PAPER

Efficient Cell-Loss Ratio Estimation for Real-Time CAC Decisions

Masaki AIDA† and Teruyuki KUBO††, Members

SUMMARY In ATM networks, Connection Admission Control (CAC) is a key part of traffic control but several challenging problems still remain. One is how to assign sufficient bandwidth fast enough to achieve real-time CAC. Although solutions to the bandwidth assignment problem have been proposed, they require a lot of calculations depending on the number of VCs and on the number of different VC types. Therefore, it is difficult to apply these solutions to real-time CAC decisions. This paper presents a cell-loss ratio evaluation algorithm that takes the peak and the average cell rates as inputs, and provides the upper-bound of the cell-loss ratio. The most remarkable characteristic of this algorithm is that it does not require exhaustive calculation and its calculation load is independent of the number of VCs and the number of different VC types. Using this approximation, we propose a real-time CAC. The experimental results show that call processing of the proposed CAC using a processor, whose performance is almost the same as that of a processor in a conventional PBX, terminates within several milliseconds.

key words: *ATM, CAC, cell-loss ratio, nonparametric approach*

1. Introduction

An essential feature of an Asynchronous Transfer Mode (ATM) based solution for B-ISDN is its potential to use the same set of network resources to support a variety of user services [1]. With ATM technology, networks will support many types of traffic generated by a variety of services and they must guarantee quality of service (QoS). Controlling traffic is therefore a vital technology in attaining such requirements. Connection Admission Control (CAC) is particularly important and an issue peculiar to services that guarantee QoS.

The principle of CAC is as follows: at the time of connection setup, a user specifies QoS requirements and anticipated traffic characteristics using a source traffic descriptor. The network allocates resources for the connection based on these requirements and the source traffic descriptor values, rejecting the connection if there are not enough network resources available. The QoS requirements are expressed in terms of cell-loss ratio (CLR) and/or cell-transfer delay (CTD). This paper deals with CLR [2].

There are still several challenging problems using CAC with ATM networks. One of these is how to assign sufficient bandwidth for cell arrival processes that are fast enough to achieve real-time CAC decisions.

The problem of assigning resources has been studied by many researchers. The solutions proposed for this problem so far require many calculations, which increases with the number of VCs or the number of different VC types. So, applying these solutions to real-time CAC is difficult.

This paper extends previously published studies for estimating sufficient resources with a light calculation load suitable for real-time CAC. We derive a good approximation for CLR estimation from the source traffic descriptor values.

It gives a sufficiently accurate upper bound for CLR with a light calculation load. Using this approximation, we propose a real-time CAC whose calculation load is independent of the number of VCs and the number of VC traffic types. The experimental results show that CLR evaluation for call processing of the proposed CAC using a processor, whose performance is almost the same as that of a processor in a conventional PBX, terminates within several milliseconds.

This paper is organized as follows. In Sect. 2, we define some terminology used in this paper and briefly review related work. In Sects. 3 and 4, we give the approximation for CLR evaluation, which does not require convolution. Specifically, we introduce a Bellows-like Poisson distribution for the evaluation, which turns out to be efficient at reducing the calculation load. In Sect. 5, we show the validity of the Bellows-like Poisson distribution through numerical examples. Characteristics, that is, sufficient accuracy and a light calculation load, of the proposed algorithm are discussed. Finally, based on the CLR evaluation in Sect. 4, we propose a real-time CAC in Sect. 6.

2. Background

First, we define the terminology used in this paper. There are one or more VPs accommodated in a transmission link. Each VP is assumed to have a rigid boundary, or in other words, it is guaranteed that each VP is assigned a fixed bandwidth. A typical scheme for assuming this property is to apply periodic services to each VP. It is possible that the transmission link is used as one VP. We define VCs accommodated in the

Manuscript received February 28, 1995.
Manuscript revised August 1, 1995.

† The author is with NTT Telecommunication Networks Laboratories, Yokosuka-shi, 238–03 Japan.
†† The author is with NTT Business Communications Headquarters, Tokyo, 108–19 Japan.

VP as being of the same QoS class. We define type as the traffic characteristic specified by traffic descriptor values. If VCs are described by the same traffic descriptor, we define them as being of the same type of traffic.

A nonparametric approach [2], [3] to CLR evaluation has been proposed for assigning bandwidth to any cell stream using the values of the source traffic descriptor (peak and average cell-rate). The decisions of this CAC are based on the upper bound of the CLR derived from the source traffic descriptors without assuming any particular parametric model for the cell arrival process.

In order to model the actual cell arrival process, there are many parametric models proposed in previous research, for example, MMPP [4]. These models are of actual cell arrival processes pertaining to the source traffic itself (voice, video, and so on) or its superposition. When using real-time CAC decision, however, these parametric models have the following difficulties:

1. Although, theoretically, the actual complex cell arrival process can be reproduced if many parameters are introduced and are set to appropriate values, it is almost impossible in practical application.

2. If these parameters are given appropriate values, Usage Parameter Control (UPC) cannot decide whether or not the actual cell arrival process complies with these parameters. For example, the interval variance between the arrival of cells cannot be controlled by UPC.

Therefore, if we assume the actual cell arrival process complies to a certain parametric model (e.g., MMPP with some number of states), it has no meaning when the assumption is not reasonable. UPC cannot decide whether the assumption is reasonable or not. Accordingly, it is desirable that CLR evaluation for CAC decisions be carried out using only parameters that can be controlled by UPC. If not, the framework that controls quality using UPC will fail. The nonparametric approach satisfies the above requirement.

Due to the fact that UPC can control the burst size of the source traffic, CLR evaluation should take this into account, from the viewpoint of effective utilization of the network. However, this evaluation requires solving a simultaneous equation or recursion with a number of variables corresponding to the buffer capacity, making it unsuitable for real-time CAC decisions. Therefore, this paper focuses on CLR evaluation using only the peak and average cell-rate.

In the nonparametric approach, time is divided into fixed-length slots each of which corresponds to the transmission time of an ATM cell. The slot length is defined as L/C, where L is the cell length [bits] and C is the capacity of the VP [bps]. There are n VCs in the VP, and each VC is indexed by i $(i = 1, 2, \ldots, n)$. Let $p_i(k)$ denote the probability that k cells from VC i arrive at the output queue during an observation interval of γ slots. The upper-bound for CLR B_{np} of the VP is given by

$$B_{\mathrm{np}} = \frac{\sum_{k=0}^{\infty} [k-\gamma]^{+}\, p^{(n)}(k)}{\sum_{k=0}^{\infty} k p^{(n)}(k)}, \tag{1}$$

where

$$\begin{aligned} p^{(n)}(k) &= \{p_1 \star p_2 \star \cdots \star p_n\}(k), \\ [x]^{+} &= \max(0, x). \end{aligned}$$

For an output buffer size of K, γ is chosen to be $K+1$ [5]. The binomial operator $\star$ denotes the convolution of the form $f \star g(k) = \sum_l f(l) g(k-l)$. To attain the worst case estimate, $p_i(k)$ is defined, using only traffic descriptors, as

$$p_i(k) = \begin{cases} 1 - A_i/R_i & k = 0, \\ A_i/R_i & k = R_i, \\ 0 & \text{otherwise}. \end{cases} \tag{2}$$

In this expression, R_i and A_i denote the maximum and average number of cells arriving during γ slots from the i-th VC, respectively, and are expressed as

$$R_i = \lceil r_i \gamma L/C \rceil, \quad A_i = a_i \gamma L/C, \tag{3}$$

where $\lceil x \rceil$ denotes the minimum integer greater than or equal to x, and r_i and a_i are the peak and average cell rates of the i-th VC, respectively.

Notice that traffic model (2) is independent of the burst size. This model represents a long burst-length limit which consists of peak and average cell rates. It is known that this model is the worst pattern for maximizing (1) [3].

Here, the relationship between the previously described CLR evaluation and UPC is briefly described. First, a case is described for a Sliding Window type or Jumping Window type of UPC. The peak rate observation is valid if the window size is less than or equal to γ [5]. The average rate observation prefers a large window size. However, since A_i is in terms of an infinite time average, we accept that the average rate observation with a finite window size is more strict. Second, a case is described for a Dual Leaky Bucket type of UPC defined as a GCRA [6] in the ATM Forum. The peak rate observation has no problems, and the average rate observation makes the cell stream average become less than or equal to A_i, taking into consideration a long time period. Regardless of the UPC mechanism, if an appropriate UPC, meaning that the actual peak and average cell-rate are less than or equal to the negotiated ones, is used, we can obtain the upper-bound CLR using (1) [7].

We assume that there are no cell delay variations. However, even when there are, equations similar to (3) are still applicable [8] and the following discussion is valid. To simplify the explanation, this paper does not take cell delay variations into consideration.

Calculation of the upper-bound CLR $B_{\rm np}$ using (1) involves convolution. Therefore, the calculation load increases rapidly as the number of VCs or number of VC types increase. Suppose there are m classes and the number of class m VCs is n_m; then the calculation complexity of (1) is on the order of

$$O\left(\prod_{j=1}^{m} n_j\right), \tag{4}$$

and it increases exponentially when m is large.

To achieve real-time CAC, we must reduce the calculation load for the CLR estimation. With an appropriate UPC of the peak and average cell rate, $B_{\rm np}$ gives the correct upper bound for CLR [3]. Therefore, our goal is reducing the calculation load without decreasing the accuracy of the nonparametric approach. We achieve this by introducing several effective heuristics with which evaluation is given in terms of a closed form function, diminishing the number of convolution calculations.

Our discussion begins with a simple case where all the VCs in a class have the same peak rate, then we extend the results to a more general case.

3. Preliminary

To reduce the calculation times for CLR evaluation, we start from a simple case where there are many VCs and all VCs are of the same type.

We suppose that all VC i $(i = 1, 2, \ldots, n)$ are of the same type. We assume that each one has a distribution of the form in (2) and that parameter values are given by $R_i = R$ and $A_i = A$. The distribution $\{p^{(n)}(k);\ k = 0, 1, \ldots\}$, which is the probability of k cells arriving during γ slots, for the aggregate traffic is thus given by

$$p^{(n)}(k) = \begin{cases} {}_nC_{k/R}\left(\frac{A}{R}\right)^{\frac{k}{R}}\left(1-\frac{A}{R}\right)^{n-\frac{k}{R}} & \\ & k = 0, R, 2R, \ldots, \\ 0 & \text{otherwise.} \end{cases}$$

We take $n \to \infty$ while keeping nA constant. Then

$$p^{(n)}(kR) \longrightarrow \exp\left(-nA/R\right)\frac{(nA/R)^k}{k!}.$$

This suggests that it is reasonable to approximate $p^{(n)}(kR)$ using the Poisson distribution $P_\lambda(k) = \exp(-\lambda)\,\lambda^k/k!$ with parameter $\lambda = nA/R$ for $n \gg 1$. It should be noted that the Poisson distributions in this paper do not describe input processes.

Next, we consider a case where all VCs accommodated in a VP have the same peak cell rate; $R_i = R$. We assume the number of type j $(j = 1, 2, \ldots, m)$ VCs is n_j and that the traffic of each VC is characterized by A_j and R. We define the distribution $\{p_j^{(n_j)}(k);\ k = 0, 1, \ldots\}$, which is the probability that the number of type j cells arriving during γ slots is k, and $A = \sum_j n_j A_j$. We take $n_j \to \infty$ while keeping $n_j A_j$ constant. Then

$$p_j^{(n_j)}(kR) \longrightarrow \exp\left(-n_jA_j/R\right)\frac{(n_jA_j/R)^k}{k!},$$

where k is a non-negative integer. We calculate the convolution of $p_j^{(n_j)}$ for $n_j \to \infty$ as

$$\begin{aligned} \widehat{p}(kR) &\equiv \{\widehat{p}_1 \star \widehat{p}_2 \star \cdots \star \widehat{p}_m\}(kR) \\ &= \exp\left(-A/R\right)\frac{(A/R)^k}{k!}, \end{aligned} \tag{5}$$

where

$$\widehat{p}_j(k) = \lim_{n_j \to \infty} p_j^{(n_j)}(k).$$

Therefore, we can also approximate $\widehat{p}(kR)$ to the Poisson distribution $P_\lambda(k) = \exp(-\lambda)\lambda^k/k!$ with parameter $\lambda = A/R$ for $n_j \gg 1$. Note that the derivation of (5) is valid when all R_j are equal to R.

Using (5), the CLR evaluation formula is obtained by replacing $p^{(n)}(k)$ in (1) with $\widehat{p}(k)$. The CLR $\widehat{B}$ in this approximation is expressed as

$$\widehat{B} = \frac{1}{A}\sum_{k=0}^{\infty}\left[kR - \gamma\right]^+ \widehat{p}(kR). \tag{6}$$

The calculation time for (6) depends on the convergence speed of this series. The speed directly depends on the values R and A, but does not directly depend on the number of VCs or the number of different VC types.

The result can be applied to CAC in the following way. VCs of the same peak rate are grouped together and accommodated in separate VPs. A CAC decision is applied individually to each VP in isolation. Classifying traffic into classes of uni-peak types reduces the complexity of the traffic control. We can control the admission of VCs by taking only the average cell rate of each VC into consideration. Consequently, it is sufficient for CAC to only handle the offered load of each VP. This approach gives an accurate upper bound of CLR in the case of many VCs. In this case, the previous methods using convolution require heavy calculation. Therefore, this approach is effective in reducing the calculation load.

4. CLR Evaluation for a Class of Multiple Types of Traffic

4.1 Bellows-Like Poisson Distribution

The CLR evaluation in (6) is only applicable where all types of VCs in a VP have the same peak rate. In this section, the previous results are extended to deal with a case where VCs with various peak rates are multiplexed into a VP. As preparation for this extension, we define a Bellows-like Poisson distribution.

Definition: Bellows-like Poisson distribution $\{\Theta(\lambda, a, b;\ k);\ \lambda,\ a,\ b : \text{constant},\ k = a\ell + b\ (\ell = 0, 1, 2, \ldots)\}$ is defined as

$$\Theta(\lambda,\ a,\ b;\ k) = \begin{cases} P_\lambda(\ell) & k = a\ell + b, \\ 0 & \text{otherwise}, \end{cases} \tag{7}$$

where $P_\lambda(\ell) = \exp(-\lambda)\,\lambda^\ell/\ell!$.

The Bellows-like Poisson distribution $\Theta(\lambda,\ a,\ b;\ k)$ is an extension of Poisson distribution $P_\lambda(\ell)$, such that $\ell\ (= 0, 1, 2, \ldots)$ is transformed by scale-parameter a and shift-parameter b. $\Theta(\lambda,\ a,\ b;\ k)$ is reduced to Poisson distribution $P_\lambda(k)$ when $a = 1$ and $b = 0$. Distribution $\widehat{p}(kR)$ in (5) is an example of a Bellows-like Poisson distribution, namely

$$\widehat{p}(k) = \Theta(A/R, R, 0; k).$$

Our strategy for this extension is as follows: we assume that it is valid to use a Bellows-like Poisson distribution for CLR evaluation as shown in the previous section. A Bellows-like Poisson distribution has desirable properties for efficient CLR evaluation. These properties are:

1. A CLR evaluation using a Bellows-like Poisson distribution is easy to calculate because a series for computing the expectation value converges as fast as a series for a Poisson-distributed random variable, with respect to the summation of k in (7). Therefore, using a Bellows-like Poisson distribution reduces the calculation load.
2. A Bellows-like Poisson distribution has three degrees of freedom, which are parameters λ, a, and b. Therefore, we can handle even high order moments of this distribution.

According to this strategy, we expect the extension using a Bellows-like Poisson distribution to give an accurate CLR evaluation with a light calculation load.

The remainder of this section shows how CLR evaluation is accomplished using the above strategy.

First, we show the effect of parameter b in $\Theta(\lambda, a, b; k)$ using the relationship between parameter b and γ. We consider two distributions of the number of cell arrivals during γ slots. One is $\{p^{(c)}(k);\ k = 0, 1, 2, \ldots\}$ of a VC, with $R^{(c)} = A^{(c)}$, which is a special case of (2); that is

$$p^{(c)}(k) = \begin{cases} 1\ (= A^{(c)}/R^{(c)}) & k = R^{(c)}, \\ 0 & \text{otherwise.} \end{cases} \tag{8}$$

From definition (3), although $R^{(c)}$ of the former VC has been assumed to be an integer, we can make the same interpretation as stated previously when $R^{(c)}$ is not an integer. In addition, we permit $R^{(c)}$ and $A^{(c)}$ to be negative. The other is $\{q(k);\ k = 0, 1, 2, \ldots\}$ of aggregate traffic for arbitrary VCs. We replace distributions of the number of cell arrivals in (1) with $\{q \star p^{(c)}\}(k)$. The numerator of (1) can then be expressed as

$$\begin{aligned} &\sum_k [k-\gamma]^+\, \{q \star p^{(c)}\}(k) \\ &= \sum_k [k-\gamma]^+\, q(k - A^{(c)}) \\ &= \sum_k [k - (\gamma - A^{(c)})]^+\, q(k). \end{aligned} \tag{9}$$

Therefore, the former VC varies parameter γ to $\gamma - A^{(c)}$, or in other words, varies the bandwidth of VP corresponding to $A^{(c)}$.

If distribution $q(k)$ is a Bellows-like Poisson distribution such that

$$q(k) = \Theta(\lambda, a, b; k),$$

then $\{q \star p^{(c)}\}(k)$ is also a Bellows-like Poisson distribution; that is

$$\{q \star p^{(c)}\}(k) = \Theta(\lambda, a, b + A^{(c)}; k).$$

Therefore, from (9), we have

$$\begin{aligned} &\sum_k [k-\gamma]^+\, \Theta(\lambda, a, b + A^{(c)}; k) \\ &\quad = \sum_k [k - \gamma + A^{(c)}]^+\, \Theta(\lambda, a, b; k). \end{aligned} \tag{10}$$

Next, we show the relationships between parameters λ, a, and b and the moments of Bellows-like Poisson distribution $\Theta(\lambda, a, b; k)$. Due to the fact that we introduce $\Theta(\lambda, a, b; k)$ as the distribution of the number of arriving cells, we restrict the range of parameters: λ and $a > 0$.

We assume that the i-th VC ($i = 1, 2, \ldots, n$) has R_i and A_i, and that the distribution of the number of arriving cells is given by (2). In this case, the average C_1, the variance C_2, and the 3rd central moment C_3 of the number of arriving cells during γ slots from all VCs, that is the cumulants of $p^{(n)}(k)$ in (1), are expressed as

$$\begin{aligned} C_1 &= \sum_{i=1}^{n} A_i, \\ C_2 &= \sum_{i=1}^{n} A_i\,(R_i - A_i), \text{ and} \\ C_3 &= \sum_{i=1}^{n} A_i\,(R_i - A_i)(R_i - 2A_i). \end{aligned} \tag{11}$$

4.2 In the Case of $C_3 > 0$

For $C_3 > 0$, let us introduce a heuristic in which the distribution of the number of cell arrivals $\{\widetilde{p}(k);\ k = \ell R + \delta A\ (\ell = 0, 1, 2, \ldots)\}$ is the Bellows-like Poisson distribution

$$\begin{aligned} \widetilde{p}(k) &= \Theta(A/R, R, \delta A; k) \\ &= \begin{cases} \exp(-A/R)(A/R)^\ell/\ell! & k = \ell R + \delta A, \\ 0 & \text{otherwise,} \end{cases} \end{aligned}$$

where

$$R \equiv \frac{C_3}{C_2},\ \ A \equiv \frac{(C_2)^2}{C_3}, \tag{12}$$

and

$$\delta A \equiv C_1 - A = C_1 - \frac{(C_2)^2}{C_3}.$$

With this setup, it is easily verified that this distribution function gives the same moments as (11); that is

$$C_1 = \sum_{\ell=0}^{\infty} k\,\widetilde{p}(k),$$
$$C_2 = \sum_{\ell=0}^{\infty} (k - C_1)^2\,\widetilde{p}(k),$$
$$C_3 = \sum_{\ell=0}^{\infty} (k - C_1)^3\,\widetilde{p}(k),$$

where $k = \ell R + \delta A$.

This observation suggests that it is reasonable to replace the distribution $p^{(n)}(k)$ in (1) with $\Theta(A/R, R, \delta A; k)$. Thus, we obtain CLR evaluation formula $\widetilde{B}$ in the form

$$\widetilde{B} = \frac{1}{C_1}\sum_{k=0}^{\infty}[k-\gamma]^+\,\Theta(A/R, R, \delta A; k)$$
$$= \frac{1}{C_1}\exp(-A/R) \times \sum_{\ell=0}^{\infty}[\ell R - \gamma + \delta A]^+\,\frac{(A/R)^\ell}{\ell!}, \qquad (13)$$

where we use the relation in (10). The calculation time for (13) depends on the convergence speed of this series. This speed directly depends on the values R, A, and δA, but does not directly depend on the number or types of VCs. Moreover, C_1, C_2, and C_3 are the cumulants of distribution $\Theta(A/R, R, \delta A; k)$ or $p^{(n)}(k)$ in (1), and calculation of C_1, C_2, and C_3 requires only a minimal amount of processing because these values can be updated from previous values by applying a small number of addition and subtraction operations.

An example of how to calculate (13) is shown in the Appendix.

4.3 In the Case of $C_3 \leqq 0$

Next, we consider the case where $C_3 \leqq 0$. When $C_3 = 0$, we cannot define A using (12). In addition, when $C_3 < 0$, then $R < 0$ and $A < 0$ are obtained from (12). To avoid these problems, we redefine C_3 as a positive value.

In addition, although C_3 is positive, when it is a small value, it incurs a problem. This is because the convergence speed of series (13) depends on the value. For small values of C_3, a large number of iterations is necessary. To avoid this problem, we also redefine C_3 for not only $C_3 \leqq 0$ but also $C_3 \approx 0$.

An example of how to determine a new C_3 when $C_3 < 0$ or $C_3 \approx 0$ is shown in the Appendix.

5. Numerical Examples

5.1 Class Example of Uni-Peak Types of Traffic

Here we show the accuracy of CLR evaluation using (6). Consider two types of traffic, type-1 and type-2, both having the same peak rate of 10 Mbps/VC, where the ratio of the average rate of type-1 to that of type-2 is 2 : 1. Figure 1 shows the CLR evaluation using (6) and the upper-bound for CLR using (1) where the number of multiplexed type-1 and type-2 VCs is 50 VCs + 50 VCs, 100 VCs + 100 VCs, and 500 VCs + 500 VCs. We choose a VP capacity of 150 Mbps and an output buffer size of 100 cell places. Our approximation over estimates CLR. When there is a large number of VCs, our approximation evaluates CLR quite accurately. In this case, because (1) requires a heavy calculation load for convolution, our approximation is efficient. Moreover, our approximation uses only the average cell rate of VCs, so we can operate the network using the offered load of VPs that accommodates a uni-peak class.

5.2 Class Example of Multiple Types of Traffic

Now, we examine the accuracy of CLR evaluation using (13) and the validity of the assumptions in Sect. 4. Consider two types of traffic, type-1 and type-2. Type-1 has a peak rate of 10 Mbps and an average rate of 0.5 Mbps, type-2 has a peak rate of 1.5 Mbps and an average rate of 0.2 Mbps. We choose a VP capacity of 150 Mbps and an output buffer size of 100 cell places. Figure 2 shows the CLR evaluation using (13) and the upper-bound for CLR using (1) when the ratio of the average rates of type-1 to type-2 is 8 : 2, 5 : 5, and 2 : 8. Our approximation gives an accurate evaluation over a broad range without the need for convolution calculation.

Next, we examine the accuracy of the CLR evaluation where $C_3 \leqq 0$ as described in Sect. 4.3. Consider two types of traffic, type-1 and type-2. Type-1 has a peak rate of 10 Mbps and an average rate of 2 Mbps.

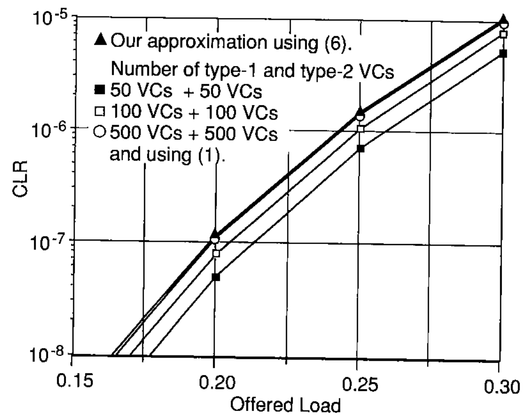


Fig. 1 Example of CLR estimation using Poisson approximation (uni-peak rate class).

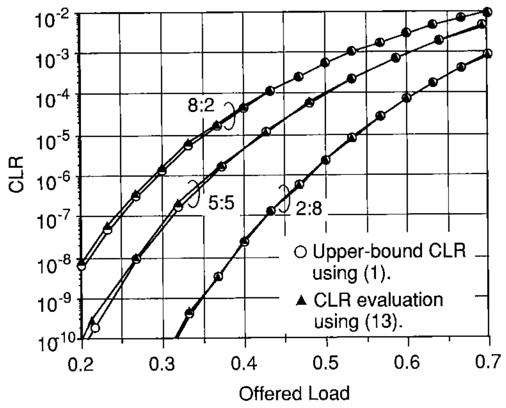


Fig. 2 Example of CLR estimation using our approximation.

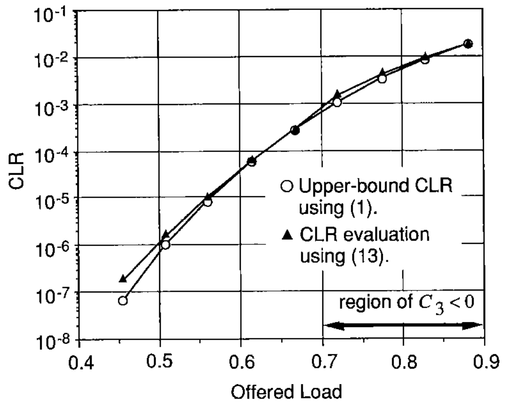


Fig. 3 Example of CLR estimation using our approximations when $C_3 < 0$.

Type-2 has a peak rate of 10 Mbps and an average rate of 8 Mbps. We choose a VP capacity of 150 Mbps and an output buffer size of 100 cell places.

Type-2 has a negative 3rd central moment and C_3 becomes negative according to an increase in the number of type-2 VCs. Figure 3 shows the CLR evaluation using the prescription described in Sect. 4.3 and the upper-bound for CLR using (1) when the number of type-1 VCs is 10 and the number of type-2 VCs is 6, 7, ..., or 14. In the region, in which the offered load is greater than 0.7, C_3 is a negative value. Our approximation using the prescription gives an accurate evaluation where $C_3 \leqq 0$.

5.3 Iteration Times for the CLR Evaluation

Figure 4 shows how the iteration times for CLR evaluation with (13) and (1) depend on the number of aggregate VCs for the two types of traffic. Type-1 and type-2 traffic have peak rates of 10 Mbps and 1.5 Mbps, average rates of 0.5 Mbps and 0.2 Mbps, respectively, and the capacity of the VP is 150 Mbps, the output buffer size is 100 cell places. The same number of type-1 and type-2 VCs are multiplexed in the VP. We measured the calculation load of (13) such that the truncation error

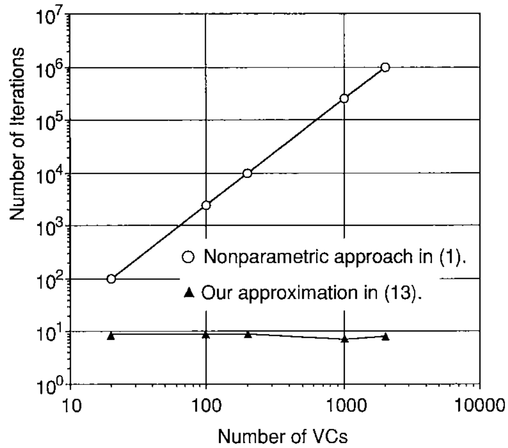


Fig. 4 Number of iterations for our CLR approximation.

Table 1 Calculation time for CLR evaluation.

iteration times N	10	20	30
with co-processor (msec)	0.6	0.8	1.1
without co-processor (msec)	2.5	3.7	4.9

for the convergence value of (13) is less than 0.03%. The number of iterations for our approximation is small and insensitive to the number of VCs.

5.4 Calculation Time for the CLR Evaluation

Here, we show the calculation time for CLR evaluation using the proposed method. We assume that a processor for call processing in SW is 68030, and its processing rate is 10 Mips. We consider two processors: one with a floating-point co-processor and the other without one.

When we calculate CLR evaluation by truncating the summation to the first 10, 20, 30 terms in (A·1), that is where N =10, 20, 30 shown in the Appendix, the resulting calculation time for both cases is shown in Table 1. The case without a co-processor is supposed as a conventional PBX case. Therefore, the results show that call processing of the proposed CAC using a processor, whose performance is almost the same as that of a processor in a conventional PBX, terminates within several milliseconds.

6. Application to CAC

Based on our CLR evaluation, we propose the following real-time CAC.

1. At the time of connection setup, a user specifies the QoS requirements and the anticipated traffic characteristics using a source traffic descriptor.

2. Calculate C_1, C_2, C_3, R, and A using (12) at each VP accommodated in the connection.

3. If C_3 does not satisfy (A·2), then re-calculate C_3.

4. Calculate the CLR evaluation using (13).
5. Compare the CLR evaluation with the QoS requirements.
6. Connection is accepted if the CLR evaluation is smaller than all the QoS requirements of VPs accommodated in the connection. Otherwise connection is rejected.

7. Conclusion

In this paper, we have presented an approximation of CLR for multi-class VCs without convolution calculations. Due to the fact that convolution is not needed, the time necessary for calculation is short and is independent of the number of VCs and the number of types of VCs accommodated in the VP. Moreover, the approximation gives an accurate evaluation of the upper bound of CLR and is applicable to CAC. Therefore, we can achieve real-time CAC using this CLR evaluation.

References

[1] CCITT SG XVIII, Recommendation I.150.
[2] H. Saito, "Call admission control using upper bound of cell loss probability," IEEE Trans. on Commun., vol.40, no.9, pp.1512–1521, 1992.
[3] H. Saito, "Teletraffic Technologies in ATM Networks," Artech House, Boston, 1994.
[4] H. Heffers and D.M. Lucantoni, "A Markov modulated characterization of packetized voice and data traffic and related statistical multiplexer performance," IEEE JSAC, vol.4, pp.856–868, 1986.
[5] S. Shioda and H. Saito, "Satisfying QOS Standard with Combined Strategy for CAC and UPC," ICC'95, Seattle, Washington, 1995.
[6] The ATM Forum, "ATM User Network Interface Specification ver. 3.0," Prentice Hall, 1993.
[7] M. Aida and H. Saito, "Traffic contract parameters and CAC guaranteeing cell-loss ratio in ATM networks," IEICE Trans., vol.E78-B, no.3, pp.336–343, 1995.
[8] H. Saito, "CDV and Bandwidth Management in ATM Networks," MICC'93, Kuala Lumpur, 1993.

Appendix: Relationship between C_3 and Iteration

This section describes the method for determining a new C_3 when $C_3 < 0$ or $C_3 \approx 0$. To begin with, we consider the following quantities:

$$\Lambda = \frac{\gamma - \delta A}{R}, \quad M = \lceil \Lambda \rceil .$$

Using Λ and M, we can rewrite (13) as

$$\begin{aligned}\widetilde{B} &= \frac{R}{C_1} \exp(-A/R) \sum_{k=0}^{\infty} (M - \Lambda + k) \frac{(A/R)^{M+k}}{(M+k)!} \\ &\equiv \frac{R}{C_1} \exp(-A/R)\, S, \qquad \text{(A· 1)}\end{aligned}$$

and we define the summation part, with respect to k, in (A· 1) as S. When we calculate (A· 1) using the summation of the first N terms, that is, N is the number of iterations, we split S into two parts:

$$\begin{aligned}S = &\sum_{k=0}^{N-1} (M - \Lambda + k) \frac{(A/R)^{M+k}}{(M+k)!} \\ &+ \sum_{k=N}^{\infty} (M - \Lambda + k) \frac{(A/R)^{M+k}}{(M+k)!}.\end{aligned}$$

We define the first term and the second term on the right-hand side as S_1 and S_2, respectively. S_2 corresponds to the error caused by truncation of the summation in (A· 1). When applying this to CAC, it is sufficiently accurate that the CLR evaluation using S_1 is in the same order of $\widetilde{B}$ using (13). Therefore, if $S_2 \leqq S_1$, we can recognize that S_1 is an efficient CLR evaluation.

S_2 is upper-bounded as

$$\begin{aligned}S_2 &= \sum_{k=0}^{\infty} (\alpha + k) \frac{(A/R)^{\beta+k}}{(\beta+k)!} \\ &< \frac{(A/R)^{\beta}}{\beta!} \sum_{k=0}^{\infty} (\alpha + k) \frac{(A/R)^k}{(\beta+1)^k} \\ &= \frac{(A/R)^{\beta}}{\beta!} \left(\frac{\alpha}{1-Y} + \frac{Y}{(1-Y)^2} \right),\end{aligned}$$

where, $\alpha = M - \Lambda + N$, $\beta = M + N$ and $Y = \frac{A/R}{\beta+1}$. On the other hand, S_1 is lower-bounded as

$$\begin{aligned}S_1 &= \sum_{k=0}^{N-1} (M - \Lambda + k) \frac{(A/R)^{M+k}}{(M+k)!} \\ &> \frac{(A/R)^{\beta}}{\beta!} \sum_{k=0}^{N-1} (M - \Lambda + k) \\ &= \frac{(A/R)^{\beta}}{\beta!} \left(N(M - \Lambda) + \frac{1}{2} N(N-1) \right).\end{aligned}$$

Therefore, a sufficient condition such that $S_2 \leqq S_1$ is

$$\frac{\alpha}{1-Y} + \frac{Y}{(1-Y)^2} \leqq N(M - \Lambda) + \frac{1}{2} N(N-1).$$

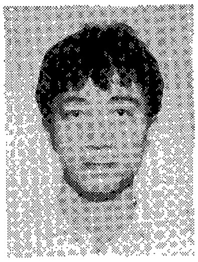
We approximate $M - \Lambda$ as 1, we can get a sufficient condition on C_3 as

$$C_3 \geqq \frac{F(N) + \sqrt{(F(N))^2 + G(N)}}{2X(N)(N-1)}, \qquad \text{(A· 2)}$$

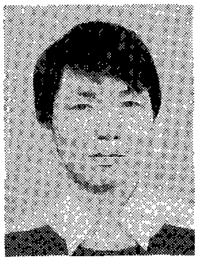
where

$$\begin{aligned}X(N) &= 1 - \frac{1 + \sqrt{3 - 2/N}}{N-1}, \\ F(N) &= C_2 X(N)(\gamma - C_1), \\ G(N) &= 4X(N)(1 - X(N))(C_2)^3(N-1).\end{aligned}$$

Therefore, if C_3 does not satisfy (A· 2), we give C_3 a minimum value using (A· 2).



Masaki Aida received his B.S. and M.S. degrees in Theoretical Physics from St. Paul's University, Tokyo, Japan, in 1987 and 1989, respectively. Since he joined NTT Laboratories in 1989, he has been mainly engaged in research on traffic control in ATM-LAN. He is currently a Research Engineer in NTT Telecommunication Networks Laboratories.



Teruyuki Kubo graduated from Kyushu University in 1974, and received M.E. degree of Information Engineering from the same university in 1976. After joining NTT Labs. in 1976, he had been active in research and development of customer premises communication systems and networks until spring of 1995. He current works with NTT Business Communications HQs and has interests in ATM LANs and internetworking technologies.