Tight upper bounds for cell loss probabilities and required bandwidth estimation in ATM multiplxers

Xiaoming Liu[1], Brahim Bensaou[2], Danny H. K. Tsang[3], Liming Liu[1]

[1] Department of Industrial Engineering & Engineering Management
[3] Department of Electrical & Electronic Engineering
Hong Kong University of Science & Technology,

[2] Centre for Wireless Communications
National University of Singapore

Abstract

Estimating the cell loss probability in an ATM multiplexer is a key issue in network management and traffic control tasks such as call admission control and bandwidth allocation. In this paper, we derive a new approximation to estimate the total and individual cell loss probabilities in an ATM multiplexer fed by a superposition of heterogeneous on-off sources. Based on this approximation, a simple algorithm is proposed to estimate accurately the aggregate required bandwidth to guarantee a given cell loss probability. Numerical results and comparisons to alternative approaches are also shown.

Supported partly by Hongkong Telecom Institute of Information Technology grant HKTIIT93/94.EG01

This is the Pre-Published Version

I. INTRODUCTION

Due to its ability to integrate different types of traffic with different characteristics and quality of service (QoS) requirements, ATM has been regarded as the desirable transfer mode in the future broadband integrated service digital network (B-ISDN). Generally, QoS can be expressed in terms of cell delay, cell delay jitter and cell loss ratio (CLR) in the multiplexers. The delay can be bounded by limiting the buffer size. Knowing the buffer size, the CLR can be bounded by allocating the adequate bandwidth and implementing the appropriate traffic control mechanisms such as traffic policing and call admission control.

Usually, buffer dimensioning and bandwidth allocation rely on a given model of the ATM multiplexer. An ATM multiplexer is widely modelled as a single deterministic server queue, fed by a superposition of independent on/off sources. The actual arrival process is approximated by different simpler and analytically tractable models such as, stochastic fluid flow (SFF), Markov modulated Poisson process (MMPP), or Markov modulated deterministic process (MMDP). Among these models, the SFF approximation [1] and the MMDP approach [2] prove particularly appealing to represent accurately the congestion that occurs at the so-called burst scale. In the SFF approach, the input and the service processes are assumed to be continuous rather than discrete, and the buffer size is assumed to be infinite. In [1], closed form formulae for the overflow probability as well as the asymptotic behavior for an infinitely large queue are obtained for a superposition of homogeneous on/off sources. The extension of these results to the superposition of heterogeneous traffic sources is presented in [3], [4]. However, in this latter case the solution is no longer analytical. Besides, it requires a large amount of computations. Based on the work in [1] giving the overflow probability (infinite buffer) in the SFF method, an estimate of the cell loss probability (finite buffer) has been given in [5].

The MMDP approach approximates the real input process by a deterministic process whose rate is controlled by a Markov chain [2], [6]. This model is similar to the SFF approach in the sense that it addresses only the burst scale component of the cell loss probability in the ATM multiplexer. In this approach, the buffer is finite and information is treated as discrete cells rather than continuous fluid flow. The cell loss probability is estimated as the ratio of the expected number of cells lost to the expected total number of cells arrived in a period, by solving a classic Markovian problem through standard numerical solutions (Gauss-Seidel or Jacobi algorithms). The cell loss probability is obtained for the superposition of homogeneous on/off sources in [2] and this result is extended to the aggregate and individual cell loss probabilities for a superposition of heterogeneous on/off sources in [6].

In theory, the most accurate approach to determine the required bandwidth, given the buffer size in the ATM multiplexer, is to invert the exact formula for the cell loss probability as a function of the server capacity. However, in practice, the exact formulae are too complex and always require a laborious amount of computations. For this reason, many simpler approximations of the cell loss probability were proposed in the literature. Most of these approximations are based on the asymptotic equivalent of the overflow probability provided by the SFF approach [1], [4]. The basic idea in these approximations is to express the survivor function of the buffer occupancy as $G(x) = \Pr(Y > x) \approx A(C)e^{z_0(C)x}$, where $z_0(C)$ and $A(C)$ are functions of the server capacity and the traffic parameters. For a fixed C, in a logarithmic scale, the approximation of $G(.)$ is a linearly decreasing function whose slope is determined by $z_0(C)$ and the intercept on the ordinates axis (for $x = 0$) is governed by the constant $A(C)$. A closed form solution for $z_0(C)$ in the case of a superposition of homogeneous exponentially distributed on/off sources is given in [1]. For the heterogeneous traffic sources case, as shown in [4], $z_0(C)$ can be obtained as the largest negative solution of a non linear equation, which can be solved numerically. The determination of the constant $A(C)$ proves, however, much more difficult. Many approximations of $A(C)$ have been proposed [1], [7], [8], [9]. The tighter the approximation of $A(C)$ to the intercept of the exact cell loss probability, the better is the approximation of $G(x)$, and thus of the required bandwidth.

In this paper, we address the determination of a new tight approximation of the intercept

$A(C)$. This approximation is determined from our previous work on the MMDP approach to estimate the cell loss probability in an ATM multiplexer. Although this approximation is tight, it can sometimes require long computation times, especially when the number of sources is large. We propose thus an approximate method to evaluate $A(C)$ in terms of its generating function. Finally, we apply our new approximation to estimate the aggregate required bandwidth.

The remainder of this paper is organized as follows. In the next section, we present briefly the previous results obtained in the estimation of the intercept. We, then, introduce the MMDP approach and derive our main result in Section III. The approximate method to estimate efficiently and quickly the value of the intercept in terms of its generating function is also presented. In Section IV the approximation is applied to the determination of the required bandwidth. Finally, to show the accuracy and efficiency of our approximation, numerical results are presented in Section V.

II. Previous work on the estimation of $A(C)$

We consider an ATM multiplexer serving a superposition of L independent classes of on/off sources. The multiplexer is modelled as a buffer of size K cells served by a single output link with deterministic service capacity C. For each class i, $i = 1, 2, ..., L$, there are M_i identical and independent sources. Each source can be in one of two states: on and off. When the source is on, it generates a stream of cells at a fixed rate Δ_i, when it is off, no cells are generated. Both on and off periods are assumed to be independent exponentially distributed random variables with mean $1/\mu_i$ and $1/\lambda_i$, respectively. When a cell arrives, it is served immediately if the output link is idle, otherwise, it is buffered if the buffer is not full. Cells are dropped (lost) when the buffer is full. The cells are served on a FIFO basis.

Let $N_i(t)$ be the number of class i, $i = 1, \ldots, L$, sources *on* at time t, and denote by $Y(t)$ the buffer content at time t. $Y(t)$ is not Markovian but the joint process $\{\mathbf{N}(t), Y(t)\}$ with $\mathbf{N}(t) = (N_1(t), N_2(t), \ldots, N_L(t))$ is Markovian. The system can thus be solved for the joint stationary probability, and the stationary distribution of the buffer content can be obtained from the marginal distribution of $\{Y(t)\}$. Before introducing our new approximation, in the next section, we briefly present the most common results known in the estimation of $A(C)$.

For the sake of simplicity, we present only the case of homogeneous sources. The input rate to the multiplexer queue is determined by the state of the finite dimensional Markov process $N(t)$ representing the number of sources *on* at time t. Let M be the total number of sources and r_i the input rate in state i, $i = 0, \cdots, M$. Denote by $\mathbf{T}$ the infinitesimal generator of $N(t)$: T_{ij} is the transition rate from state i to state j, and $T_{ii} = -\sum_{j, j \neq i} T_{ij}$. Let $\pi_i(x)$ be the steady state probability that the rate process $N(t)$ is in state i and the buffer content $Y(t)$ is not larger than x. The system of differential equations governing the dynamics of the joint process $\{N(t), Y(t)\}$ is then

$$\frac{d}{dx}\pi(x)\mathbf{D} = \pi(x)\mathbf{T}, \tag{1}$$

where $\mathbf{D} \triangleq diag\{r_i - C\}$ is the drift matrix and $\pi(x) = \{\pi_0(x), \cdots, \pi_M(x)\}$. By solving this system, the stationary buffer overflow probability $G(x)$ is given by

$$\begin{aligned} G(x) &= 1 - \langle \pi(x), \mathbf{1} \rangle \\ &= \sum_{i \geq 0} a_i \langle \phi_i, \mathbf{1} \rangle e^{z_i x}, \end{aligned} \tag{2}$$

where a_i are real coefficients determined from the boundary conditions [1], z_i are eigenvalues with negative real part, and ϕ_i are the associated eigenvectors. The pairs (z_i, ϕ_i) are solutions of the eigensystem

$$z_i \phi_i \mathbf{D} = \phi_i \mathbf{T}. \tag{3}$$

As mentioned in [10], this apparently simple solution requires sometimes a large amount of computations. Especially, the calculation of the coefficients a_i can lead to severe numerical problems. These problems, in addition to the need of developing effective solutions that can be used for real-time network management such as resources allocation and routing, lead to the development of a large number of approximations of the result in (2). Among them, the first one assumes an infinitely large buffer size, so that the sum in (2) is dominated by a leading term obtained for the largest eigenvalue say z_0 [1] giving

$$G(x) \sim a_0 \langle \phi_0, \mathbf{1} \rangle e^{z_0 x}, \text{ for } x \to +\infty. \tag{4}$$

This asymptotic approximation has two major drawbacks: the first one is that to calculate the coefficient a_0 it is necessary to calculate all the stable eigenvalues z_i. The second one, by far more severe, is that approximation (4) always underestimates the exact result. Moreover, it has been shown in [7] that under certain traffic conditions, the difference between the exact result and the approximation is very large even when the buffer size is (unrealistically) very large. In [9], an upper bound to (4) was derived by simply replacing the factor $a_0\langle \phi_0, \mathbf{1}\rangle$ by 1, leading to

$$G(x) \sim e^{z_0 x}. \tag{5}$$

Although this approximation simplifies substantially the computation of the bandwidth required for a given CLR target and buffer size, it often leads to an outrageous over-estimation of the required bandwidth under realistic traffic conditions. To overcome this problem, the authors introduced an improvement by taking the minimum of this approximation and the Gaussian approximation of the probability that the input rate exceeds the output channel capacity (for details refer to the original paper [9] or see [11]). In the same spirit, a new approximation is proposed in [8]. The factor $a_0\langle \phi_0, \mathbf{1}\rangle$ is replaced by a new factor L_0 obtained from a bufferless model by using large deviation theory, leading to

$$G(x) \sim L_0 e^{z_0 x}. \tag{6}$$

L_0 is actually the Chernoff's large deviation approximation to the probability that the instantaneous input rate exceeds the channel capacity [12]. This factor L_0 is very accurate, however, it constitutes an approximation to the overflow probability instead of an upper bound so that when the buffer size is relatively small, this approximation may underestimate the exact overflow probability by a theoretically unknown amount.

III. A new approximation of the intercept $A(C)$

A. *The MMDP approximation*

In the MMDP approximation, the discrete nature of the arrival and service processes is conserved and the buffer size is assumed to be finite. In the sequel, we will focus on the superposition of heterogeneous traffic sources. Let M_i be the total number of sources of class i, $i = 1, ..., L$. As defined above, the process $\{\mathbf{N}(t)\}$ defines a finite, irreducible, continuous-time Markov process with state space

$$\mathbf{S} = \{\mathbf{n} : \mathbf{n} = (n_1, n_2, ..., n_L), \ \ n_i = 0, 1, ..., M_i, i = 1, ..., L\}$$

and infinitesimal generator $\mathbf{T}$. Its stationary distribution is given by

$$p_{\mathbf{n}} = \lim_{t\to+\infty} \Pr\{\mathbf{N}(t) = \mathbf{n}\} = \prod_{i=1}^{L} \left(\binom{M_i}{n_i} \tau_i^{n_i} (1-\tau_i)^{M_i - n_i} \right), \quad \mathbf{n} \in \mathbf{S}, \tag{7}$$

where $\tau_i = \lambda_i/(\mu_i + \lambda_i)$.

Let ξ_k be the k^{th} transition epoch of $\mathbf{N}(t)$, $\mathbf{N}_k = \mathbf{N}(\xi_k)$, $Y_k = Y(\xi_k)$. Note that, in the MMDP approach, the process $\{Y_k\}$ denotes the content of the system including the cell being served. We assume arriving cells are equally spaced during $[\xi_k, \xi_{k+1})$ and if a cell is being transmitted at ξ_k it will be retransmitted immediately after. Then $\{\mathbf{N}_k\}$ and $\{(\mathbf{N}_k, Y_k)\}$ are embedded finite, irreducible Markov chains of $\{\mathbf{N}(t)\}$ and $\{(\mathbf{N}(t), Y(t))\}$ respectively. The Markov chain $\{\mathbf{N}_k\}$

has the stationary distribution $p^*_{\mathbf{n}} = \lim_{k\to\infty} \Pr\{\mathbf{N}(\xi_k) = \mathbf{n}\}$. Denote by $\pi_{(\mathbf{n},l)}$ the stationary distribution of the joint process $\{(\mathbf{N}_k, Y_k)\}$:

$$\pi_{(\mathbf{n},l)} = \lim_{k\to+\infty} \Pr\{\mathbf{N}_k = \mathbf{n}, Y_k = l\}, \quad \mathbf{n} \in \mathbf{S} \text{ and } l = 0, 1, \cdots, K+1.$$

Let $\pi_{\mathbf{n}} = (\pi_{(\mathbf{n},0)}, \pi_{(\mathbf{n},1)}, \cdots, \pi_{(\mathbf{n},K+1)})$ and $\omega_{\mathbf{n}} = (\omega_{(\mathbf{n},0)}, \omega_{(\mathbf{n},1)}, \cdots, \omega_{(\mathbf{n},K+1)}) = \pi_{\mathbf{n}}\mathbf{A}_{\mathbf{n}}$, where $\mathbf{A}_{\mathbf{n}} = \left(a^{\mathbf{n}}_{(l,h)}\right)$ is a $K+2 \times K+2$ matrix whose elements are given by

$$a^{\mathbf{n}}_{(l,h)} = \Pr\{Y_{k+1} = h \mid \mathbf{N}_k = \mathbf{n}, Y_k = l\}, \mathbf{n} \in \mathbf{S} \text{ and } l, h = 0, 1, \cdots, K+1.$$

Let $\mathbf{V}$ be the set of overload states:$\mathbf{V} = \{\mathbf{n} : \mathbf{n} \in \mathbf{S}, \Delta_{\mathbf{n}} > C\}$, where $\Delta_{\mathbf{n}} = \sum_{i=1}^{L} n_i \Delta_i$. The total cell loss probability $P_{loss}(C, K)$ for a given buffer size K and channel speed C is [6]

$$P_{loss}(C,K) = \frac{\sum_{\mathbf{n}\in\mathbf{V}} \left(\frac{\exp\{-\gamma_{\mathbf{n}}/(\Delta_{\mathbf{n}} - C)\}}{1 - \exp\{-\gamma_{\mathbf{n}}/(\Delta_{\mathbf{n}} - C)\}} \omega_{\mathbf{n},K+1} \right)}{\sum_{\mathbf{n}\in\mathbf{S}} \frac{\Delta_{\mathbf{n}}}{\gamma_{\mathbf{n}}} p^*_{\mathbf{n}}} \tag{8}$$

and the individual cell loss probability for class i is given by

$$P^i_{loss}(C,K) = \frac{\sum_{\mathbf{n}\in\mathbf{V}} \left(\frac{\exp\{-\gamma_{\mathbf{n}}/(\Delta_{\mathbf{n}} - C)\}}{1 - \exp\{-\gamma_{\mathbf{n}}/(\Delta_{\mathbf{n}} - C)\}} \frac{n_i\Delta_i}{\Delta_{\mathbf{n}}} \omega_{\mathbf{n},K+1} \right)}{\sum_{\mathbf{n}\in\mathbf{S}} \frac{n_i\Delta_i}{\gamma_{\mathbf{n}}} p^*_{\mathbf{n}}}, \quad i = 1, \cdots, L, \tag{9}$$

with $\gamma_{\mathbf{n}} = \sum_{i=1}^{L} [(M_i - n_i)\lambda_i + n_i\mu_i]$.

As shown by the results in the original papers [2], [6], these formulae are very accurate and the computation time is quite small when the buffer size K and/or the total number of sources are not too large. However, the calculation of $\omega_{\mathbf{n},K+1}$ becomes very difficult for relatively large sized problems.

B. A new approximation of $A(C)$

In [5] it is shown that, like the overflow probability, the cell loss probability behaves asymptotically as $A\exp\{z_0 x\}$, where x is the buffer size. Moreover, it is shown that the asymptotic slope z_0 of the cell loss probability in a logarithmic scale is governed by the same largest eigenvalue z_0 as the overflow probability, however the intercept $A(C)$ is smaller than the intercept of the overflow probability. Our main objective in this paper is to derive a new accurate closed form formula for the intercept of the cell loss probability, based on the following facts:

- when the buffer size is small the approximation should be exactly equal to the cell loss probability in a bufferless system;
- when the buffer size increases towards infinity, the slope of the cell loss probability (in log scale) becomes closer to z_0.

Based on these two facts, the approximation we propose for the cell loss probability in a system with buffer size K is of the form $P_{loss} \sim A(C)e^{z_0(C)K}$, where $A(C)$ is the cell loss probability in a bufferless model, obtained from (8) and z_0 is the largest eigenvalue obtained from (3). The determination of the eigenvalue z_0 has been the subject of thorough research, such as [1] [4] [13]. In the following, we will focus mainly on the determination and the evaluation of a more accurate approximation of the intercept $A(C)$. For more details on the determination of z_0 the reader is referred to [4].

The largest negative eigenvalue of the matrix $\mathbf{T}\mathbf{D}^{-1}$ in (3) can be determined as shown in [4] as the largest negative solution of the equation

$$\sum_{i=1}^{L} M_i g_i(z) = C, \tag{10}$$

where

$$g_i(z) = \frac{(\Delta_i z + \lambda_i + \mu_i) - \sqrt{(\Delta_i z + \lambda_i + \mu_i)^2 - 4\Delta_i \lambda_i z}}{2z}, \; i = 1, \cdots, L.$$

Particularly, when all the sources have the same characteristics, a closed form solution for z_0 is obtained in [1]

$$z_0 = -\frac{M\Delta(C\mu + \lambda C - M\Delta\lambda)}{C\mu(M\Delta - C)}.$$

To calculate the intercept $A(C)$, we have to notice that for a bufferless system ($K = 0$), in (8) and (9), $\omega_{\mathbf{n},1} = p^*_{\mathbf{n}}$. This can be obtained by first writing explicitly $\omega_{\mathbf{n},1}$ in terms of $\pi_{\mathbf{n},1}$ as $\omega_{\mathbf{n},1} = \pi_{\mathbf{n},1}$ since for all $\mathbf{n} \in \mathbf{V}$, $\pi_{\mathbf{n},0} = 0$ and second by noting that for a bufferless system, using the total probability formula, we have $p^*_{\mathbf{n}} = \pi_{\mathbf{n},0} + \pi_{\mathbf{n},1}$. Noting again that for all $\mathbf{n} \in \mathbf{V}$, $\pi_{\mathbf{n},0} = 0$, we deduce $\omega_{\mathbf{n},1} = p^*_{\mathbf{n}}$. Hence, from (8) the exact intercept for the overall cell loss probability is given by

$$A(C) = P_{loss}(C, 0) = \frac{\sum_{\mathbf{n}\in\mathbf{V}} \left(\frac{\exp\{-\gamma_{\mathbf{n}}/(\Delta_{\mathbf{n}} - C)\}}{1 - \exp\{-\gamma_{\mathbf{n}}/(\Delta_{\mathbf{n}} - C)\}} p^*_{\mathbf{n}} \right)}{\sum_{\mathbf{n}\in\mathbf{S}} \frac{\Delta_{\mathbf{n}}}{\gamma_{\mathbf{n}}} p^*_{\mathbf{n}}} \tag{11}$$

and from (9) the exact intercept for the individual cell loss probability is

$$A_i(C) = P^i_{loss}(C, 0) = \frac{\sum_{\mathbf{n}\in\mathbf{V}} \left(\frac{\exp\{-\gamma_{\mathbf{n}}/(\Delta_{\mathbf{n}} - C)\}}{1 - \exp\{-\gamma_{\mathbf{n}}/(\Delta_{\mathbf{n}} - C)\}} \frac{n_i \Delta_i}{\Delta_{\mathbf{n}}} p^*_{\mathbf{n}} \right)}{\sum_{\mathbf{n}\in\mathbf{S}} \frac{n_i \Delta_i}{\gamma_{\mathbf{n}}} p^*_{\mathbf{n}}}, \quad i = 1, \cdots, L. \tag{12}$$

Assume now that $\Delta_{\mathbf{n}} - C \gg \gamma_{\mathbf{n}}$; this can be justified intuitively by the fact that, first the units of $\Delta_{\mathbf{n}} - C$ are in cells per second, so that when the system is in an overload state, $\mathbf{n} \in \mathbf{V}$, this quantity is large; second, the system we are considering takes into account the fluctuations of the traffic only in the burst scale and neglects the cell scale, in other words, these traffic fluctuations occur in a larger time scale than the cells interarrival time, so that $1/\gamma_{\mathbf{n}}$ is a large number and thus $\gamma_{\mathbf{n}}$ is a small quantity. With this reasonably accurate assumption, $\gamma_{\mathbf{n}}/(\Delta_{\mathbf{n}} - C) \to 0$, and the Taylor series expansion can be invoked to approximate the exponential terms in (11). Particularly, when these exponential terms are replaced by their second order Taylor approximation around 0, Equation (11) becomes

$$A(C) \sim \frac{\sum_{\mathbf{n}\in\mathbf{V}} \left(\frac{\Delta_{\mathbf{n}} - C}{\gamma_{\mathbf{n}}} p^*_{\mathbf{n}} \right)}{\sum_{\mathbf{n}\in\mathbf{S}} \left(\frac{\Delta_{\mathbf{n}}}{\gamma_{\mathbf{n}}} p^*_{\mathbf{n}} \right)} \tag{13}$$

Noting that the stationary state probability $p^*_{\mathbf{n}}$ of the embedded Markov chain $\{N(k)\}$ is related to the stationary state probability $p_{\mathbf{n}}$ of the Markov process $\{N(t)\}$ by

$$p_{\mathbf{n}} = \frac{p^*_{\mathbf{n}}/\gamma_{\mathbf{n}}}{\sum_{\mathbf{m}\in\mathbf{S}} p^*_{\mathbf{m}}/\gamma_{\mathbf{m}}}, \tag{14}$$

the final value for the intercept in (13) becomes

$$A(C) \sim \frac{\sum_{\mathbf{n}\in\mathbf{V}} (\Delta_{\mathbf{n}} - C)\, p_{\mathbf{n}}}{\sum_{\mathbf{n}\in\mathbf{S}} \Delta_{\mathbf{n}} p_{\mathbf{n}}} = \frac{\sum_{\mathbf{n}\in\mathbf{V}} (\Delta_{\mathbf{n}} - C)\, p_{\mathbf{n}}}{\sum_{i=1}^{L} M_i \Delta_i \tau_i}. \tag{15}$$

Applying the same assumptions and approximations above to the intercept of the individual cell loss probability in (12), we obtain

$$A_i(C) = P^i_{loss}(C, 0) \sim \frac{\sum_{\mathbf{n}\in\mathbf{V}} (\Delta_{\mathbf{n}} - C) p_{\mathbf{n}} \frac{n_i \Delta_i}{\Delta_{\mathbf{n}}}}{M_i \Delta_i \tau_i}. \tag{16}$$

The new approximations to the aggregate and individual cell loss probabilities we propose are then

$$P_{loss} \sim \frac{\sum_{\mathbf{n}\in\mathbf{V}} (\Delta_{\mathbf{n}} - C)\, p_{\mathbf{n}}}{\sum_{i=1}^{L} M_i \Delta_i \tau_i} e^{z_0 K}, \tag{17}$$

and

$$P^i_{loss} \sim \frac{\sum_{\mathbf{n}\in\mathbf{V}} (\Delta_{\mathbf{n}} - C) p_{\mathbf{n}} \frac{n_i \Delta_i}{\Delta_{\mathbf{n}}}}{M_i \Delta_i \tau_i} e^{z_0 K}, i = 1, \cdots, L. \tag{18}$$

C. Numerical evaluation of the intercept $A(C)$

In Equations (15) and (16), it is obvious that when the number of sources is very large, the computation of the intercept becomes time consuming because the intercept is defined as a convolution. Especially when the source peak rates are not prime numbers, the number of different combinations of sources leading to an overload state becomes very large. In the following, we propose to approximate the intercept by an upper bound which can be expressed in terms of the generating function of the intercept as a function of the capacity. The generating function is then inverted numerically by using the method proposed in [14].

Let $A(C)$ be the intercept defined in (15), and define a sequence $\{A^*_k\}_{k\in\mathbb{N}}$ such that, for every C, there exists an integer k with $A(C) \approx A^*_k$. The sequence $\{A^*_k\}_{k\in\mathbb{N}}$ can be constructed by writing

$$\begin{aligned} A(C) &= \frac{\sum_{\Delta_{\mathbf{n}} > C} (\Delta_{\mathbf{n}} - C)\, p_{\mathbf{n}}}{\sum_{i=1}^{L} M_i \Delta_i \tau_i} \\ &= \frac{\sum_{\Delta_{\mathbf{n}} > \lfloor C \rfloor} (\Delta_{\mathbf{n}} - \lfloor C \rfloor)\, p_{\mathbf{n}}}{\sum_{i=1}^{L} M_i \Delta_i \tau_i} - \frac{\sum_{\Delta_{\mathbf{n}} > \lfloor C \rfloor} (C - \lfloor C \rfloor)\, p_{\mathbf{n}}}{\sum_{i=1}^{L} M_i \Delta_i \tau_i}, \end{aligned}$$

and noting that the second term of the right hand side of the above equation is negligible compared to the first term. We obtain the following definition of the sequence of numbers $\{A^*_k\}$

$$A(C) \approx \frac{\sum_{\Delta_{\mathbf{n}} > \lfloor C \rfloor} (\Delta_{\mathbf{n}} - \lfloor C \rfloor)\, p_{\mathbf{n}}}{\sum_{i=1}^{L} M_i \Delta_i \tau_i} \triangleq A^*_{\lfloor C \rfloor}. \tag{19}$$

So far, we have transformed the computation of the function $A(C)$ to its lattice approximation A^*_k. We have to notice, however, two facts: first, the term neglected in discretizing the function A into the stepwise function A^* is indeed negligible, which makes the approximation accurate;

and second, since the term neglected is positive, this makes A^* an upper bound of A, and thus prevents the approximation from underestimating the cell loss probability. We nevertheless need to compute A_k^*. For this purpose, let $G(z)$ be the generating function of the sequence $\{A_k^*\}_{k\geq 0}$

$$\begin{aligned} G(z) &= \sum_{k=0}^{\infty} A_k^* z^k, \\ &= \frac{1}{\sum_{i=1}^{L} M_i \Delta_i \tau_i} \sum_{k=0}^{\infty} \left(\sum_{\Delta_{\mathbf{n}}>k} (\Delta_{\mathbf{n}} - k)\, p_{\mathbf{n}} \right) z^k, \end{aligned}$$

which may be rewritten as

$$G(z) = \frac{1}{\sum_{i=1}^{L} M_i \Delta_i \tau_i} \sum_{\mathbf{n}\neq\mathbf{0}} p_{\mathbf{n}} \sum_{k=0}^{\Delta_{\mathbf{n}}-1} (\Delta_{\mathbf{n}} - k)\, z^k.$$

By separating the terms inside the inner summation, and noting that $\Delta_{\mathbf{n}} = 0$ for $\mathbf{n} = 0$ we can find readily that

$$G(z) = \frac{1}{\sum_{i=1}^{L} M_i \Delta_i \tau_i} \sum_{\text{all } \mathbf{n}} p_{\mathbf{n}} \left(\frac{\Delta_{\mathbf{n}}}{1-z} - \frac{z}{(1-z)^2} + \frac{z^{\Delta_{\mathbf{n}}+1}}{(1-z)^2} \right).$$

After evaluation of the sum above, $G(z)$ can be written as

$$G(z) = \frac{1}{1-z} \left(1 - \frac{z \left(1 - \prod_{i=1}^{L} (1 + \tau_i (z^{\Delta_i} - 1)^{M_i}) \right)}{(1-z) \sum_{i=1}^{L} M_i \Delta_i \tau_i} \right). \tag{20}$$

To invert the generating function $G(.)$, we invoke a useful result proved in [14]. According to [14], for any given sequence of numbers bounded by 1, such as $\{A_k^*\}_{k\geq 0}$, $|A_k^*| \leq 1$, there exists r, $0 < r < 1$ such that, for all $k \geq 1$ there exists a number $\tilde{A}_k$, such that

$$\left| A_k^* - \tilde{A}_k \right| < \frac{r^{2k}}{1 - r^{2k}}, \tag{21}$$

and

$$\begin{aligned} \tilde{A}_k &= \frac{1}{2kr^k} \sum_{j=1}^{2k} (-1)^j \Re\left(G(re^{ij\pi/k}) \right) \\ &= \frac{1}{2kr^k} \left(G(r) + (-1)^k G(-r) + 2 \sum_{j=1}^{k-1} (-1)^j \Re\left(G(re^{ij\pi/k}) \right) \right), \end{aligned} \tag{22}$$

where $i = \sqrt{-1}$ and $\Re(x)$ is the real part of the complex number x.

By using (22), we have transformed the computation of $A(C)$ from a convolution on C, involving a combinatorial sum, to a linear sum in C. However, if C is expressed in a very small unit, the linear sum can be time consuming. To use efficiently approximation (22), it is necessary to find a unit by which the peak rates and the channel capacity can be normalized to reduce the CPU time requirement. For practical computation of $\tilde{A}_k$, we chose, for example, the smallest peak rate as the unit.

IV. Estimation of the Aggregate Effective Bandwidth

Usually, the effective bandwidth for heterogeneous traffic classes is determined by equating the CLR relation in (5) to an overall cell loss requirement commonly assumed to be the most stringent individual requirement, and then solving the equation for C. This approach can guarantee a conservative bound on the required bandwidth because approximation (5) overestimates the actual CLR by a significant amount. When a tight bound on the aggregate CLR such as (17) is used, this approach becomes invalid. That is, depending on their burstiness, the different traffic classes see different CLRs in the FIFO buffer, and thus the aggregate CLR approximation may underestimate the CLR of one or more classes, particularly when the buffer size is small and the CLR requirements are sufficiently close to each other. Many researchers argued on this issue in the past (e.g. [15] [3]) and the conclusion is that the individual CLRs in the FIFO buffer are generally in the same order of magnitude, so that the aggregate CLR can be considered. However, they generally assume the use of an approximation of the CLR as conservative as (5). Our approach to determine the effective bandwidth relies on the individual CLR relation in (18).

Let α_i, $i = 1, \cdots, L$, be the cell loss requirement for traffic class i. Based on the results in Section 3, we can estimate the aggregate effective bandwidth $\hat{C}$ as the minimum C needed to satisfy the CLR requirements of all traffic classes for a fixed buffer size K as follows:

$$\hat{C} = \min\{C : P^i_{loss}(C, K) \leq \alpha_i, i = 1, \cdots, L\}. \tag{23}$$

Since for all i, $P^i_{loss}(C, K)$ is a decreasing function of C, it is easy to see that

$$\hat{C} = \max\{C_i : P^i_{loss}(C_i, K) = \alpha_i, i = 1, \cdots, L\}. \tag{24}$$

Let C_i be defined as the aggregate required bandwidth to guarantee a CLR α_i to class i. Then we have the following proposition.

Proposition 1: Let $g(z) = \sum_{i=1}^{L} M_i g_i(z)$ as defined in (10)

(i) $C \geq C_i$ if and only if $C \geq g\left(\frac{\ln \alpha_i - \ln A_i(C)}{K}\right)$;

(ii) $C \leq C_i$ if and only if $C \leq g\left(\frac{\ln \alpha_i - \ln A_i(C)}{K}\right)$.

Proof Since (i) and (ii) are symmetric, it is sufficient to prove only (i). $P^i_{loss}(C, K)$ is a decreasing function of C. We thus have

$$C \geq C_i \iff A_i(C) e^{z_0(C)K} \leq \alpha_i,$$

$$\iff z_0(C) \leq \frac{\ln \alpha_i - \ln A_i(C)}{K}.$$

Since $g(z)$ is a monotonically decreasing function of z, we have

$$z_0(C) \leq \left(\frac{\ln \alpha_i - \ln A_i(C)}{K}\right) \iff g\left(z_0(C)\right) \geq g\left(\frac{\ln \alpha_i - \ln A_i(C)}{K}\right)$$

and with (10)

$$C \geq g\left(\frac{\ln \alpha_i - \ln A_i(C)}{K}\right),$$

which proves (i) $\square$

Let $\overline{\Delta} = \sum M_i \Delta_i \tau_i$ and $\widetilde{\Delta} = \sum M_i \Delta_i$ be the aggregate mean and aggregate peak input rates respectively. We have $\hat{C} \in \left[\overline{\Delta}, \widetilde{\Delta}\right]$. For a given relative error ϵ, $\hat{C}$ can be calculated by the simple bisection algorithm below based on Proposition 1.

$C_L = \overline{\Delta}$, $C_L = \Delta$ {Initialize the upper and lower bounds of the solution}
repeat
  $\hat{C} = \dfrac{C_L + C_U}{2}$;
  if $\hat{C} - \max\limits_{i=1,\cdots,L} g\left(\dfrac{\ln \alpha_i - \ln A_i(\hat{C})}{K}\right) \geq 0$ **then**
    $C_U = \hat{C}$
  else
    $C_L = \hat{C}$
  end if
until $\dfrac{C_U - C_L}{C_L} < \epsilon$

Algorithm 1: Simple bisection algorithm to calculate $\hat{C}$

This algorithm requires only $\log_2(\widetilde{\Delta} - \overline{\Delta}) - \log_2(\varepsilon)$ iterations. In each iteration, at most L values of the simple and explicit function $g(.)$ are calculated.

From the definition of $\hat{C}$, it is clear that it will guarantee the most stringent CLR requirement. In other words, the class corresponding to this stringent CLR will experience a CLR exactly equal to this requirement while the other less stringent classes will observe better performance than required. This obviously results in a waste of bandwidth. Besides, it is well known that FIFO discipline cannot guarantee different CLRs. As argued in [15] and [3] the individual CLRs experienced by the different classes in a FIFO buffer are roughly the same for all classes and are more or less within one order of magnitude around the aggregate CLR. Based on this argument, to reduce further the complexity of the algorithm above, we propose to take the most stringent CLR α, $\alpha = \min_i \alpha_i$ as the target requirement for all the sources and approximate $\hat{C}$ by $\tilde{C}$, the solution of the equation

$$\tilde{C} - g\left(\frac{\ln \alpha - \ln A(\tilde{C})}{K}\right) = 0. \tag{25}$$

With this approximation, the function $g(.)$ is evaluated only one time in each step of the algorithm instead of L times.

To reduce the waste of bandwidth, we suggest that the traffic classes are segregated into different FIFO buffers each of which guarantees a given QoS requirement. By guaranteeing the same CLR for all the traffic sources sharing each FIFO buffer, none of the traffic sources will experience better performance than required. Besides, since the requirements of the sources in each buffer are the same, the above algorithm can be used to determine $\tilde{C}$ to guarantee the aggregate CLR in each buffer. In the resulting multi-buffer architecture, the waste of statistical multiplexing gain due to the segregation of the different traffic streams can be accounted for by using a round-robin like scheduling algorithm (e.g. [16]).

Equation (25) can be solved numerically using a bisection algorithm similar to the one Algorithm 1 above. In addition to these approximations, we provide in the following a direct and closed form formula for an upper bound to $\tilde{C}$.

Let $\tilde{C}$ be the solution of (25), we thus have

$$A(\tilde{C})e^{z_0 K} = \alpha \quad \text{and} \quad g(z_0) = \tilde{C}.$$

Since $A(C)$ is a decreasing function of C and $C \in [\overline{\Delta}, \widetilde{\Delta}]$,

$$\begin{aligned} z_0 &= \frac{\ln(\alpha) - \ln(A(\tilde{C}))}{K} \\ &\geq \frac{\ln(\alpha) - \ln(A(\overline{\Delta}))}{K}. \end{aligned}$$

Since $g(.)$ is a decreasing function,

$$g(z_0) \leq g\left(\frac{\ln(\alpha) - \ln(A(\overline{\Delta}))}{K}\right),$$

and thus we have

$$\tilde{C} \leq g\left(\frac{\ln(\alpha) - \ln(A(\overline{\Delta}))}{K}\right) \triangleq \overline{C},$$

where by definition, $\overline{C}$ is an upper bound to $\tilde{C}$.

V. Numerical results

A. Accuracy of the bounds on P_{loss} and P^i_{loss}

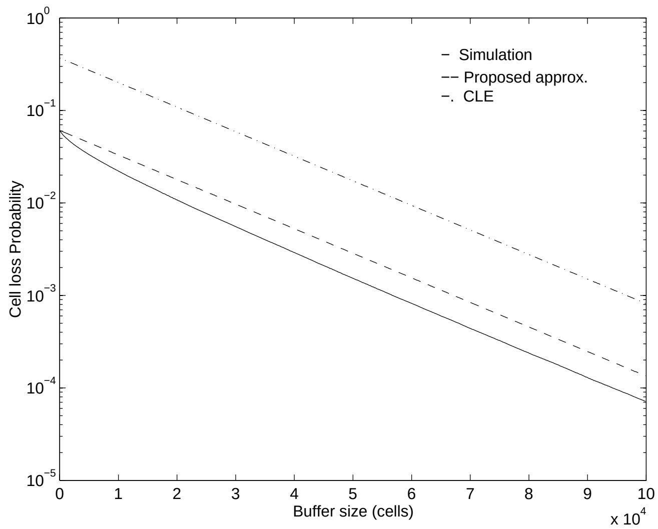

Fig. 1. Cell loss probability vs. buffer size: Homogeneous sources

In this subsection, we show the accuracy of the approximations proposed for the aggregate and individual CLRs. We present in the following a sample of figures representing the cell loss probability as a function of the buffer size. On these figures, our results are compared to those obtained from a simulation and to those obtained from approximation (6) proposed in [8].

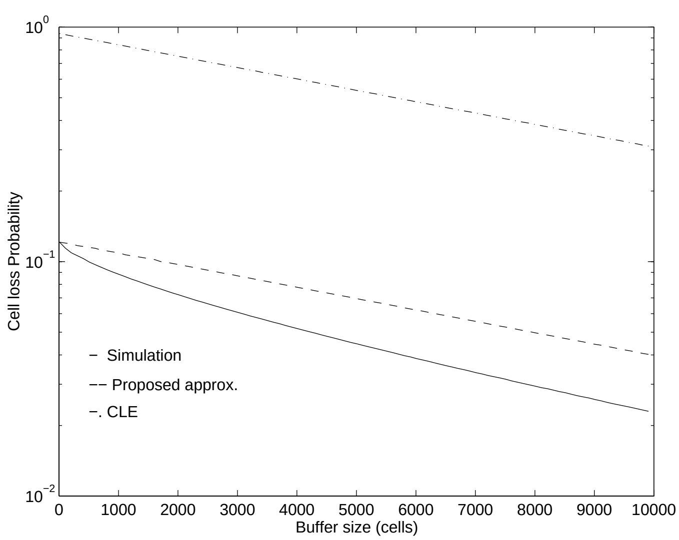

Fig. 2. Overall cell loss probability vs. buffer size: Heterogeneous sources

Figure 1 shows the cell loss probability against the buffer size for a superposition of 100 independent on/off sources with exponentially distributed on and off periods. The peak bitrate is 2 Mbits/s, the mean bitrate is 0.087 Mbits/s and the mean burst size is 5000 cells. The channel capacity is equal to 12 Mbits/s. In this figure we compare our approximation to the simulation results on one hand and to the "Chernoff's largest eigenvalue" approximation (CLE) (6) on the other hand. As expected, our approximation proves very tight and outperforms the CLE approximation. The CLE gives in fact an approximation of the overflow probability instead of the cell loss probability. The intercept of our approximation is almost exactly the same as that given by simulation.

To illustrate the accuracy of the proposed bounds and approximations for a superposition of heterogeneous traffic sources, we present in the following a sample of figures where we compare our results to those obtained from simulation and/or the other alternative methods. The traffic characteristics used in this case are depicted in Table I.

Class	Peak rate (Mb/s)	Mean rate (Mb/s)	Burst (Cells)
1	10	1	2000
2	2	0.087	400
3	0.064	0.021	53

TABLE I
TRAFFIC CHARACTERISTICS FOR THE HETEROGENEOUS CASE

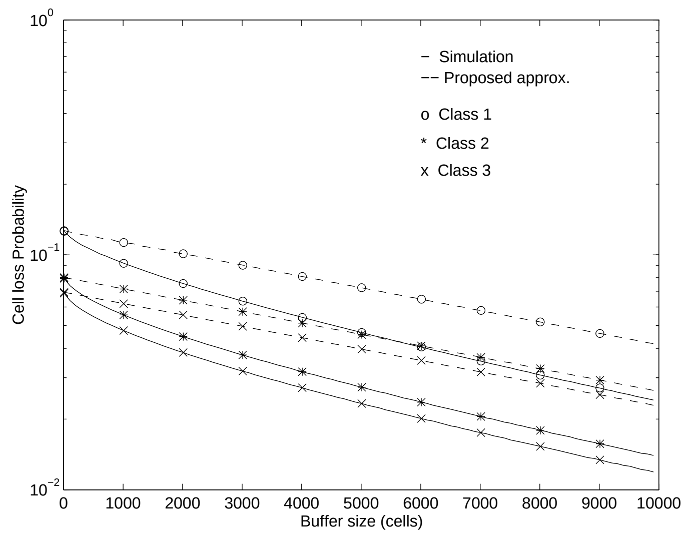

Fig. 3. Individual cell loss vs. buffer size: Heterogeneous sources

Figures 2 and 3 show the overall and the individual cell loss probabilities respectively, for a superposition of three different traffic streams, all with exponentially distributed and independent on and off periods. The traffic characteristics are shown in Table I and the channel capacity is 40 Mbits/s. Similar to the homogeneous case, Figure 2 shows that our approximation outperforms the CLE approximation for a superposition of heterogeneous classes of sources, by at least one order of magnitude.

B. *Accuracy of approximation* $\tilde{A}_{\lfloor C \rfloor}$

In this subsection we want to show the accuracy of approximation (22). The input traffic is a mix of 30 class-1, 30 class-2 and 40 class-3 sources with the traffic characteristics as given in Table I. As shown in Table II, the proposed approximation proves very accurate. Besides, the

Channel capacity C Mbits/s	Exact intercept $A(C)$	Approximation $\tilde{A}_{\lfloor C \rfloor}$
20	4.508e-01	4.516e-01
40	1.216e-01	1.216e-01
60	1.840e-02	1.847e-02
80	1.581e-03	1.581e-03
100	7.938e-05	7.983e-05
120	2.393e-06	2.392e-06
160*	4.953e-10	1.036e-09

TABLE II

COMPARISON OF THE EXACT $A(C)$ TO THE APPROXIMATION $\tilde{A}_{\lfloor C \rfloor}$

evaluation of $\tilde{A}_{\lfloor C \rfloor}$ requires only a linearly increasing computation time in C, while the calculation of the exact value of the intercept $A(C)$ requires an exponentially increasing computation time in C.

In Table II, we notice that the approximation $\tilde{A}_{\lfloor C \rfloor}$ begins to be inaccurate when C is large (entries marked with a $*$ symbol). This is mainly due to two reasons. First, the absolute error bound we choose 10^{-6} is larger than the exact value of $A(C)$; and second, the load of the system for the corresponding values of C is very low. For more practically interesting cases, when the system load is reasonable, the intercept should be large (e.g. $> 10^{-4}$), in which case our approximation is quite accurate.

C. Accuracy of the required bandwidth estimates

In the following, to illustrate the performance of the proposed upper bounds for the effective bandwidth, some numerical examples are given. Our results are compared to the alternative solutions based on approximations (5), (6) and to the Gaussian approximation of the stationary input rate tail proposed in [9].

Buffer size (cells)	$\hat{C}$	$\tilde{C}$	$\overline{C}$	(5)	(6)	Gaussian
2500	100.7	104.9	243.8	249.4	112.1	102.1
5000	93.3	96.9	194.1	202.4	103.9	102.1
10000	80.3	83.7	145.9	152.8	89.9	102.1

TABLE III

AGGREGATE REQUIRED BANDWIDTH WITH DIFFERENT ALGORITHMS (IN MBITS/S)

In Table III, we show the required bandwidth estimated by different methods, for different buffer sizes. The input traffic is a mix of 20 class-1 sources with a CLR requirement equal to 10^{-5}, 50 class-2 sources with a CLR requirement equal to 10^{-7} and 100 class-3 sources with a CLR requirement equal to 10^{-3}. The results show that our bounds $\hat{C}$ and $\tilde{C}$ outperform the alternative methods in terms of tightness.

As we argued in the previous section, we believe it is more valuable to segregate the traffic streams in different buffers based on their CLR requirements (more generally based on the CLR and the delay). In this case our bounds give a very accurate estimate of the required bandwidth. To show the accuracy of our approximation under this kind of situation, we compare our bounds to the exact result obtained from simulation, we chose different cases with different traffic mixes and buffer sizes. Besides, we set the QoS requirement for each class of traffic exactly equal to the individual cell loss probability observed in the simulation. The four different traffic mixes we used are depicted in Table IV.

Cases	Buffer (Cells)	Class	Number of Sources	Peak rate (Mb/s)	Mean rate (Mb/s)	Burst (Cells)	α_i
1	110						
		1	30	10	1	2000	0.118782
		2	30	2	0.087	400	0.073950
		3	40	0.064	0.021	52	0.063790
2	9910						
		1	30	10	1	2000	0.024103
		2	30	2	0.087	400	0.013984
		3	40	0.064	0.021	53	0.011924
3	9910						
		1	50	1	0.1	400	0.000739
		2	50	2	0.1	600	0.001040
4	4600						
		1	20	10	2	2000	0.000028
		2	40	2	0.1	600	0.000014
		3	40	0.064	0.021	60	0.000010

TABLE IV
Traffic characteristics for the different cases

Cases	Buffer size (cells)	Exact	$\hat{C}$	$\tilde{C}$	$\overline{C}$	(5)	(6)	Gaussian
1	110	40.0	40.7	47.3	332.1	349.5	62.7	65.2
2	9910	40.0	42.7	45.7	55.6	79.4	53.9	77.4
3	9910	12.0	12.3	12.4	13.0	14.6	13.5	23.3
4	4600	90.0	99.1	103.4	181.0	194.3	113.1	128.1

TABLE V
Aggregate required bandwidth with different algorithms (in Mbits/s)

As shown in Table V, the bound $\hat{C}$ obtained from the individual cell loss requirements proves very accurate and, as expected, performs better than all the other approximations. Extensive calculations have shown that the bound $\tilde{C}$ is always close to $\hat{C}$. This is expected since the overall cell loss probability is always close to the individual ones (see for example [15]). The approximation $\overline{C}$ overestimates the exact aggregate bandwidth. This is due to the fact that like approximation (5), $\overline{C}$ is obtained from the largest upper bound one can choose for the cell loss probability. Intuitively, we conjecture that the difference between these two approximations, namely, $\overline{C}$ and (5), is proportional to the difference between the exact aggregate cell loss probability and the overflow probability in the same system.

Table VI shows computation times required to obtain the values in Table V. The elapsed computational times in the simulation have been omitted in this table as these are obviously much larger (many orders of magnitude) than those required by the different approximations. The computation time for every algorithm is in general too small to be accurate enough, since most of the approximations have very low complexity. In order to compare the computation times required by the different methods, each algorithm had to executed a certain number of times and the execution time is averaged over the number of iterations. For instance, the value $34e-6$ means that the algorithm has been executed (in a *for* loop) 10^6 times in 34 seconds. From Tables VI and V, it can be seen that $\tilde{C}$ is probably the most attractive approximation among all, as it balances between computational complexity and accuracy of the approximation. In heavy traffic conditions, when the conditions for applying the central limit theorem are satisfied, it

Cases	C	C	C	(5)	(6)	Gaussian
1	182e-3	48e-3	3e-3	34e-6	44e-3	15e-6
2	212e-3	55e-3	3e-3	33e-6	43e-3	15e-6
3	16e-3	6e-3	2e-4	24e-6	31e-3	14e-6
4	268e-3	83e-3	7e-3	33e-6	45e-3	16e-6

TABLE VI

Required processing times (seconds) for different algorithms

is worth considering the Gaussian approximation. That is, under heavy traffic, the Gaussian approximation is quite accurate, while, as shown in Table VI, it requires a negligibly small computation time.

VI. Conclusion

In this paper we proposed tight and simple upper bounds to the aggregate and individual cell loss probabilities in an ATM multiplexer. To evaluate these bounds, a long CPU time is sometimes required. An approximate method for evaluating the bounds had been given. This approximation reduces the complexity from a combinatorial to a linear function of the channel capacity. Based on our proposed bounds, a fast and accurate algorithm to estimate the aggregate required bandwidth is provided. Numerical results show the better performance of our algorithm when compared to the alternative methods in terms of both accuracy and efficiency.

References

[1] D. Anick, D. Mitra, and M. Sondhi, "Stochastic theory of a data handling system with multiple sources," *Bell Systems Technical Journal*, vol. 61, no. 8, pp. 1871–1894, 1982.

[2] T. Yang and D. H. K. Tsang, "A novel approach to estimating the cell loss probability in an ATM multiplexer loaded with homogeneous sources," *IEEE Transactions on Communications*, vol. 43, pp. 117–126, January 1995.

[3] N. Baiocchi, N. Blefari-Melazzi, A. Roveri, and F. Salvatore, "Stochastic fluid analysis of an ATM multiplexer loaded with heterogeneaous on-off sources: an effective computational approach," in *IEEE INFOCOM'92*, pp. 405–414, 1992.

[4] A. I. Elwalid and D. Mitra, "Effective bandwidth of general markovian traffic sources and admission control in high speed networks," *IEEE/ACM Transactions on Networking*, vol. Vol. 3, pp. 329–343, 1993.

[5] D. Sykas, K. M. Vlakos, K. P. Tsoukatos, and E. N. Protonotarios, "Performance evaluation of analytical models for effective bandwidth allocation in atm networks," *European Transactions on Telecommunications*, vol. 5, pp. 391–396, May/June 1994.

[6] J. H. S. Chan and D. H. K. Tsang, "Bandwidth allocation of multiple QoS classes in ATM environment," in *IEEE Infocom'94*, pp. 360–367, 1994.

[7] B. Bensaou, J. Guibert, J. W. Roberts, and A. Simonian, "Performance of an ATM multiplexer queue in the fluid approximation using the Beneš approach," *Annals of Operations Research*, vol. 49, pp. 137–160, 1994.

[8] A. I. Elwalid and D. Mitra, "Analysis, approximations and admission control of multi-service multiplexing system with priorities," in *IEEE Infocom'95*, pp. 463–472, 1995.

[9] R. Guérin, H. Ahmadi, and M. Naghshineh, "Equivalent capacity and its application to bandwidth allocation in high-speed networks," *IEEE Journal on Selected Areas in Communications*, vol. 9, pp. 968–981, September 1991.

[10] J. W. Roberts, "Queueing models for variable bit rate traffic streams," *International Journal of Electronics and Communications*, vol. 47, no. 5/6, pp. 435–439, 1993.

[11] F. P. Kelly, "Effective bandwidths at multi-class queues," *Queueing Systems*, vol. 9, pp. 5–16, 1991.

[12] J. Y. Hui, "Ressource allocation for broadband networks," *IEEE Journal on Selected Areas in Communications*, vol. 6, pp. 1598–1608, December 1988.

[13] G. Kesidis, J. Warland, and C. S. Chang, "Effective bandwidths for multiclass Markov fluids and other ATM sources," *IEEE/ACM Transactions on Networking*, vol. 1, pp. 424–428, August 1993.

[14] J. Abate and W. Whitt, "Numerical inversion of probability generating functions," *Operations Research Letters*, vol. 12, pp. 245–251, October 1992.

[15] I. Norros and J. Virtamo, "Who loses cells in the case of burst scale congestion?," in *Teletraffic and Datatraffic in a Period of Change. Proceedings of the Thirteenth International Teletraffic Congress*, 1991.

[16] K. T. Chan, B. Bensaou, and D. H. K. Tsang, "Credit-based fair queueing for packet networks," *IEE Electronics Letters*, vol. 33, March 1997.