l - y)b_1(x)\,dx \end{aligned}$$

$$= \frac{\lambda_2}{\lambda_1} \sum_{k=0}^{t} (k+1) II(k+1, \geqslant (N-i-j-k), \lambda_1, \lambda_2, b_1(x))$$
$$- \sum_{k=0}^{t} (N-i-j-k) II(k, \geqslant (N-i-j-k+1), \lambda_1, \lambda_2, b_1(x))$$
$$+ (\lambda_1+\lambda_2)\bar{b}_1 - \sum_{k=0}^{t} kI(k, \lambda_1, b_1(x)) - tI(\geqslant (t+1), \lambda_1, b_1(x))$$
$$- \sum_{l=0}^{y} lI(l, \lambda_2, b_1(x)) - \frac{\lambda_2}{\lambda_1} \sum_{k=0}^{t} (k+1) II((k+1), \geqslant y, \lambda_1, \lambda_2, b_1(x))$$
$$- yII(\geqslant (t+1), \geqslant (y+1), \lambda_1, \lambda_2, b_1(x)). \tag{23}$$

The case of the next packet to be served being a class 2 packet can be derived similarly. The result will be the same except that t and y are calculated slightly differently: $t = \min\{N_1 - (i+1), N-i-j\}$ and $y = \max\{0, N_2 - (j-1)\}$, where $j \geqslant 1$. Let

$$g(i, j, t, y, b(x)) = \frac{\lambda_2}{\lambda_1} \sum_{k=0}^{t} (k+1) II(k+1, \geqslant (N-i-j-k), \lambda_1, \lambda_2, b(x))$$
$$- \sum_{k=0}^{t} (N-i-j-k) II(k, \geqslant (N-i-j-k+1), \lambda_1, \lambda_2, b(x))$$
$$+ (\lambda_1+\lambda_2)\bar{b} - \sum_{k=0}^{t} kI(k, \lambda_1, b(x)) - tI(\geqslant (t+1), \lambda_1, b(x))$$
$$- \sum_{l=0}^{y} lI(l, \lambda_2, b(x)) - \frac{\lambda_2}{\lambda_1} \sum_{k=0}^{t} (k+1) II((k+1), \geqslant y, \lambda_1, \lambda_2, b(x))$$
$$- yII(\geqslant (t+1), \geqslant (y+1), \lambda_1, \lambda_2, b(x)), \tag{24}$$

where $\bar{b} = \int_0^\infty xb(x)\,dx$. Then, s_{12}, the mean number of losses of packets of the two classes in a service time, is

$$s_{12} = \sum_{i>0, j\geqslant 0} q(i, j)\alpha_1(i, j) g(i, j, t_1(i, j), y_1(j), b_1(x))$$
$$+ \sum_{j>0, i\geqslant 0} q(i, j)\alpha_2(i, j) g(i, j, t_2(i, j), y_2(j), b_2(x)), \tag{25}$$

where

$$t_1(i, j) = \min\{N_1 - i, N-i-j\}$$
$$y_1(j) = \max\{0, N_2 - j\}$$
$$t_2(i, j) = \min\{N_1 - (i+1), N-i-j\}$$
$$y_2(j) = \max\{0, N_2 - (j-1)\}. \tag{26}$$

Special Case: if $N_1 = N$, i.e. class 1 packets can take any buffer space in the system, the computation for s_{12} is much simpler. Suppose there are i and j packets of classes 1 and 2,

respectively, at the beginning of a service time. Any arrival after the first $(N-i-j)$ arrivals from both classes either is lost or pushed out a class 2 packet. Therefore, Eq. (25) is simplified to the following form:

$$s_{12} = \sum_{i>0, j\geq 0} q(i, j)\alpha_1(i, j)L(N-i-j, \lambda_1+\lambda_2, b_1(x)) + \sum_{j>0, i\geq 0} q(i, j)\alpha_2(i, j)L(N-i-j, \lambda_1-\lambda_2, b_2(x)). \quad (27)$$

6. Exact computation of the queue length distribution and the mean waiting time for class 1

In this section, we compute the probability of being i, $0 \leq i \leq N_1$, class 1 packets in the system at a random time. The result is then used to compute the mean waiting time of a class 1 packet. Since Poisson arrivals see time average [9], the probability that there are i, $0 \leq i \leq N_1$, class 1 packets in the system at a random time is equal to the probability that there are i class 1 packets in the system at the arrival time of a class 1 packet. So we will compute the probability from the point of view of an arriving class 1 packet. As before, T is used to denote a period of time when the system is in the steady state.

Let a_{idle} be the probability that a class 1 packet finds the server idle upon its arrival. This is possible only when there is no packet in the system at a service completion time and the next arrival is a class 1 packet. Therefore, we have

$$a_{\text{idle}} = \frac{(\lambda_1+\lambda_2)T(1-l_{12})p(0, 0)\lambda_1/(\lambda_1+\lambda_2)}{\lambda_1 T} = (1-l_{12})p(0, 0). \quad (28)$$

Let a_k, $0 \leq k \leq N_1$, be the average number of class 1 arrivals which see k class 1 packets in the system upon their arrivals during a service time. The population of class 1 packets can be divided into two sets: those lost upon their arrivals and those served. The class 1 packets lost can see only $N_1 - 1$ or N_1 class 1 packets in the system upon their arrivals, while class 1 packets served can see $0 \leq k \leq N_1 - 1$ class 1 packets in the system upon their arrivals. The number of losses of class 1 packet in a service time is

$$s_1 = a'_{N_1} + a'_{N_1-1}, \quad (29)$$

where

$$a'_{N_1} = \sum_{i>0, j\geq 0} q(i, j)\alpha_1(i, j)L(N_1-i, \lambda_1, b_1(x)), \quad (30)$$

and

$$a'_{N_1-1} = \sum_{j>0, i\geq 0} q(i, j)\alpha_2(i, j)L(N_1-i-1, \lambda_1, b_2(x)). \quad (31)$$

a'_{N_1-1} and a'_{N_1} are the average numbers of class 1 packets lost during a service time which see $N_1 - 1$ and N_1 class 1 packets in the system upon their arrivals, respectively.

The average number of class 1 packets served which see k, $0 \leqslant k \leqslant N_1 - 1$, class 1 packets in the system upon their arrivals during a service time can be computed as follows. If there is no class 1 packet at the beginning of a service, there should be at least $k+1$ class 1 arrivals during the service time for only the $(k+1)$st arrival will observe k class 1 packets in the system upon its arrival. If there are i, $i \geqslant 1$, class 1 packets in the system at the beginning of a service, there should be at least $k-i+1$ class 1 arrivals during the service time for only the $(k-i+1)$st arrival will observe k class 1 packets in the system upon its arrival. Obviously, $i \leqslant k$ and $i+1 < N_1$ if a packet of class 2 is in service. Therefore we have

$$a_k = \begin{cases} \Sigma_{i \leqslant k} q(i, j)\alpha_1(i, j) I(\geqslant (k-i+1), \lambda_1, b_1(x)) \\ \quad + \Sigma_{i \leqslant k} q(i, j)\alpha_a(i, j) I(\geqslant (k-1+1), \lambda_1, b_2(x)) \\ \qquad \text{if } 0 \leqslant k < N_1 - 1 \\ \Sigma_{i \leqslant k} q(i, j)\alpha_1(i, j) I(\geqslant (k-1+1), \lambda_1, b_1(x)) + a'_{N_1-1} \\ \qquad \text{if } k = N_1 - 1 \\ a'_{N_1} \quad \text{if } k = N_1. \end{cases} \tag{32}$$

Now let B_k, $0 \leqslant k \leqslant N_1$, denote the probability that a class 1 packet finds that there are k class 1 packets in the system upon its arrival. B_k is then

(a) If $k = 0$:

$$B_0 = \frac{(\lambda_1 + \lambda_2)T(1 - l_{12})a_0}{\lambda_1 T} + a_{\text{idle}}$$

$$= \frac{(\lambda_1 + \lambda_2)(1 - l_{12})a_0}{\lambda_1} + a_{\text{idle}}. \tag{33}$$

(b) If $k = 1, 2, \ldots, N_1$:

$$B_k = \frac{(\lambda_1 + \lambda_2)T(1 - l_{12})a_k}{\lambda_1 T}$$

$$= \frac{(\lambda_1 + \lambda_2)(1 - l_{12})a_k}{\lambda_1}. \tag{34}$$

It can be verified that

$$\sum_{k=0}^{N_1} B_k = 1. \tag{35}$$

The mean system time of a class 1 packet can be computed by Little's law. The average number of packets in the system equal to the mean system time multiplied by the effective arrival rate. Therefore, the mean waiting time of a class 1 packet, $\bar{w}_1$, is

$$\bar{w}_1 = \frac{\Sigma_{k=1}^{N_1} k B_k}{(1 - l_1)\lambda_1} - \bar{b}_1 \tag{36}$$

7. Approximate mean waiting time for class 2

We are unable to compute the mean waiting time of a class 2 packet exactly due to the fact that a class 2 packet may get pushed out after joining the waiting queue. However, in the context of an ATM switch, the loss probability of a class 2 packet, which can be computed exactly by the method described in Section 3, is usually very small. When the loss probability is small, we can analyze approximately the mean waiting time of a class 2 packet by overlooking part of lost packets. Particularly, in the following we will present an approximate method of computing the mean waiting time of a class 2 packet with the assumption that only those arrivals of class 2 packets which arrive after the first $(N-i-j)$ arrivals of class 2 packets during a service time beginning with i class 1 and j class 2 packets in the system will be lost and there is no push-out loss.

If there are i class 1 and j class 2 packets in the system at the beginning of service, $(N-i-j)$ is the number of unoccupied buffer spaces at the beginning of the service. An arrival among the first $N-i-j$ arrivals of class 2 packets during a service may or may not be lost, depending on the number of class 1 packets arrived ahead of it during the service time. The arrivals of class 2 packets after the first $N-i-j$ arrivals of class 2 are always lost. Thus, the number of actual losses of class 2 packets in computing the mean waiting time of a class 2 packet is reduced. We will comment on the accuracy of it shortly. As before, we compute the mean number of class 2 packets in the system at a random time first, which can be carried out equivalently by computing the mean number seen by an arriving class 2 packet. We then use Little's law to compute the mean waiting time. Because of the assumption, the mean number of class 2 packets in the system at a random time computed this way is greater than the actual mean number of class 2 packets served. So the mean waiting time computed with the assumption is an upper bound of the actual mean waiting time. We can also estimate a lower bound of the mean waiting time of a class 2 packet as follows. Let N_u be the upper bound of the mean number of class 2 packet at a random time in the system. Therefore, $N_u(1-l_2)$ is a lower bound of the actual mean number of class 2 packets in the system. So the error in the mean waiting time introduced by the assumption is, by Little's law, no more than l_2 fraction of the actual mean waiting time. For example, suppose the loss probability of a class 2 packet is 10^{-2}, our approximate computation of the mean waiting time of a class 2 packet will have an error of less than 1% of that of exact computation, which is probably acceptable for practical interest.

Similar to the previous section let c_{idle} be the probability that a class 2 packet finds the server idle upon its arrival, then

$$c_{\text{idle}} = (1-l_{12})p(0,0). \tag{37}$$

Let c_k, $0 \leqslant k \leqslant N$, be the average number of class 2 arrivals which see k class 2 packets in the system upon their arrivals during a service time. The arrivals of class 2 during a service time can be divided into two sets, depending on whether they are lost or not upon their arrivals. Let c'_k, $N_2+1 \leqslant k \leqslant N$, be the average number of class 2 packets in a service time which see k class

2 packets in the system upon their arrivals and are lost at the same time due to no unoccupied buffer space, then

$$c'_k = \sum_{j<k} q(N-k, j)\alpha_1(N-k, j)L(k-j, \lambda_2, b_1(x)) + \sum_{j<k} q(N-k, j)\alpha_2(N-k, j)L(k-j, \lambda_2, b_2(x))$$

$$\text{for } N_2 + 1 \leqslant k \leqslant N, \tag{38}$$

and c_k, $0 \leqslant k \leqslant N$ is

$$c_k = \begin{cases} \left[\sum_{j\leqslant k \text{ and } i+k<N} q(i, j)\alpha_1(i, j)I(\geqslant (k-j+1), \lambda_2, b_1(x)) \right. \\ \quad \left. + \sum_{j\leqslant k \text{ and } i+k<N} q(i, j)\alpha_2(i, j)I(\geqslant (k-j+1), \lambda_2, b_2(x))\right] \\ \quad \text{if } k < N_2 \\ \left[\sum_{j\leqslant k \text{ and } i+k<N} q(i, j)\alpha_1(i, j)I(\geqslant (k-j+1), \lambda_2, b_1(x)) + \right. \\ \left. \sum_{j\leqslant k \text{ and } i+k<N} q(i, j)\alpha_2(i, j)I(\geqslant (k-j+1), \lambda_2, b_2(x)) + c'_k\right] \\ \quad \text{if } N_2 \leqslant k < N \\ c'_N \quad \text{if } k = N. \end{cases} \tag{39}$$

Let D_k, $0 \leqslant k \leqslant N$, denote the probability that a class 2 packet finds there are k class 2 packets in the system upon its arrival. Then

(1) If $k = 0$:

$$D_0 = \frac{(\lambda_1+\lambda_2)T(1-l_{12})c_0}{\lambda_2 T} + c_{\text{idle}} = \frac{(\lambda_1+\lambda_2)(1-l_{12})c_0}{\lambda_2} + c_{\text{idle}} \tag{40}$$

(2) If $0 < k \leqslant N$:

$$D_k = \frac{(\lambda_1+\lambda_2)T(1-l_{12})c_k}{\lambda_2 T} = \frac{(\lambda_1+\lambda_2)(1-l_{12})c_k}{\lambda_2} \tag{41}$$

The mean waiting time of a class 2 packet, $\bar{w}_2$, is approximately

$$\bar{w}_2 \approx \frac{\sum_{k=1}^{N} kD_k}{(1+l_2)\lambda_2} - \bar{b}_2 \tag{42}$$

8. Numerical examples

In this section we present some of the experimental computations conducted in the study. It is assumed, in all of our examples, that service times for two classes are constant and equal to 1. Three service disciplines, namely HOL, SLF and LLF, are used for comparison. Let $\rho = \lambda_1 + \lambda_2$ be the total load to the system (since the service time is normalized to 1). An admissible load with respect to a certain GOS for classes 1 and 2, which is specified in terms of loss probabilities and mean waiting times for classes 1 and 2 in the study, is the maximum total load without violating the GOS. At a given load, three different mixes of loads from classes 1 and 2 are tried. The three mixes are $\lambda_1 = \lambda_2$, $\lambda_1 = 2\lambda_2$ and $2\lambda_1 = \lambda_2$.

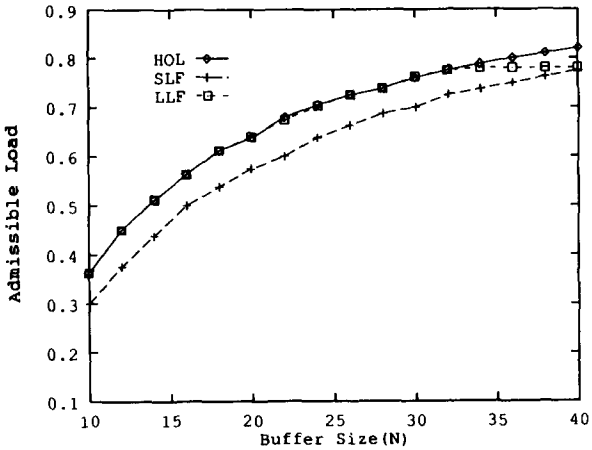


Fig. 3. Admissible load versus buffer size ($\lambda_1 = \lambda_2$).

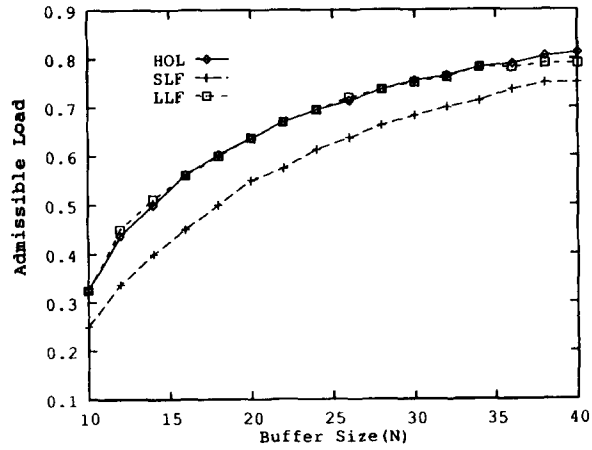


Fig. 4. Admissible load versus buffer size ($\lambda_1 = 2\lambda_2$).

The first set of examples (Figs. 3, 4 and 5), displays the relationship between admissible loads and the total buffer sizes. The same GOS are used in three figures with $\lambda_1 = \lambda_2$ in Fig. 3, $\lambda_1 = 2\lambda_2$ in Fig. 4 and $2\lambda_1 = \lambda_2$ in Fig. 5. The GOS is

$$l_1 \leqslant 10^{-10}, \qquad l_2 \leqslant 10^{-6}$$

$$\bar{w}_1 \leqslant 1.5, \qquad \bar{w}_2 \leqslant 5.$$

The admissible loads are represented on y-axes and the total buffer sizes N are on x-axes where $N = N_1$ is assumed. The three curves in each figure correspond to the three service disciplines.

As we can see, HOL administers the largest admissible loads with respect to the GOS used here. This is true not only for different buffer sizes but also for different load mixes. In Fig. 3, the limiting factor of the admissible load is the loss probability of class 2 in all three service disciplines. In Fig. 4, where $\lambda_1 = 2\lambda_2$, the limiting factor differs with the service discipline. For HOL, the limiting factor is the loss probability of class 1 when N, the total buffer size, is less then or equal to 20 and the loss probability of class 2 when $N > 20$. However, at $N = 40$ both the loss probability and the mean waiting time of class 2 approach the GOS limit simultaneously. For SLF, the limiting factor is the loss probability of class 1 when $N \leqslant 38$ and the mean waiting time of class 1 when $N = 40$. For LLF, the limiting factor is the loss probability of class 1 when $N \leqslant 12$, the loss probability of class 2 when $12 < N \leqslant 32$ and the mean waiting time of class 1 when $N \geqslant 34$. In Fig. 5, where $2\lambda_1 = \lambda_2$, the limiting factor is the loss probability of class 2 for HOL, the loss probability of class 1 for SLF, the loss probability of class 2 when $N \leqslant 26$ and the mean waiting time of class 1 when $N \geqslant 28$ for LLF.

The second set of examples (Figs. 6 to 11), shows how loss probabilities and mean waiting times of two classes vary with the total load. Again, three service disciplines and three load mixes are used. In all these examples, $N = N_1 = 40$ is assumed and the total load changes from 0.05 to 0.95. Figs. 6 to 8 are curves of loss probabilities versus total load with $\lambda_1 = \lambda_2$ in Fig. 6, $\lambda_1 = 2\lambda_2$ in Fig. 7 and $2\lambda_1 = \lambda_2$ in Fig. 8. Figs. 9 to 11 are curves of mean waiting times of mean waiting times versus total load with $\lambda_1 = \lambda_2$ in Fig. 9, $\lambda_1 = 2\lambda_2$ in Fig. 10 and $2\lambda_1 = \lambda_2$ in Fig. 11.

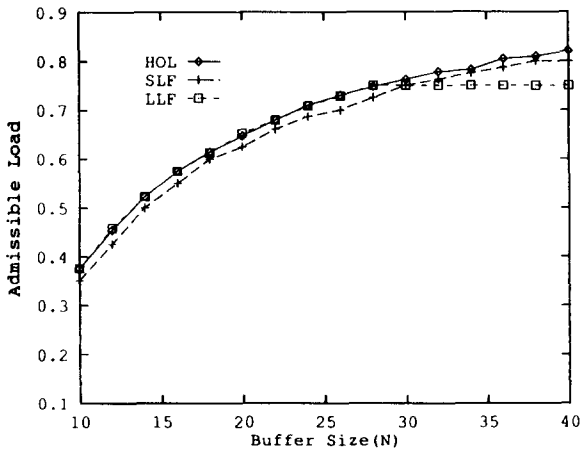


Fig. 5. Admissible load versus buffer size ($2\lambda_1 = \lambda_2$).

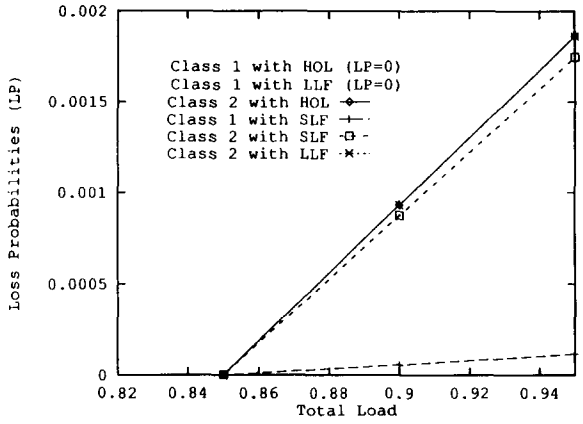


Fig. 6. Load versus loss probability ($N = N_1 = 40$, $\lambda_1 = \lambda_2$).

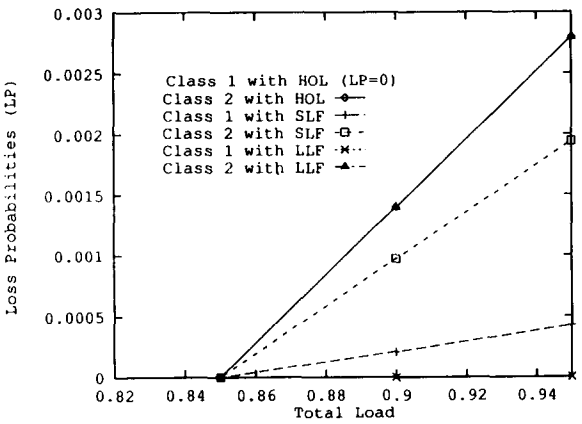


Fig. 7. Load versus loss probability ($N = N_1 = 40$, $\lambda_1 = 2\lambda_2$).

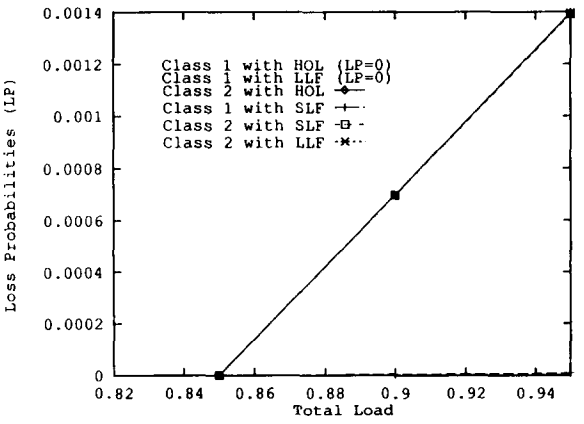


Fig. 8. Load versus loss probability ($N = N_1 = 40$, $2\lambda_1 = \lambda_2$).

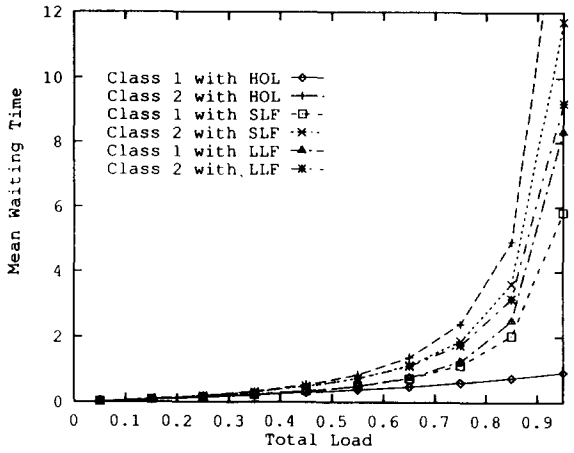


Fig. 9. Load versus mean waiting time ($N = N_1 = 40$, $\lambda_1 = \lambda_2$).

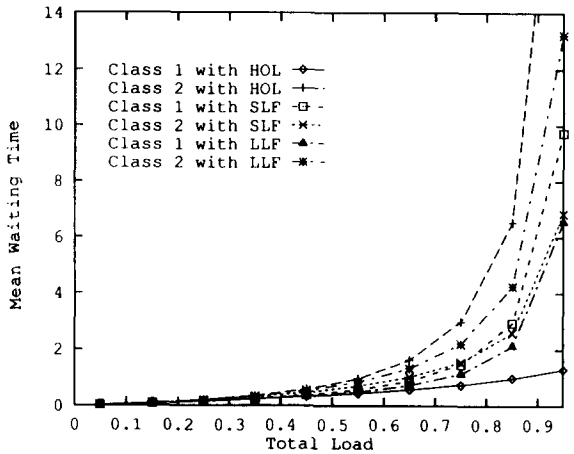


Fig. 10. Load versus mean waiting time ($N = N_1 = 40$, $\lambda_1 = 2\lambda_2$).

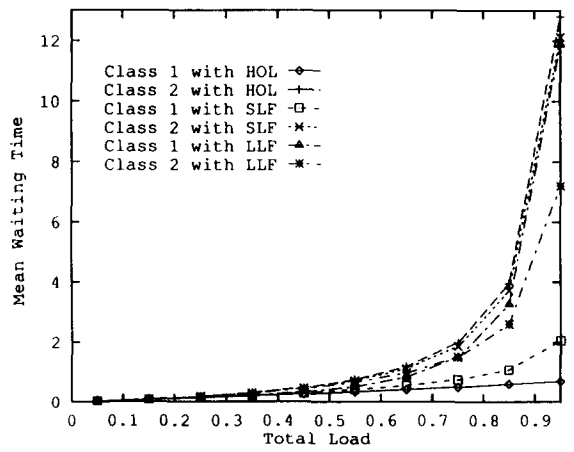


Fig. 11. Load versus mean waiting time ($N = N_1 = 40$, $2\lambda_1 = \lambda_2$).

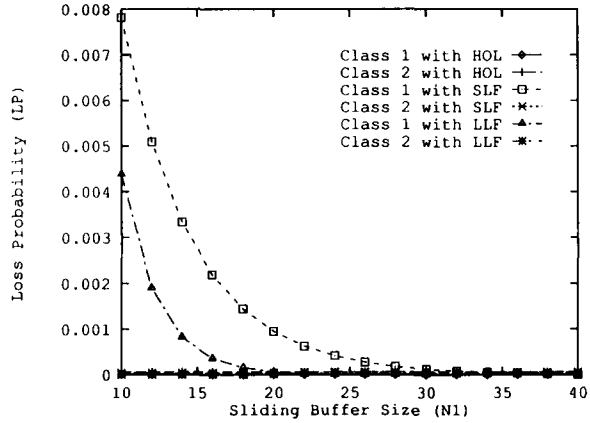


Fig. 12. Sliding buffer size (N_1) versus loss probabilities ($N = 40$, $\lambda_1 = \lambda_2$, total load = 0.9).

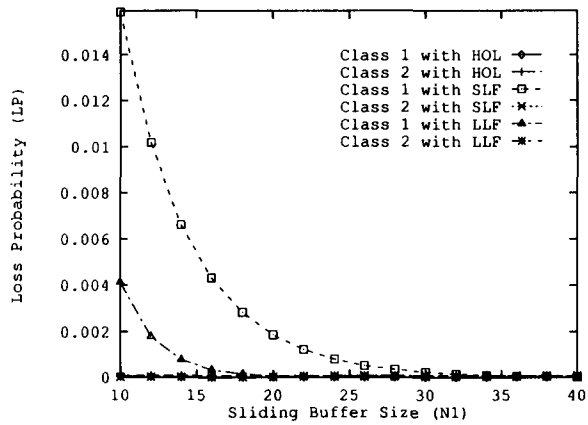


Fig. 13. Sliding buffer size (N_1) versus loss probabilities ($N = 40$, $\lambda_1 = 2\lambda_2$, total load = 0.9).

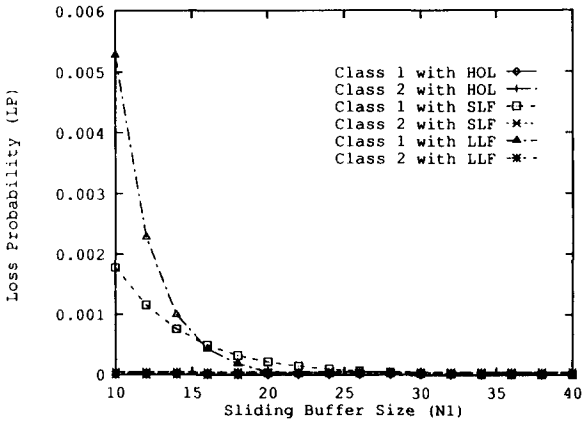


Fig. 14. Sliding buffer size (N_1) versus loss probabilities ($N = 40$, $2\lambda_1 = \lambda_2$, total load = 0.9).

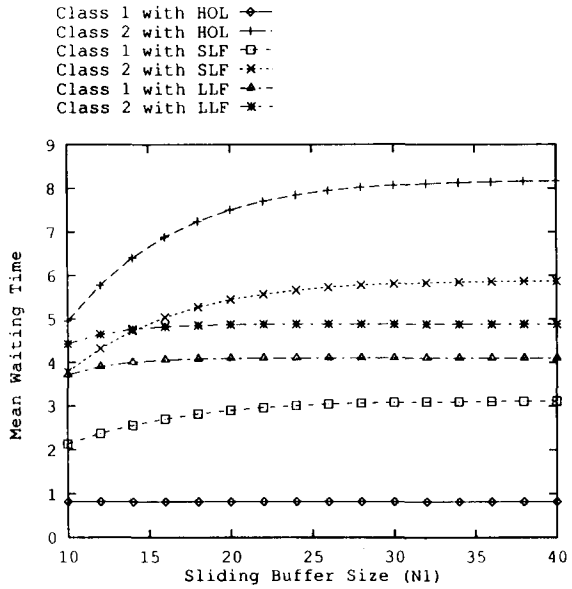


Fig. 15. Sliding buffer size (N_1) versus mean waiting time ($N = 40$, $\lambda_1 = \lambda_2$, total load = 0.9).

The loss probabilities of LLF surprisingly resemble the loss probabilities of HOL in all three figures. On the other hand, the mean waiting times of HOL and LLF are in opposite directions. HOL tends to minimize the mean waiting time of class 1 and maximize the mean waiting time of class 2 while LLF holds also in the next set of numerical examples when $N \approx N_1$ and can be explained intuitively. It seems that loss probabilities and mean waiting times of HOL are least

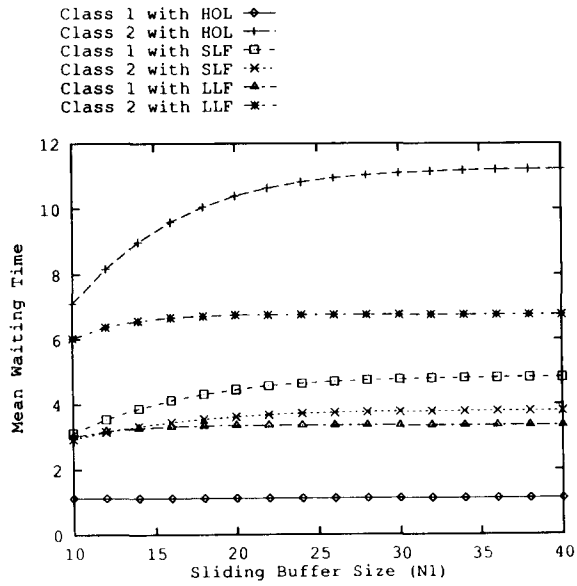


Fig. 16. Sliding buffer size (N_1) versus mean waiting time ($N = 40$, $\lambda_1 = 2\lambda_2$, total load = 0.9).

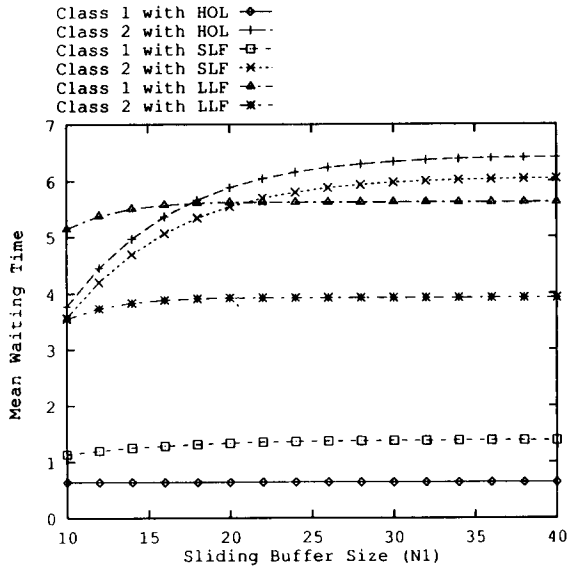


Fig. 17. Sliding buffer size (N_1) versus mean waiting time ($N = 40$, $2\lambda_1 = \lambda_2$, total load = 0.9).

sensitive to the change of the ratio of λ_1 and λ_2 for a given total load, while LLF and SLF are more and most sensitive.

The last set of example (Figs. 12 to 17), shows changes of loss probabilities and mean waiting times of two classes with the increase of N_1. In all these examples, $N = 40$ and $\rho = 0.9$ are assumed. Figs. 12 to 14 are curves of loss probabilities versus N_1 with $\lambda_1 = \lambda_2$ in Fig 12, $\lambda_1 = 2\lambda_2$ in Fig. 13 and $2\lambda_1 = \lambda_2$ in Fig. 14. Figs. 15 to 17 are curves of mean waiting times versus N_1 with $\lambda_1 = \lambda_2$ in Fig. 15, $\lambda_1 = 2\lambda_2$ in Fig. 16 and $2\lambda_1 = \lambda_2$ in Fig. 17.

These examples show that once N_1 surpasses certain value, it no longer significantly affects the loss probabilities and mean waiting times.

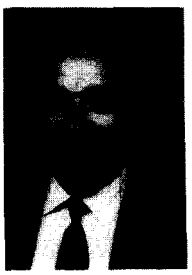
9. Conclusions

In this work we analyzed a queueing model M_1, M_2/G_1, G_2/N with different scheduling and pushout schemes. Our work can be used to evaluate the performance of an output link of ATM switches with traffic of two classes with different priorities. and may also have other applications in computer and communications systems.

By introducing $\alpha_1(i, j)$, a service discipline function, we were able to consider various scheduling disciplines such as HOL, SLF, LLF and Random Scheduling. By dividing the total buffer space into two parts, we created a push-out scheme that permits a controlled share of the buffer space between the two classes. We gave an exact solution for the loss probabilities of both classes, the queue length distribution and the mean waiting time for class 1. An approximate solution for the queue length distribution and the mean waiting time for class 2 were also obtained. We gave a set of numerical examples which considered the loss probabilities and mean waiting time simultaneously. It remains to extend these results to cases of bursty arrivals.

References

[1] CCITT Recommendation I.121, On the Broadband Aspects of ISDN (*CCITT Blue Book*, Geneva, 1989).
[2] CCITT Draft Recommendation I.361, ATM Layer Specification for B-ISDN (*Study Group XVIII*, Geneva, January, 1990).
[3] B.T. Doshi and H. Heffes, Overload performance of several processor queueing disciplines for the M/M/1 queue, *IEEE Trans. Comm.* **34** (6) (1986) 538–546.
[4] S. Sumita and T. Ozawa, Achievability of performance objectives in ATM switching nodes, *Proc. Int. Seminar on Performance of Distributed and Parallel Systems*, Kyoto, 1988, pp. 45–56.
[5] S. Sumita, Synthesis of an output buffer management schemes in a switching system for multimedia communications, *Proc. IEEE INFOCOM'90*, San Francisco, June 1990, pp. 1226–1233.
[6] G. Hebuterne and A. Gravey, A space priority queueing mechanism for multiplexing ATM channels, *Computer Network and ISDN Systems* **20** (1990) 37–43.
[7] H. Kroner, Comparative performance study of space priority mechanisms for ATM networks, *Proc. IEEE INFOCOM'90*, San Francisco, June 1990, pp. 1136–1143.
[8] T. Takenaka, Buffer management schemes for a heterogeneous packet switching system, *Trans. IECE Japan* **J67-B** (6) (1984) 505–512 (in Japanese).
[9] R.W. Wolff, Poisson arrivals see time averages, *Oper. Res.* **30** (1982) 223–231.
[10] H. Saito, Queueing analysis of a cell loss probability control in ATM networks, *Proc. Int. Teletraffic Congress*, Copenhagen, 1991.
[11] D.P. Heyman, The push-out priority queue discipline, *Oper. Res.* **33** (1985) 397–403.
[12] A. Gravey and G. Hebuterne, Mixing time and loss priorities in a single server queue, *ITC-13*, Copenhagen, June 1991
[13] H. Kröner, G. Hébuterne, P. Boyer and A. Gravey, Priority management in ATM switching nodes, *IEEE J. Selected Areas Comm.* (April 1991).
[14] N. Mitrou and D. Pendarakis, Cell-level statistical multiplexing in ATM networks analysis, dimensioning and call-acceptance control w.r.t. QoS criteria, *ITC-13*, Copenhagen, June 1991.



Ian F. Akyildiz received his BS, MS, and PhD degrees in Computer Engineering from the University of Erlangen-Nürnberg, Germany, in 1978, 1981 and 1981 and 1984, respectively. Currently, he is an Associate Professor with the School of Electrical and Computer Engineering, Georgia Institute of Technology. He has held visiting professorships at the Universidad Tecnica Federico Santa Maria, Chile, Université Pierre et Marie Curie (Paris VI) and Ecole Nationale Superieure Telecommunications in Paris, France. He has published over eighty technical papers in journals and conference proceedings. He is a co-author of a textbook entitled "Analysis of Computer Systems" published by Teubner Verlag in Germany in 1982. He is an associate editor for *Computer Networks and ISDN Systems* and an editor for *IEEE Transactions on Computers*, and *Journal of Wireless Networks*.

He guest-edited several special issues, such as on "Parallel and Distributed Simulation Performance" for *ACM Transactions on Modeling and Simulation*; on "Teletraffic Issues in ATM Networks" for *Computer Networks and ISDN Systems* and on "Networks in the Metropolitan Area" for *IEEE Journal of Selected Areas in Communications*.

Dr. Akyildiz is a senior member of IEEE since 1989, member of ACM (SIGCOMM, SIGMETRICS, SIGOPS) and is a National Lecturer for ACM since 1989. He received the "Don Federico Santa Maria Medal" for his services to the Universidad of Federico Santa Maria in Chile. Dr. Akyildiz is listed on "Who's Who in the World (Platinum Edition)".

His current research interests are in B-ISDN (ATM) networks, optical local area networks, performance evaluation, parallel simulation, high performance computing, wireless networks and network security.

Xian Cheng received the B.S. degree in computer science and engineering from Zhejiang University, China, in 1982, and the M.S. and Ph.D. degrees in computer science from Georgia Institute of Technology in 1984 and 1987, respectively. From 1987 to 1991, he worked as a software engineer and research scientist in Aeronomics Inc., Atlanta, in the area of yield management for airline and hotel industries. He is currently a member of technical staff at AT&T Bell Laboratories, working on the performance of computer hardware and software system. His research interests include queueing systems, communications networks and distributed systems.