Estimating the Loss Probability in a Multiplexer Loaded with Multipriority MMPP Streams

Nelson L.S. Fonseca and John A. Silvester

CENG Technical Report 93-03

Department of Electrical Engineering - Systems
University of Southern California
Los Angeles, California 90089-2562
(213)740-4579

ESTIMATING THE LOSS PROBABILITY IN A MULTIPLEXER LOADED WITH MULTI-PRIORITY MMPP STREAMS

Nelson L.S. Fonseca and John A. Silvester

Department of Electrical Engineering - Systems

University of Southern California

Los Angeles, CA 90089-2562

ABSTRACT

Future Broadband Integrated Service Digital Networks (B-ISDN) will carry a wide spectrum of applications with different loss requirements. By adopting a multi-priority scheme at the cell level, we gain flexibility in coping with these diverse requirements in an efficient way. In this paper, we address the issue of furnishing a *multiclass selective discard mechanism*. We compute the loss probability per class where each class is modeled as a two-state Markov Modulated Poisson Process (a 2-MMPP$_1$...2-MMPP$_n$/D/1/K queue). We make use of a Multiple Class Buffer Priority algorithm, which has linear complexity as a function of the number of classes. We illustrate the advantages of a multi-priority scheme as well as the effect of the burstiness on the loss probability.

I) INTRODUCTION

Future Broadband Integrated Services Digital Networks (B-ISDN) will carry a wide spectrum of multimedia applications with different Quality of Service (QOS) requirements. Coping with diverse delay and loss requirements in a cost/effective way calls for the implementation of differential services at the cell level. An alternate approach is to provide a *unique* bearer service and to dimension the network resources according to the most demanding class. This solution either results in network underutilization or leads to unnecessary restriction of user's demands.

In a statistical multiplexer, whenever the incoming stream of cells exceeds the storage capacity, loss occurs. A ***multiclass selective discard mechanism*** discards cells in an overflow situation according to their priority level. Whenever a higher priority cell finds the buffer full and there is an enqueued lower priority cell, the lower priority one is discarded in order to release a buffer (at the tail of the queue) for the higher priority cell [1] [2]. By providing different service levels to different streams, the network can support several different loss requirements that would be impossible otherwise.

Multimedia traffic modelling has gained a lot of attention in the network field starting with the advent of Integrated Networks, for example [3]. Two-state *Markov Modulated Poisson Arrival* (MMPP) models have been successfully used for modelling the superposition of *on-off* sources [4]. Several procedures have been developed for matching the statistics of the aggregate stream with the parameters of a two state MMPP [5]-[8]. In this paper, we consider a finite capacity multiplexer loaded with *n* two state MMPP inputs where each input has different buffer priority ($2\text{-MMPP}_1 \ldots 2\text{-MMPP}_n/D/1/K$). We make use of the *Multiple Class Buffer Priority* (MCBP) algorithm, first presented in [9], to compute the loss probability per class in a multiple class selective discard system. The MCBP algorithm allows the analysis of a multi-priority queue in O (*n*) steps where *n* is the number of classes. We, finally, emphasize the advantages of a multi-priority system and the impact of the arrival stream's burstiness on system performance.

II) THE SUPERPOSITION OF ON-OFF SOURCES

The integration of video, voice and data has led to many interesting traffic modelling problems. The *on-off* model [4] accurately captures the behavior of sources such as: voice, still image and interactive data. However, the modelling of video sources with scene changes is still an active research area.

The major challenge in modelling the superposition of several *on-off* sources is that the aggregate arrival process is not renewal [10]. The non-renewal nature comes from the variability of the instantaneous arrival rate due to the fluctuation of the number of sources in the *on* state. The aggregate process differs from Poisson in the correlation between successive interarrival times. Although this is small, at high traffic intensity, the long-term accumulation of these small correlations has a major impact on the system performance

In a Markov Modulated Poisson Process, the Markovian arrival rates are determined by the state of a continuous time Markov chain [11]. The two state MMPP is completely defined by the arrival rates and the sojourn time of each state. The two-state MMPP has received a lot of attention due to the accuracy and tractability of mapping its parameters to the statistics of a process resulting from the superposition of homogeneous *on-off* sources [5]-[8]. The original work by Heffes and Lucantoni [5] considered the mean, the variance and the third moment of the aggregate process. Biocchi et al [6] divide the superposition process into overload and underload periods. An overload period occurs whenever the number of active sources exceeds the system capacity. They match the mean arrival rate and the sojourn time of the underload/overload states with the states of the MMPP. Wang and Silvester [7] use a similar approach where they consider the upper bound of the residence time in the overload state instead of an asymptotic approximation [6]. Narajan et al. [8] also used the underload/overload approach. They match the variance of the number of arrivals in the overload state. Regarding the superposition of heterogeneous on/off sources, Li [12] developed a procedure based on a generating function approach which takes into consideration the characteristics of each individual source. Although very useful, in general the computational complexity is high.

In order to analyse the 2-MMPP..2-MMPP$_n$/D/1/K queue, we will use the definition of the superposition of n two-state MMPP's, as in [13] [14]. This aggregate process is still MMPP with 2^n states. Given that the j^{th} 2-state MMPP is completely defined by the transition rate matrix Q_j and the matrix $\Lambda_j = diag(\lambda_1, \lambda_1)$; the superposition of the n 2-state MMPP's is defined by the 2^n x 2^n matrices:

$$Q = Q_1 \oplus Q_2 \oplus \ldots \oplus Q_n$$

$$\Lambda = \Lambda_1 \oplus \Lambda_2 \oplus \ldots \oplus \Lambda_n$$

where $Z \oplus Z$ represents the Kronecker matrix sum. The Kronecker sum of n matrices is given by:

$$M_1 \oplus M_2 \oplus \ldots \oplus M_n = M_1 \otimes I \otimes \ldots \otimes I + I \otimes M_2 \otimes \ldots \otimes I + \ldots + I \oplus I \oplus \ldots \otimes M_n$$

where $A \otimes B$ is the Kronecker product of two matrices. This product is defined as the matrix with block elements equal to $[a_{ij}B]$ In the case where A and B are 2x2 matrices, we have:

$$A \oplus B = A \otimes I + I \otimes B =$$

$$\begin{bmatrix} a_{11} + b_{11} & b_{12} & a_{12} & 0 \\ b_{21} & a_{11} + b_{22} & 0 & a_{12} \\ a_{12} & 0 & a_{22} + b_{11} & b_{12} \\ 0 & a_{21} & b_{21} & a_{22} + b_{22} \end{bmatrix}$$

Note that the states of the aggregated process defined by the Kronecker sum of the n 2-state MMPP's correspond to the lexicographic ordering of the MMPP's that are in the first state.

III) THE MCBP ALGORITHM

Selective discard has been extensively studied in the literature. Several results are available for two class systems [16]-[24]. Studies differ in: i) the arrival assumptions (Poisson [18], Geometric [19] and batch arrival [20]), ii) different service time (exponential [18], deterministic and general [21]), and iii) the buffer organization (complete sharing [21] [22] and partial sharing [23] [24]). Recently, Elwalids and Mitra [25] have derived a model for multiple class systems with partial buffer sharing organization [25]. The Multiple Class Buffer Priority (MCBP) algorithm computes the loss probability per class in a multi-priority queue with complete buffer sharing organization [9].

In addition to the selective discard mechanism that discards cells in an overflow situation according to their priority level, we define how this principle interacts with the buffer organization. Buffer space can be shared in two different ways: partial sharing and complete sharing. In partial buffer sharing a threshold is set for the number of buffers shared among all classes. If the number of cells exceeds this threshold, only cells with higher priority are admitted to the system. In the complete buffer sharing scheme there is no threshold and a cell is always granted the right to wait as long as there is an available buffer. A queueing scheme is considered work-conservative if no customer is refused the right of using a queueing resource whenever it is available. If: i) no server is left idle when the buffer is not empty, ii) customers are lost only when the buffer is full, iii) all classes of customers have the same service requirement, iv) served customers are immediately removed from the system and v) no work (i.e. service requirement) is created or destroyed by any particular employed rule, then the queue is work-conservative [1,2]. According to this definition, complete buffer sharing with push out and fixed size cells is work-conservative whereas partial buffer sharing is not, since lower priority cells are lost whenever the threshold is reached. The MCBP algorithm exploits an important property of work-conservative queues [18] [20] [21], that:

$$\sum_{i=1}^{n} \gamma_i \Phi_i = \gamma \Phi$$

where Φ and Φ_i are the aggregate loss probability and the loss probability per class respectively. γ is the total arrival rate and γ_i is the per class arrival rate.

The main problem in solving the *n*-class case is the complexity growth as a function of the number of classes. For example, a straightforward approach to the exact solution for a two class system has O (K^2) complexity whereas for a *n* class one it is O (K^n) where *K* is the number of buffers and *n* the number of classes. The MCBP algorithm avoids this complexity by computing the solution of a *n* priority class system by solving *n-1* two priority class systems [9]. Each of these two priority class systems corresponds to a different grouping of the original *n* classes into two classes. The algorithm starts with the mapping of the highest priority class of the *n* class system into the high priority class of the two class system and the other *n-1* classes are mapped into the low priority class. At step *i*, the highest *i* priority classes are mapped into the high priority class of the two class system and the remaining *n-i* are mapped into the low priority class. The loss probabilities of the original *n* class system are computed based on the loss probabilities of the *n-1* two class systems.

Before showing the MCBP algorithm in more detail, we introduce some notation. Let Θ_n denote a *n* class model to be solved and Ψ_g a corresponding two class model. Let the *g* highest classes of Θ_n be mapped into the high class of Ψ_g and let the lowest *n* - *g* classes of Θ_n be mapped into the low class of Ψ_g.

Let γ_{1g} (γ_{2g}) be the high (low) class arrival rate and $\Phi_{1,g}$ ($\Phi_{2,g}$) loss probabilities of the high (low) class. Let assume that the classes are ordered in decreasing order of priority, that is the highest priority class is the first class.

The Multiple Class Buffer Priority algorithm (solution for Θ_n) can be stated as follows:

STEP 1 - For $g = 1$ to $n - 1$

Solve Ψ_g

STEP 2- $\Phi_n = \Phi_{2,n-1}$ and $\Phi_1 = \Phi_{1,1}$

STEP 3 - For $j = n - 1$ to 2 compute

$$\Phi_j = \frac{1}{\gamma_j}\left[\gamma_{2,j-1}\Phi_{2,j-1} - \sum_{z=j+1}^{n}\gamma_z\Phi_z\right]$$

Note that step 3 could be computed either top down or bottom up. Actually, Φ_j could be computed after the j^{th} step.

Step 1 contains *n-1* iterations and each iteration is essentially the solution of two class system, Ψ_g. Step 3 computes the *n-2* remaining loss probabilities by substitution of known values. Thus, the MCBP algorithm runs in O (*n*) steps, and its complexity depends on the complexity of the elementary step Ψ_g. The MCBP algorithm does not introduce any approximation and, again, its accuracy depends only on the accuracy of the solution of the two class model used.

We illustrate, below, the MCBP algorithm for a 5 class system:

STEP 1

	HIGH	*LOW*
Ψ_1	1	2,3,4,5
Ψ_2	1,2	3,4,5
Ψ_3	1,2,3	4,5
Ψ_4	1,2,3,4	5

STEP 2

$\Phi_5 = \Phi_{2,4}$

$\Phi_1 = \Phi_{1,1}$

STEP 3

$$\Phi_4 = \frac{1}{\gamma_4}[\gamma_{2,3}\Phi_{2,3} - \gamma_5\Phi_5]$$

$$\Phi_3 = \frac{1}{\gamma_3}[\gamma_{2,2}\Phi_{2,2} - \gamma_5\Phi_5 - \gamma_4\Phi_4]$$

$$\Phi_2 = \frac{1}{\gamma_2}[\gamma_{2,1}\Phi_{2,1} - \gamma_5\Phi_5 - \gamma_4\Phi_4 - \gamma_3\Phi_3]$$

IV) THE $MMPP_1$,$MMPP_2$/D/1/K

In this section, we derive the solution of a two priority class system that can be used as the elementary step Ψ_g of the MCBP algorithm. The inputs to each class are Markov Modulated Poisson Processes and each contains 2^{m_i} states where m_i is the number of superimposed MMPP sources that each input represents.

To compute the loss probability per stream, we follow the behavior of a general tagged low priority cell. Without loss of generality we compute the loss probability of the low priority class and then by use of the conservation law, we derive the loss probability of the high priority class. We proceed by showing a solution for the aggregate MMPP/1/D/1/K queue that follows the notation defined in Lucantoni [27]. The MMPP/1/D/1/K was originally studied by Blondia [26] and by Nagarajan et al. [8].

THE MMPP/D/1/K QUEUE

If we observe the queue at departure instants $(\tau_i, i \geq 0)$, the sequence $(n_i, m_i, \tau_{i+1} - \tau_i)$ forms a Markov renewal sequence where n_i and m_i are the number of cells in the queue and the state of the MMPP immediately after τ_i respectively. We can then define the embedded Markov chain with state (N_i, M_i) where N_i is the number of cells left behind by a departing cell and M_i is the state of the MMPP process. The transition matrix T of the embedded Markov chain is expressed by:

$$T = \begin{bmatrix} B_0 \; B_1 \; B_2 \cdots & \sum_{n=K}^{\infty} B_n \\ A_0 \; A_1 \; A_2 \cdots & \sum_{n=K}^{\infty} A_n \\ 0 \;\; A_0 \; A_1 \cdots & \sum_{n=K-1}^{\infty} A_n \\ 0 \;\; 0 \;\; A_0 \cdots & \sum_{n=K-2}^{\infty} A_n \\ \vdots \;\; \vdots \;\; \vdots \;\; \vdots & \vdots \\ 0 \;\; 0 \;\; 0 \;\; \cdots & \sum_{n=1}^{\infty} A_n \end{bmatrix}$$

where the $(i, j)^{th}$ element of the submatrix A_n, denoted by $a_{n,ij}$, is the probability that the MMPP makes the transition from i to j and n arrivals occur. Similarly, the $(i, j)^{th}$ element of B_n is the probability that n cells arrive and that the state of the MMPP is j at the end of a service time given that an idle period followed the last departure and the state of the MMPP was i at that departure.

The steady state probability $\pi_{n,m} = P\,[N = n, M = m]$ that a departing cell leaves n cells behind and the MMPP process is in state m can be computed by solving:

$$\Pi T = \Pi \qquad \Pi e = 1$$

where e is the unit columm vector.

In order to compute the A_n matrices, we consider an uniformized version of the MMPP process with uniformizing parameter $\theta = max(Q - \Lambda)_{ii}$ [28]. A_n is translated into:

$$A_n = \sum_{j=0}^{\infty} e^{-\theta\tau} \frac{(\theta\tau)^j}{j!} R_n^{(j)}$$

where $R_n^{(j)}$ is recursively defined by:

$$R_n^{(j)} = \theta^{-1} R_{n-1}^{(j-1)} \Lambda + R_n^{(j-1)} (I + \theta^{-1} (Q - \Lambda))$$

$$R_0^{(0)} = I$$

$$R_n^{(0)} = 0 \quad for \quad (n \geq 1)$$

The B_n matrices are given by:

$$B_n = (\Lambda - Q)^{-1} \Lambda A_n$$

The loss probability can be calculated as the ratio between the average number of lost cells and the average number of arrivals:

$$\Phi = \frac{\phi_1}{\phi_2}$$

$$\phi_1 = \sum_{k=1}^{K} \sum_{i=1}^{\overline{M}} \sum_{y=K-k}^{\infty} \sum_{j=1}^{\overline{M}} (y - K + k) \times a_{y,\,ij} \times a_{ij} \times \pi(k, i)$$

$$+ \sum_{y=0}^{\infty} \sum_{i=1}^{\overline{M}} \sum_{j=1}^{\overline{M}} (y - K - 1) \times b_{y,\,ij} \times a_{ij} \times \pi(0, i)$$

$$\text{and } \phi_2 = \sum_{k=1}^{K} \sum_{i=1}^{\overline{M}} \sum_{y=0}^{\infty} \sum_{j=1}^{\overline{M}} y \times a_{y,ij} \times a_{ij} \times \pi(k,i)$$

$$+ \sum_{y=0}^{\infty} \sum_{i=1}^{\overline{M}} \sum_{j=1}^{\overline{M}} y \times b_{y,ij} \times a_{ij} \times \pi(0,i)$$

where $\overline{M}$ is the number of MMPP states and $A = [a_{ij}] = \sum_{n=0}^{\infty} A_n$.

Note that A satisfies $XA = X$ and $Xe = 1$ where X is the MMPP steady state probability.

To derive the low priority loss probability, Φ_2, we focus on a tagged cell and compute its survival probability, Φ_s; that is the probability that during its waiting time, the tagged cell is not pushed out by a higher priority cell. We assume that the tagged cell arrives immediately after the beginning of a service period and that the dropping policy is Last in First Drop. To compute the survival probability, we first condition on the position that the tagged cell joins the queue (ie, the number of cells left by a departuring cell):

$$\Phi_s = \sum_{k=0}^{K} \sum_{i=1}^{M} S(k,i) \times \pi(k,i)$$

S (k,i) = P (*survive / joined the queue at position k and MMPP was in state i*)

The tagged cell survives the first service period if the maximum number of high priority arrivals is equal to $K - k$. It survives up to the second service period if and only if the maximum number of high priority arrivals is equal to $K - k + 1$. In general, it survives up to the w^{th} service period if and only if the cumulative number of high priority arrivals is at most equal to $K - k - (w-1)$ [18] [20]. This dynamics is translated into the following recursive discrete convolution algorithm:

$$\beta_1(z,i) = \begin{cases} \alpha_1(z,i) & 0 \le z \le K-k \\ 0 & otherwise \end{cases}$$

$$\beta_w(z,i) = \begin{cases} \alpha_w(z,i) \otimes \beta_{w-1}(z,i) & 0 \le z \le K-k+(w-1) \\ 0 & otherwise \end{cases}$$

where $\alpha_1(z,i) = \sum_{j=1}^{M} a_{z,ij} \times a_{ij}$ and $\alpha_w(z,i) = \sum_{i=1}^{M} \sum_{j=1}^{M} a_{z,ij} \times a_{ij} \times x_i$

The conditional probability *S (k,i)* can be derived as:

$$S(k,i) = \begin{cases} 1 & k = 0 \\ \sum_{z=0}^{K} \beta_w(z,i) & 1 \le k \le K \end{cases}$$

In turn, the low and high priority loss probabilities are given by:

$$\Phi_2 = 1 - \Phi_s$$

$$\Phi_1 = \frac{1}{\gamma_1}[\gamma\Phi - \gamma_2\Phi_2]$$

V) NUMERICAL EXAMPLES

In this section, we make use of our multiple class model to investigate the advantages of multiple priority classes and also to look at the effect of burstiness on network performance.

We have claimed that a multi-priority scheme enhances network flexibilty in coping with traffic scenarios that have diverse loss requirements. The first two examples illustrate this claim. In the first example, we consider four traffic classes with the parameters defined in table 1 (the transition rates are the same for all MMPP in this section). Class A is the highest priority and class D is the lowest . Figure 1 shows how these four classes are multiplexed when there are 1, 2, 3 or 4 priority levels and a buffer size equal to 10. Initially there is no priority, i.e. all classes receive the same performance. When we move to two prioritiey classes, we split off the most demanding traffic class (A). If three priority levels are available we use (A), (B) and (C, D). Finally for four priority levels each traffic class can be assigned to a different level. We can, thus, verify the advantage of an additional priority level by the difference in the loss probability in the corresponding traffic class. The flexibility in accomodating different loss requirements may reduce buffer requirements. Figure 2 shows the minimum buffer size required to transport the traffic scenario of table 2 as a function of the number of priority levels provided by the network.

The burstiness of a stream is a critical parameter that influences the loss probability. We use the ratio between the peak and average arrival rate as the burstiness descriptor. In order to vary the burstiness, we keep the mean arrival rate and the sojourn times constant and vary the difference between the arrival rate in each state [29]. It was noted in [29] that the burstiness of the low priority stream has no significant impact on the loss probability of the high priority class. The next three examples consider figure 3 traffic scenario. In the third example, figure 3, we have a four priority-level multiplexer loaded with four traffic classes. The average arrival rate is 0.125 and the burstiness is equal to1.5. In figure 3, we see the impact of the burstiness of the highest priority class on the loss probability of each class. In figure 4, we investigate the buffer requirements needed to mantain a given loss (or better) probability for all priority classes as a function of the burstiness of the highest priority class. In figure 5, we show for distinct burstiness values the average arrival rate of the highest priority class that causes two different total loss probability.

VI) CONCLUSIONS

In this paper, we determine the loss probabilities for each class for a multi priority class queueing system where the arrival process for each class is Markov Modulated Poisson Process. We make use of the Multiple Class Buffer Priority algorithm. The MCBP algorithm exploits the conservation property of the complete buffer sharing organization. It computes the exact solution in O (n) steps where n is the number of classes.

We show the advantage of having a multiple class selective discard mechanism to provide diverse QOS requirements. We also investigate the effect of burstiness on the loss probability. Estimating network performance for different traffic scenarios is an ongoing task that requires a fast accurate tool such as the one presented in this paper.

VII) REFERENCES

[1] L. P. Clare and I Rubin, "On the design of priotized multiplexing systems", in Proc. IEEE ICC'83, pp E5.3.1-E5.3.5, June 1983.

[2] L.P. Clare and I. Rubin. "Preemptive buffering disciplines for time-critical sensor communications", in Proc. IEEE ICC'86, pp 904-909, 1986.

[3] C Yuan and J. A. Silvester, "Queueing analysis of delay coinstrained voice traffic in a packect switching system", IEEE J. Select. Areas Commun., vol. 7, pp. 729-738, Jun1989.

[4] P.T. Brady, "A statistical analysis of on-off pattern in 16 conversations", Bell Systems Technical Journal, pp 73-91, Jan. 1968.

[5] H. Heffes and D. Lucantoni, "A markov modulated characterization of packetized voice and data traffic and related statistical multiplexer performance", IEEE J. Select. Areas Commun., vol. 4, pp. 856-868, Sep. 1986.

[6] A. Baiocchi, N.B. Melazzi, M. Listani, A. Roveri and R. Winkler, "Loss performance analysis of an ATM multiplexer loaded with high speed on-off processes", IEEE J. Select.

Areas Commun., vol. 9, pp. 388-393, Apr. 1991.

[7] S.S. Wang and J.A. Silvester,"A model for performance analysis of voice/data ATM multiplexer", University of Southern California, Tech Report CENG 92-7, 1992.

[8] R. Narajan, J.F. Kurose and D. Towsley,"Approximation techniques for computing packet loss in finite-buffered voice multiplexer", IEEE J. Select. Areas Commun. , vol. 9, pp. 368-377, Apr 1991; and University of Massachusetts at Amherst Tech Report n 90-12,1990.

[9] N.L.S. Fonseca and J. A. Silvester,"A multiple class buffer priority algorithm for the design of B-ISDN networks", Proc of the First Conference on Computer Communications and Networks, pp. 38-42 , San Diego, June 1992; and University of Southern California Tech Report CENG 92-8, 1992.

[10] K. Sriram and W. Whitt, "Characterizing superposition arrival processes in packet multiplexer for voice and data", IEEE J. Select. Areas Commun., vol 4, pp 833-846, Sep 1986.

[11] M.F. Neuts, Matrix-Geometric Solutions in Stochastic Models: An Algorithmic Approach, John hopkins University Press, Baltimore, MD, 1981.

[12] S.Q. Li, "A general technique for discrete queueing analysis of multimedia traffic on ATM", IEEE Trans. Commun., vol. 39, pp. 1115-1132, Jul.1991.

[13]M.F. Neuts and G. Latouche,"The superposition of two PH-renewal processes. In "Semi-Markov Models: Theory and Applications", J. Janssen, ed., London:Plenum Publishers, 131-177, 1986.

[14] K.S. Meiere-Hellstern,"The analysis of a queue arising in overflow models', IEEE Trans. Commun., vol 37, April 1989.

[15] R Bellman, Introduction to matrix analysis, McGraw Hill, New York, 1960.

[16] I. Rubin and M. Quaily, "Performance of finite capacity communication and queueing systems under various service and buffer preemptive policies", in Proc INFOCOM'88, pp 505-514, 1988.

[17] G. Hebuterne and A Gravey., "A space priority queueing mechanism for multiplexing ATM channels". InProc ITC Specialist Seminar'89, Adelaide, pp. n0 7.4, Sept 1989.

[18] H. Kroner, "Comparative performance study of space priority mechanisms for ATM networks", In Proc. IEEE INFOCOM'90, pp1136-1143, June 1990.

[19] J. Y. Le Boudec, "An efficient solution method for markov models of ATM links with loss priorities", IEEE IEEE J. Select. Areas Commun, vol. 9, pp. 408-417, Apr. 1991.

[20] A. Y-M Lin and J. A. Silvester, "Priority queueing strategies and buffer allocation protocols for traffic control at an ATM integarted broadband switching system, IEEE J. Select. Areas Commun., vol. 9 , pp. 1524-1536 , Dec. 1991.

[21] S. Sumita and T. Ozawa: Achievability of performance objectives in ATM switching nodes, Proc of the International Seminar on Performance of Distributed and Parallel Systems, Kyoto, pp. 45-56, Dec. 1988.

[22] P Landsberg and C Zukowski, "A novel buffer sharing method: complete sharing subject to guaranteed queue minimum", Proc of the First Conference on Computer Communications and Networks, pp 43-48, San Diego, June 1992.

[23] S.Q.Li, "Overload control in a finite message storage buffer", IEEE Trans. Commun, vol. COM-37, pp. 1330-1338, Dec. 1988.

[24] N. Yin, S Q. Li and T. E. Stern, "Congestion control for packet voice by selective packet discarding", IEEE Trans Commun, vol COM-38, no 5, pp 674-683, May 1990

[25] A.I. Elwaid and D. Mitra, "Fluid models for the analysis and design of statistical multiplexing with loss priorities on multiple classes of bursty traffic", Proc of INFOCOM'92, pp 0415-0425, Florence, 1992

[26] C. Blondia,"The N/G/1 finite capacity queue", Communication on Statistics - Stichastics Models, vol. 5, n0 2, pp. 273-294, 1989.

[27] D. M. Lucantoni, "New results on the single server queue with a batch markovian arrival process", Stochastic Models, vol.7, no 1, pp. 1-46, 1991.

[28] S.M. Ross, Introduction to probability models, Academic Press Inc, San Diego, 1989.

[29] J.J. Bae, T. Suda and R. Simha, "Analysis of individual packet loss in a finite buffer queue with heterogeneous markov modulated arrival processes: a study of traffic burstiness and priority packet discarding", in Proc INFOCOM'92, pp. 0219-0230 , Florence, 1992

Table 1: MMPP parameters for example 1

	transition rate	arrival rate
state 1	10^{-4}	0.175
state 2	10^{-2}	0.2625

Table 2: Traffic scenario for example 2

	A	B	C	D
load	0.05	0.35	0.2	0.3
loss	10^{-11}	10^{-9}	10^{-7}	10^{-5}

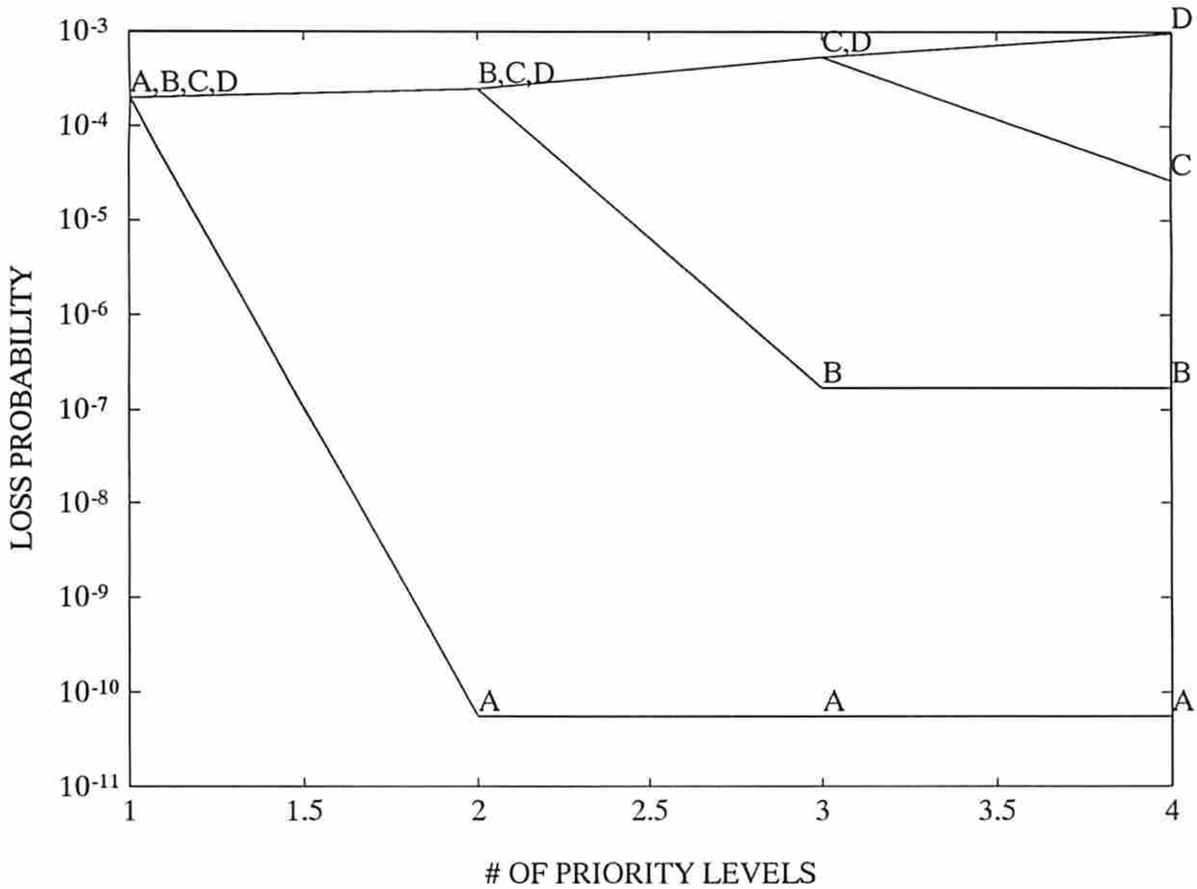

Figure 1: Loss probability as a function of the number of priority levels

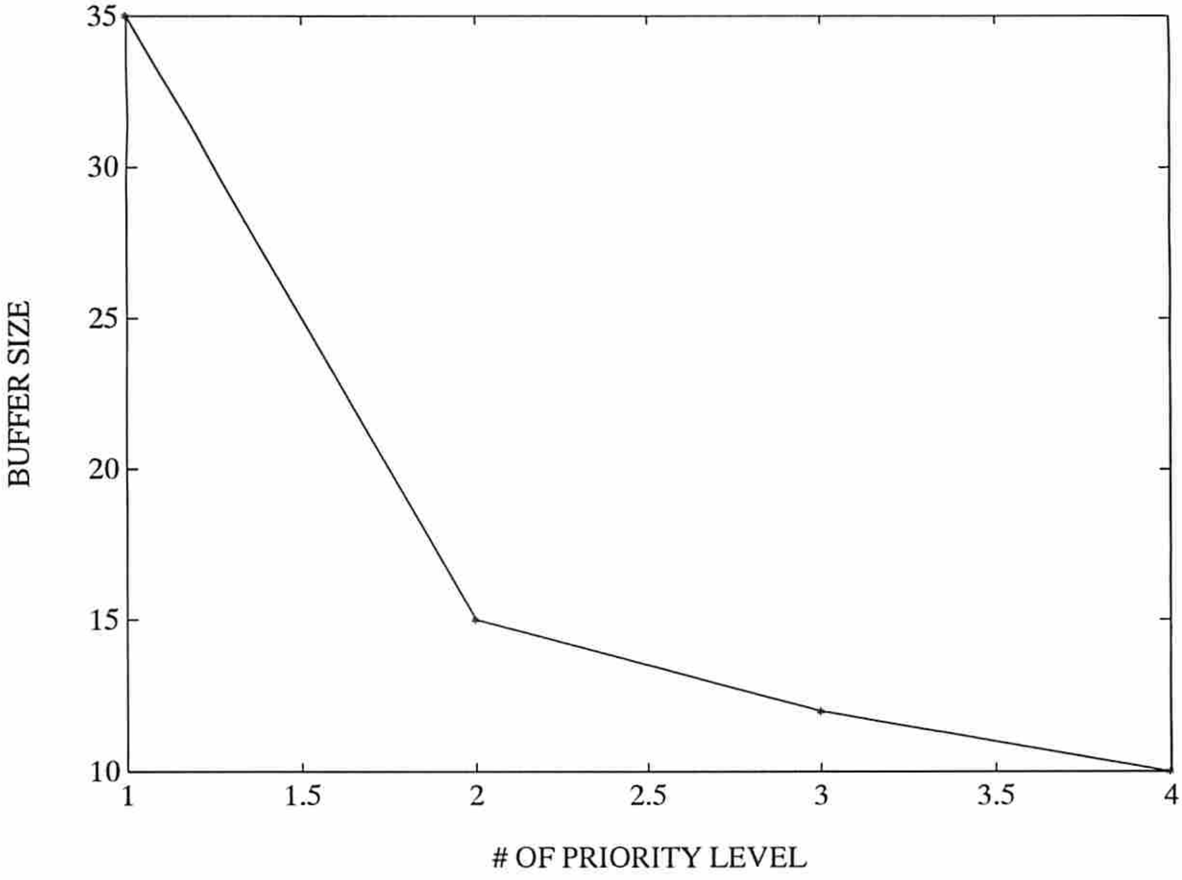

Figure 2 : Buffer requirements as a function of the number of priority levels

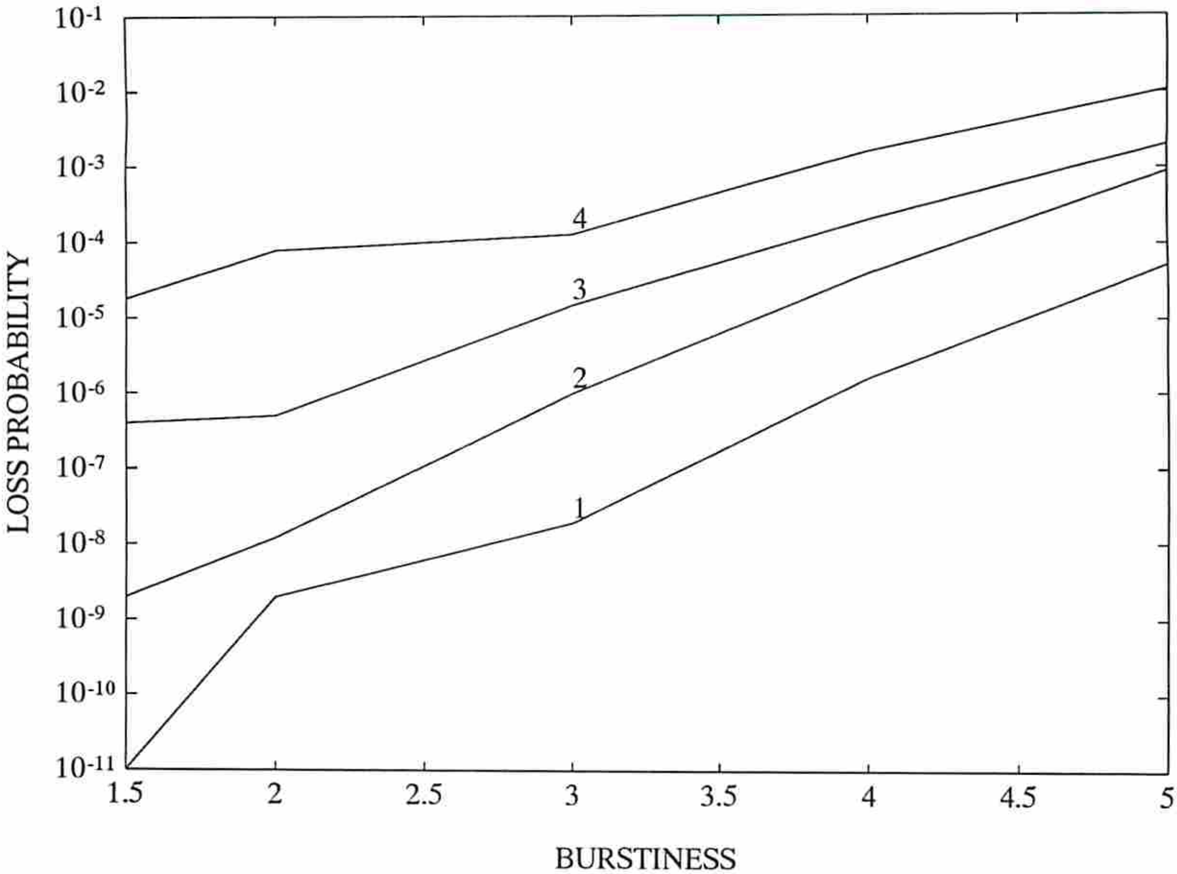

Figure 3: Loss probability per class as a function of the highest priority class burstiness

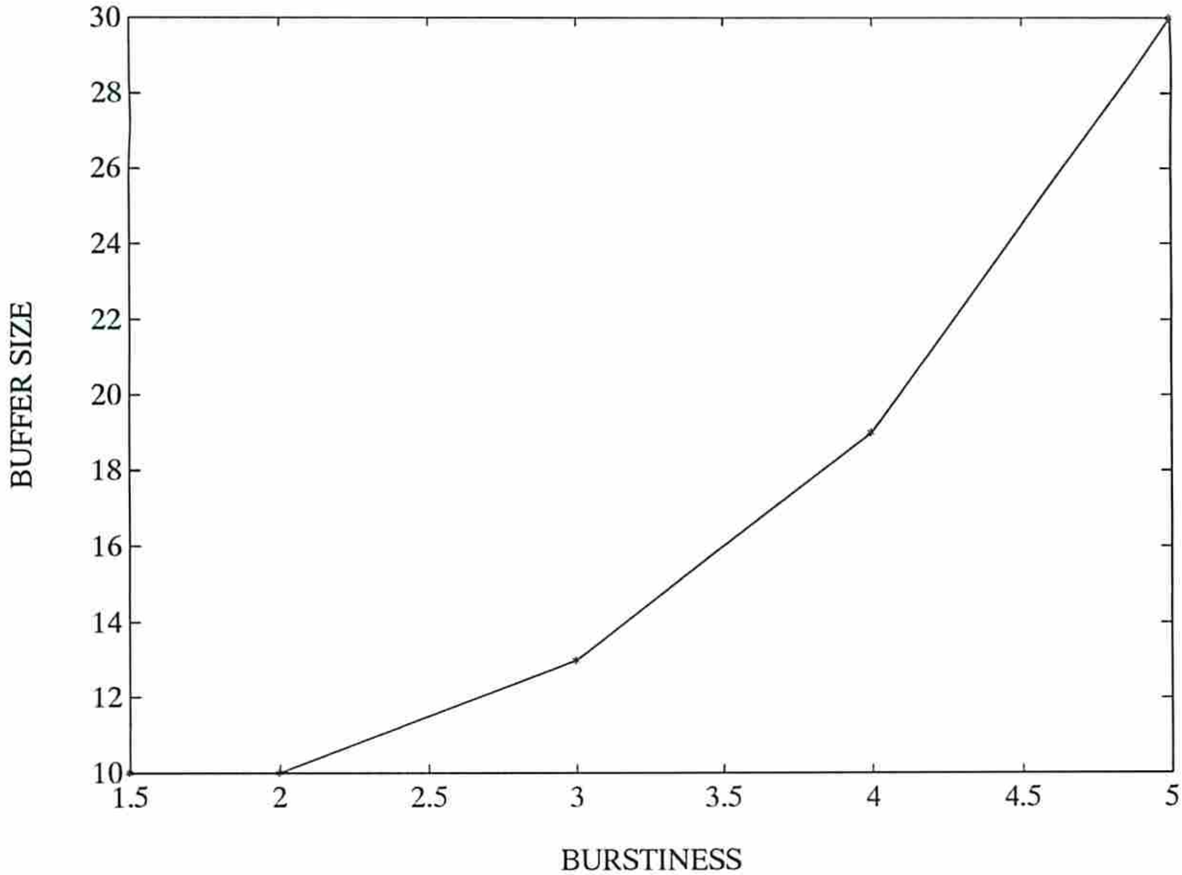

Figure 4: Buffer requirements as a function of the highest priority class burstiness

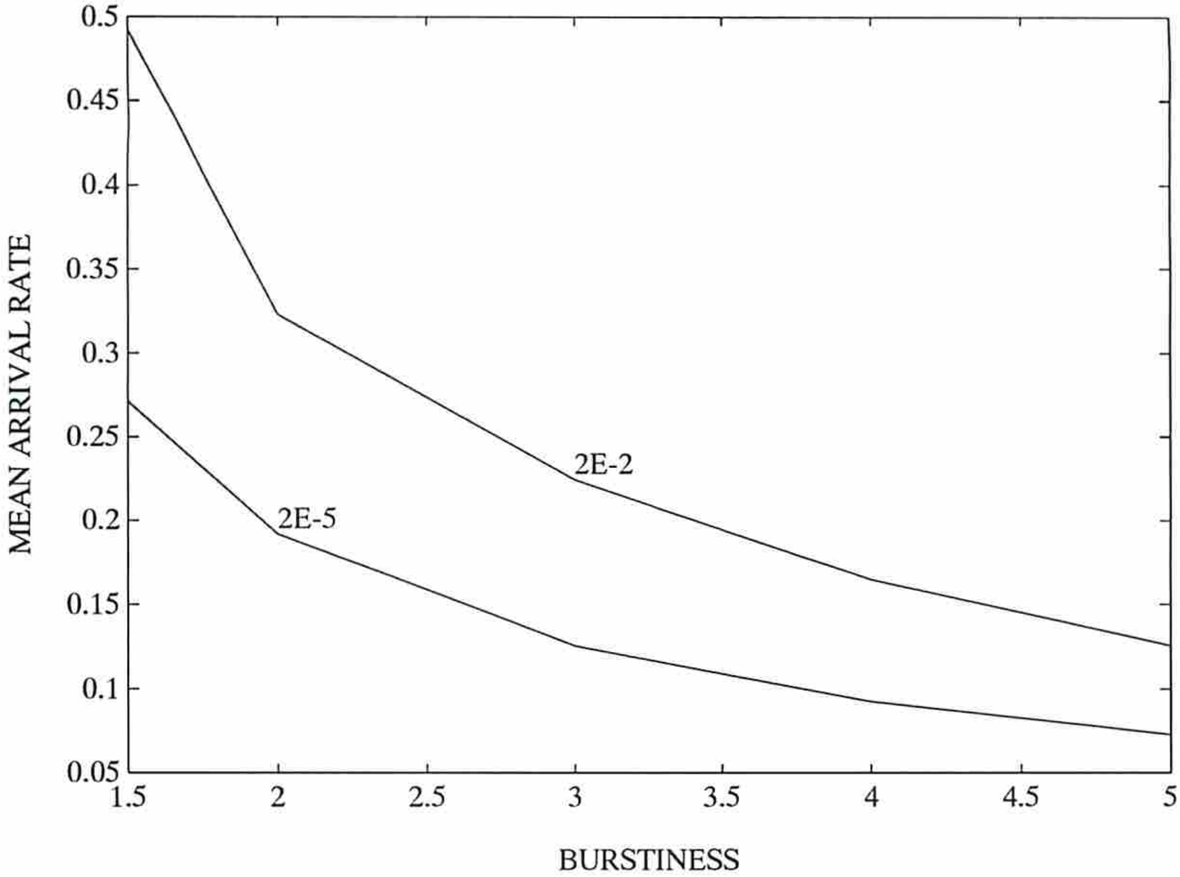

Figure 5: Equivalent mean arrival rate as a function of the highest priority class burstiness