論文 94-19-4-18

A Study on Real Time and Non-real Time Traffic Multiplexing with Congestion Control

Kwan Ho Song*, Jae Ho Lee* *Regular Members*

폭주제어를 포함한 실시간 및 비실시간 트래픽의 다중화에 관한 연구

正會員 宋 官 浩* 正會員 李 在 昊*

Abstract

In this paper we propose a multiplexing scheme of real time and non-real traffics in which a congestion control is embedded. Real time traffics are assumed to be nonqueuable and have preemptive priority over non-real time traffics in seizing the common output link, whereas the non-real time traffics wait in the common buffer if the output link is not available for transmission. Real time traffics are encoded according to the bandwidth reduction strategy, particularly when congestion occurs among non-real time traffics. This scheme provides us an efficient way for utilizing the costly bandwidth resources, by accommodating as many real time traffics as possible with gauranteeing its mimimun bandwidth requirements, and also resloving the congestion encountered among non-real time traffics. We describe the system as a Markov queueing system, provide the analysis by exploiting the matrix geometric method, and present the performance for various performance measures of interest. Some numerical results are also provided.

요 약

본 논문에서는 폭주제어 기능을 내장한 실시간 및 비실시간 트래픽의 통합방식을 제안한다. 실시간 트래픽은 대기능력이 없으며, 출력 링크를 점유함에 있어 비실시간 트래픽보다 우선순위를 갖는다고 가정한다. 출력 링크가 모두 점유되어 있을시 도착하는 실시간 트래픽은 망의 수용이 불가능하여 손실되며, 반면 비실시간 트래픽은 버퍼에 저장되어 추후에 서비스를 받을 수 있도록 한다. 실시간 트래픽은 대역폭 축소 전략에 따라 인코딩되며, 이 전략은 비실시간 트래픽에서의 폭주가 증가함에 따라 강화된다. 본 제안 방식은 실시간 트래픽의 최소 대역폭 요구를 만족시키면서 실시간 트래픽의 수를 최대로 허용하고, 또한 비실시간 트래픽간에 발생하는 폭주를 감소시킴으로서, 통신 대역폭을 효율적으로 사용할 수 있는 방식이다. 제안한 방식에 따른 트래픽 통합 방식을 Markov 시스템으로 모델링하였으며, matrix geometric 방식을 이용하여 수학적인 분석을 수행하였다. 또한 관심의 대상이 되는 성능인자에 대한 성능분석을 수행하였으며, 이에 대한 수치적인 결과도 제공하였다.

*光云大學校 電子通信工學科
論文番號 : 9404
接受日字 : 1994年 1月 8日

I. Introduction

We consider a situation in which a bandwidth resource multiplexes heterogeneous traffics that request a different amount of network bandwidths. This could represent an intergrated terminal generating different types of traffics, or a multiplexing /switching unit accepting traffics with different characteristics. A control mechanism is necessary to regulate the input stream of mixed traffics, thus to achieve the efficient use of the bandwidth of the common ouput link.

This kind of control problem can be formulated as a dynamic optimization problem in which one would find the structural form of the optimal access control policy with an appropriate set performance measures [1], such as maximizing the system throughput constrained by a given bounded delay for a certain traffic [6]. In most cases, however, it is known that finding an optimal policy is tremendously, although not possible, difficult, and also not practical due to the computational complexity involved.

For this reason, many researchers have dealt with this control problem by adopting heuristic schemes. In this approach, one first considers as access control policy that is intuitively appealing easily implementable, then evaluates performances of the system under this policy. Several schemes have been proposed, and some representatives are the Complete Sharing (CS) scheme, the Complete Partitioning (CP) scheme, the Movable Boundary (MB) scheme, and the Bandwidth Reduction (BR) scheme. The CS is an uncontrolled, where the incoming traffics share the total bandwidths on a First-In-First-Out (FIFO) basis. In the CP scheme, the bandwidth resource is divided into partitions, each of which is dedicated to a particular traffic. These two schemes, although simple, become inefficient in using the bandwidth under unbalanced load conditions. For instance, in the CS schems, the traffic condition loaded with several narrowband traffics(traffics that require a small amount of bandwidths) may leave insufficient bandwidth for accommodating wideband traffics (traffics that require a large amount of bandwidths), resulting in bandwidth inefficiency. In the CP scheme, sometimes called fixed boundary scheme, on the other hand, the bandwidths for a certain traffic may be exhausted, resulting in substantial amount of delay, while the bandwith for the other traffic remain unused. Among alternatives that overcome this bandwidth inefficiency are the movable boundary and bit rate compression schemes, which are in fact the variations of the CP scheme. In the MB sheme, a traffic is allowed to use the bandwidths that originally allocated to the other traffic, but can be preempted by the latter [4, 8]. In the BR scheme [2], it is assumed that a cerain class of traffics, such as voice and video, is tolerant to bandwidth reduction to a certain extent, thus can be transmitted at a lower bit rate whitout loss of information for its transaction. Traffics are normally transmitted without compression in lightly loaded condition, however the band-width reduction strategy along data compression is executed congestion becomes serious. Embedded voice coding (see the references in [9]) and subband video coding (see the references in [3]) are examples employed in the BR scheme.

In this paper, we consider a multiplexing scheme of real time and non-real time traffics in which a congestion control is embedded by adopting the bandwidth reduction strategy for real time traffics. Real time traffics are assumed to be nonqueueable and have preemptive priority over non-real time traffics in seizing the common output link, whereas the non-real time traffics wait in the common buffer if the output link is not available for transmission. The underlying idea of the proposed bandwidth reduction scheme is as follows. Real-time traffics are encoded with as much bandwidth as possible to carry a sufficiently large amount of informations ; but, the bandwidths assigned in this fashion are larger than the predetermined minimum bandwidth requirements. Furthermore,

if the number of non-real time traffics present in the buffer exceeds the given threshold, the real time traffics are transmitted with the minimum bandwidths that guarantee the certain level of quality of the real time traffic at the receiving end. And the bandwidths extracted from real time traffics in this way are offered to non-real time traffics, thus resolving the congestion occurred among the non-real time traffics. In consequence, this scheme provides us an efficient way for utilizing the costly bandwidth resources, by accommodating as many real time traffics as possible with gauranteeing its mimimun bandwidth requirements, and also resloving the congestion encountered among non real time traffics. This proposed scheme is relatively simple compared to the one reported earlier [3], thus we believe it can be easily implementable and suitable as a congestion control scheme in a high speed environment.

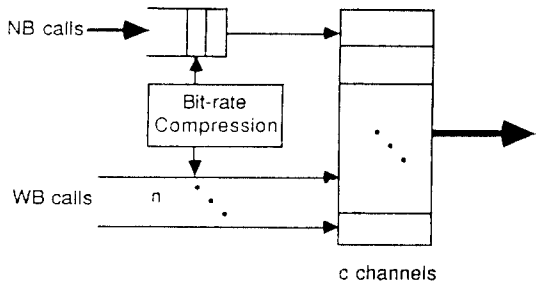


Fig 1. A traffic multiplexer

In section 2, we formulate the problem as a Markov queueing model, and propose the congestion control algorithm. Then, the model is analyzed in section 3 by exploiting the matrix geometric method with some performance measures provided. In the subsequent sections, some numerical computations for the performance measures of interest are presented, followed by concluding remark.

II. Problem Formulation

Figure 1 shows a multiplexer for intergrating real time and non-real time traffics. In this figure, n is the number of lines for WB calls, i.e., the maximum number of WB calls that can be simultaneously accommodated in the system. From now on, we assume that the real time traffics require larger bandwidth than the non-real time traffics, so that for convenience we call the former as the wideband (WB) calls and the latter the narrowband (NB) calls. Let b_1 and $b_2(>b_1)$ be the bandwidth requirement of NB and WB calls respectively, and c be the bandwidth capacity of the common output link. Note that the bandwidth requirement b_2 can be reduced to a certain amount as the number of NB calls in the buffer increases. However, it cannot be reduced below m, which is the minimum bandwidth requirement gauranteeing a certain level of quality of real time traffics (WB calls) at the receiving end. On the other hand, the bandwidth requirement of NB calls is fixed to b_1 in any case. NB calls may join a common buffer shared by all NB calls if ouput link is not available transmission. NB calls constitutes a Poisson arrivals with rate λ_1, and the service time is exponentially distributed with rate μ_1. We also assume that the time in active for WB calls and the idle time, i.e., the time interval between two adjacent active WB calls, have exponential distribution with rate μ_2 and λ_2, respectively. And all the stochastic processes involved in the model are assumed to be mutually independent.

In the following we describe the congestion control algorithm in the proposed multiplexing scheme. We define the light load as the traffic condition in which there are enough rooms to accommodate the arriving NB and WB calls, and the heavy load otherwise. During light load condition, the incoming WB calls are accepted without bandwidth reduction as well as NB calls. During heavy load condition, on the other hand,

WB calls have proeemptive priority over NB calls, and the preempted NB calls are assumed to be stored in the common buffer for future service. The bandwidth requirement of NB calls is not affected in any case. However, the bandwidth requirement of WB calls can be changed depending on the congestion level of NB calls present in the buffer. Namely, if the the number of NB calls present in the buffer does not exceed the given threshold value h, then WB calls are transmitted with reduced bandwidth, but as large as possible in order to carry as much information as possible. And, otherwise, WB calls are carried with minimum bandwidth m, and the bandwidths thus extracted from WB calls are offered to serve more NB calls, thus can resolve the congestion among NB calls. Let (i, j) be the state of the system, with i and j denoting the number of NB calls and WB calls in the system respectively, then they take values $i=0, 1, \cdots$, and $j=0, 1, \cdots, n$. And n is the maxmum number of WB calls that can be accommodated to the multiplexer. This congestion control algorithm is summerized as follows :

IF $b_1 i + b_2 j \le c$,
  THEN send all NB and WB calls with bandwidth b_1 and b_2, respectively.
  ELSE
   IF $i - d \le h$,
    THEN send WB calls with calls with b'_2 and NB calls with b_1 througth the remained channels.
    ELSE and WB calls with m and NB calls with b_1 througth the remained channels.

where $b'_2 = \min\left\{ b_2, \max\left\{ m \left\lceil \frac{c}{j} \right\rfloor \right\} \right\}$ explains the reduced bandwidth of WB calls, and $d = \left\lceil \frac{c - b'_2 j}{b_1} \right\rfloor$ is the number of NB calls that can be accommodated simultaneously for the remained bandwidth obtained after allocating WB calls. $\lceil \cdot \rfloor$ is a floor operator that takes the greatest integer value among the ones smaller than its argument.

The first IF condition determines whether the current bandwidth requirement of the system resides within the output link capacity. If it does, then both the NB and WB calls are sent without compression with their normal bandwidth requirements, b_1 and b_2 respectively, since enough bandwidths are available to accommodate them. Otherwise, the second IF statement is evaluated to determine whether the number of NB calls in the buffer exceed the given threshold value h, i. e., to determine the congestion level among NB calls. If the queue size of NB calls is not greater than h, we see that the congestion among NB calls is not serious. In this case, when the number of WB calls are large enough to outweigh the output link capacity (i.e., $jb_2 > c$), then the bandwidth of each of WB calls is compressed in such a way as to accommodate as many WB calls as possible, but not below the level of its minimum bandwidth requirement of m. The expression of b'_2 above explains this notion. If the queue size of NB calls exceeds h, then it is assumed that congestion becomes serious and each of WB calls is transmitted with is minimum bandwidth requirement m. In any case, the remained bandwidth after allocating WB calls are used to accommodate the NB calls, and the WB calls that cannot be accommodated due to the resource limitation (i.e., when $jb'_2 > c$) are blocked to enter the system.

Ⅲ. Analysis and Numerical Computation

We can describe the system as a continuous-time Markov process, and establish the state transition diagram as shown in Figure 2. Considering the algorithm described above, the transition rate from a certain (source) state to the (destination) state, the number of whose NB calls is one less than the source state (i.e., the transition rate marked in exclamation in Figure 2) is computed by

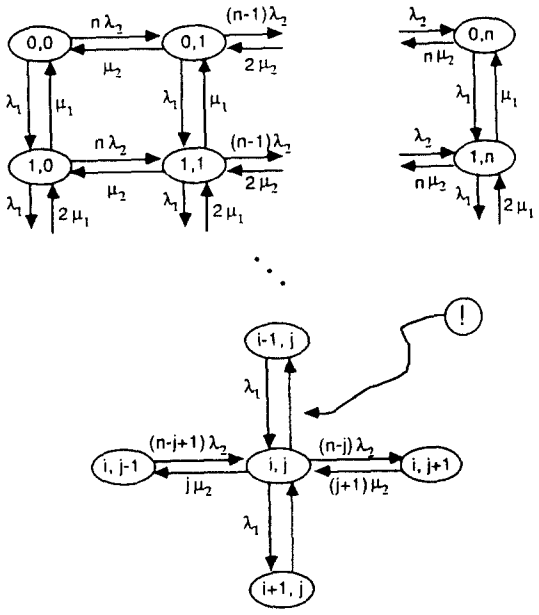


Fig 2. State transition diagram

$$\begin{cases} i\mu_1 & \text{, if } b_1 i + b_2 j \le c \\ \min\left(i, \lceil \frac{c - b_2' j}{b_1} \rfloor\right)\mu_1 & \text{, if } b_1 i + b_2 j > c \text{ and } (i-d) \le h \\ \min\left(i, \lceil \frac{c - mj}{b_1} \rfloor\right)\mu_1 & \text{, if } b_1 i + b_2 j > c \text{ and } (i-d) > h \end{cases}$$

where $b_2' = \min\left\{ b_2, \max\left\{ m, \lceil \frac{c}{j} \rfloor \right\} \right\}$.

For the stability of the system, we assume that $\lambda > \mu$. Let p_{ij} be the steady state probability when system in state (i, j). For the continuous-time Markov chain, the following equation holds

$$\pi \boldsymbol{Q} = 0, \ \pi = (\boldsymbol{p}_0, \boldsymbol{p}_1, \cdots) \qquad (1)$$

where 0 is an infinite dimensional row vector with all its elements 0. $\boldsymbol{p}_0, \boldsymbol{p}_1, \cdots$, are the steady state probability row vectors. The steady state probability vector $\boldsymbol{p}_i$ is composed of n steady state probabilites along the i-th row in the state transition diagram in Figure 2, i.e., $\boldsymbol{p}_i = (p_{i0}, p_{i1}, \cdots, p_{in})$. From the state transition diagram, we obtain the infinitesimal generator of the Markov chain which is of the following block partitioned form

$$\boldsymbol{Q} = \begin{bmatrix} A_0 & B & & & & \\ C_1 & A_1 & & & & \\ & C_2 & & & 0 & \\ & & \ddots & & & \\ & & & B & & \\ & & & A_{l-1} & B & \\ & 0 & & C_l & A_l & \\ & & & & C_{l+1} & \\ & & & & & \ddots \end{bmatrix}$$

Each component of infinitesimal generator $\boldsymbol{Q}$ is a square matrix of dimension n, whose i-th row and j-th column element is given by

$$B(i, j) = \begin{cases} \lambda_1 & \text{, } i = j \\ 0 & \text{, otherwise} \end{cases}$$

$C_k(i, j) =$

$$\begin{cases} k\mu_1 & \text{, } i = j, \text{ and } b_1 k + b_2 j \le c \\ \min\left(i, \lceil \frac{c - b_2' j}{b_1} \rfloor\right)\mu_1 & \text{, } i = j, \ b_1 k + b_2 j > c \text{ and} (i-d) \le h \\ \min\left(i, \lceil \frac{c - mj}{b_1} \rfloor\right)\mu_1 & \text{, } i = j, \ b_1 k + b_2 j > c \text{ and} (i-d) > h \\ 0 & \text{, otherwise} \end{cases}$$

where $k = 1, 2, \cdots, l$, and $d = \lceil \frac{c - b_2' j}{b_1} \rfloor$.

$A_k(i, j) =$

$$\begin{cases} (n-i)\lambda_2 & \text{, } i = j-1 \\ i\mu_2 & \text{, } i = j+1 \\ -[\lambda_1 + C_k(i, i) + (n-i)\lambda_2 + i\mu_2] & \text{, } i = j \\ 0 & \text{, otherwise} \end{cases}$$

where $k = 0, 1, \cdots, l$.

where h is the threshold value of the number of NB calls in the queue excluding the ones in service, and l is the maximum number of NB calls that can be accommodated simultaneously in the system and given by $l = \lceil \frac{c}{b_1} \rfloor$.

For easy understanding of the state transition diagram and the composition of the infinitesimal generator, we provide a simple example, where

we set the values of the parameters as follows : $b_1=1$, $b_2=3$, $m=2$, $c=7$, $h=1$, and $n=2$. With these values, $b_2'=\min\left\{b_2,\ \max\left\{m,\ \left\lceil\frac{c}{j}\right\rceil\right\}\right\}$ becomes 3 regardless of j, and $d=c-b_2'j$ becomes 7, 4, 1 for $j=0$, 1, 2, repectively. Then, the state transition diagram can be depicted as shown in Figure 3 and the infinitesimal generator of this example becomes

$$\begin{bmatrix}
-(\lambda_1+2\lambda_2) & 2\lambda_2 & 0 \\
\mu_2 & -(\lambda_1+\lambda_2+\mu_2) & \lambda_2 \\
0 & 2\mu_2 & -(\lambda_1+2\mu_2) \\
\mu_1 & 0 & 0 \\
0 & \mu_1 & 0 \\
0 & 0 & \mu_1
\end{bmatrix}$$

$$\begin{bmatrix}
\lambda_1 & 0 & 0 \\
0 & \lambda_1 & 0 \\
0 & 0 & \lambda_1 \\
-(\lambda_1+\mu_1+2\lambda_2) & 2\lambda_2 & 0 \\
\mu_2 & -(\lambda_1+\mu_1+\lambda_2+\mu_2) & \lambda_2 \\
0 & 2\mu_2 & -(\lambda_1+\mu_1+2\mu_2) \\
2\mu_1 & 0 & 0 \\
0 & 2\mu_1 & 0 \\
0 & 0 & \mu_1 \\
 & & \ddots
\end{bmatrix}$$

Note that since $n=3$ the dimension of all submatrices is 3 and the submatrices of $\boldsymbol{Q}$ become

$$A_0=\begin{bmatrix}
-(\lambda_1+2\lambda_2) & 2\lambda_2 & 0 \\
\mu_2 & -(\lambda_1+\lambda_2+\mu_2) & \lambda_2 \\
0 & 2\mu_2 & -(\lambda_1+2\mu_2)
\end{bmatrix},$$

$$C_1=\begin{bmatrix}
\mu_1 & 0 & 0 \\
0 & \mu_1 & 0 \\
0 & 0 & \mu_1
\end{bmatrix}, \text{ and so on.}$$

Visiting each state of the state transition diagram from left to right and from top to bottom, the bandwidth reduction for the WB call is applied to the state (3, 2), since $b_1i+b_2j=9$ which exceeds the channel capacity $c=7$, and $i-d=3-1=2$ exceeds the threshold $h=1$. Thus, the bandwidth for the WB call is reduced to its minimum bandwidth requirement of $m=2$, and the extracted bandwidths in this way of the amount of 2(1 for each of two WB calls) are allocated to NB calls. Thus, total of 3 NB calls, one originally allocated and two allocated to the bandwidths extracted from WB calls, are served simultaneously. Similarly, the bandwidth reduction strategy applies to all the states in the third column from the state (3, 2), and all the states in the second column down from the state (6, 1) of the state transition diagram. Also, note that the

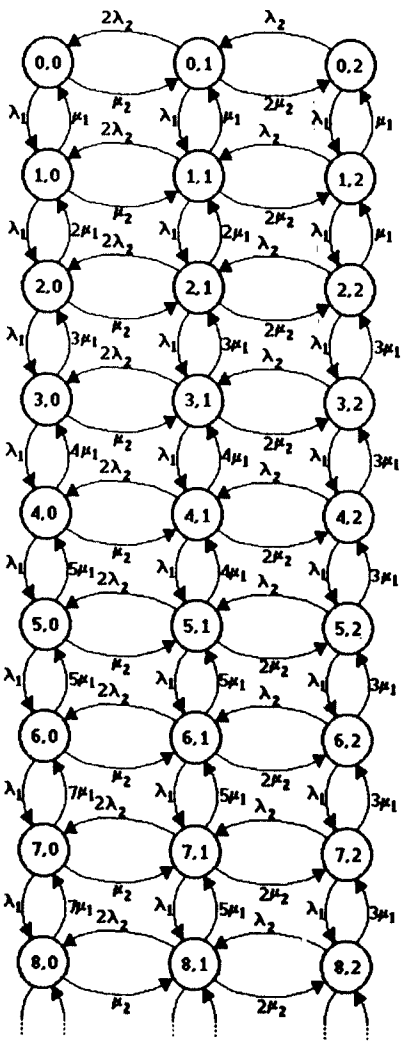


Fig 3. State trasition diagram of the exemple

state transitions in each row of the state transition diagram all the way down from 8th row are identical, which leads to submatrices B, A_l, and C_l of the same form located at each column starting from lth column of the infinitesimal generator $\boldsymbol{Q}$. In here, l is the maximum number of NB calls that can be served simultaneously when WB calls are not in the system, and is 7 in our example.

Steady state probabilities p_{ij}'s are now uniquely determined by equation (1) togeter with the normalization equation

$$\pi \boldsymbol{e} = 1 \tag{2}$$

where $\boldsymbol{e}$ is a column vector with all its elements 1. We note that the state transition diagram infinitely repeats all the way to the downward, which leads to infinitesimal generator $\boldsymbol{Q}$ being composed of block partitioned submatrices with its off-diagonal blocks 0. This structural form of $\boldsymbol{Q}$ and the stability condition $\lambda_1 < \mu_1$, which makes the process positive recurrent, enables us to use the matrix geometric method [7] as a solution technique. To do this, we first obtain the minimal non-negative solution R to the matrix quadratic equation

$$R^2 C_{l+1} + R A_l + B = 0 \tag{3}$$

which has all its eigenvalues within radius 1. Then, the steady state probability vectors $\boldsymbol{p}_k$, for $k \geq l$, each of which corresponds to the steady state probabilities of a row in the lower portion of the state transition diagram is computed by

$$\boldsymbol{p}_k = \boldsymbol{p}_{l-1} R^{k-l+1}, \quad k \geq l$$

And the steady state probabilities corresponding to each row in the upper portion of the state transition diagram is obtained from the finite set of linear equations (the 2nd step in the following procedure) combined with appropriate normalization equation. The whole procedure to obtain the steady state probilities is summerized as follows :

1. Obtain R from $R^2 C_{l+1} + R A_l + B = 0$.
2. Obtain $\boldsymbol{p}_k$ for $k \geq l$, $\boldsymbol{p}_k = \boldsymbol{p}_{l-1} R^{k-l+1}$.
3. Obtain $\tilde{\pi} = (\boldsymbol{p}_0, \boldsymbol{p}_1, \cdots, \boldsymbol{p}_{l-1})$ from the following finite set of linear equations along with normalization equation. In fact, these steady state probability vectors are iteratively obtained computing backward starting from $\boldsymbol{p}_l$ that is obtained from step 2.

$$\tilde{\pi} \tilde{\boldsymbol{Q}} = 0$$
$$\{\tilde{\pi} + \boldsymbol{p}_{l-1} R (I-R)^{-1}\} \boldsymbol{e} = 1$$

where

$$\tilde{\boldsymbol{Q}} = \begin{bmatrix} A_0 & B & & & \\ C_1 & A_1 & 0 & & \\ & C_2 & & & \\ & & \ddots & B & \\ & 0 & & A_{l-1} & B \\ & & & C_2 & A_{l-1} + R C_l \end{bmatrix}$$

and I is the indentity matrix dimension n.

Once the steady state probabilities are determined, we can now derive the various performance measures of interest. Let $E[N_{nb}]$, $E[N'_{nb}]$, P_h, $E[B_{wb}]$ be the mean number of NB calls in the common buffer, the mean number of NB calls in the system, the probability that the number of NB exceeds the threshold value h, and the mean bandwidth allocated to each WB call, respectively. These are expressed as :

$$E[N_{nb}] = \sum_{i,j} n_{ij} p_{ij}$$

where n_{ij} is the number of NB calls in the queue when the system in the state (i, j), i.e.,

$$n_{ij} = \begin{cases} 0 & , \text{if} \quad b_1 i + b_2 j \leq c \\ i - \min(i, d) & , \text{otherwise} \end{cases}$$

$$E[N'_{nb}] = \sum_{i,j} i p_{ij}$$

$$P_h = \sum_{\substack{i,j \\ b_1 i + b_2 j > c \\ i - \min(i,d) \geq h}} p_{ij}$$

$$B_{wb}=\sum_{i,\,j} m_{ij}\,p_{ij}$$

where m_{ij} is number of channels of a WB call when the system in the state $(i,\ j)$, i.e.,

$$m_{ij}=\begin{cases} j & \text{, if } b_1 i+b_2 j\le c \\ j-d & \text{, otherwise} \end{cases}$$

We give some comments on the solution procedure of matrix geometric method mentioned above, in terms of computational complexity. It is shown that matrix R can be obtained from the following iteration whose convergence has been proved in [5]

$$R(0)=0$$

$$R(n+1)=-BA_1^{-1}-R^2(n)C_{l+1}A_l^{-1} \quad \text{and}$$

$$R=\lim_{n\to\infty} R(n)$$

This iteration and the linear set of equations involve a large amount of matrix operations, and requires lots of computation time. Furhermeore, due to the round-off and the precision of the number, the accuracy of computation becomes critical, particularly as the dimension of the matrix increases. The accuracy of computation can be improved somewhat by proper rearragement of equation (3), so that the matrix inversion in the above is applied to the diagonal matrix with its diagonal elements respectively being equal to the diagonal elements of the matrix A_l itself. Some other computational methods are studied by researchers in order to save the computation time and increase the computation accuracy.

Some numerical computations are carried out for various performance measures derived in the above, and their results are shown in Figure 4-9. For all computations, it is assumed that $\lambda_2=1/180$, $\mu_1=1/20$, $\mu_1=1/90$, $b_1=1$, $b_2=8$, $c=24$, and $n=7$. Then, we examine the performance as a function of the threshold value h for a different set of the arrrival rate of NB calls, $\lambda_1=0.01, 0.03, 0.05$, and the minmum bandwidth requirement of WB call, $m=1,2,3$.

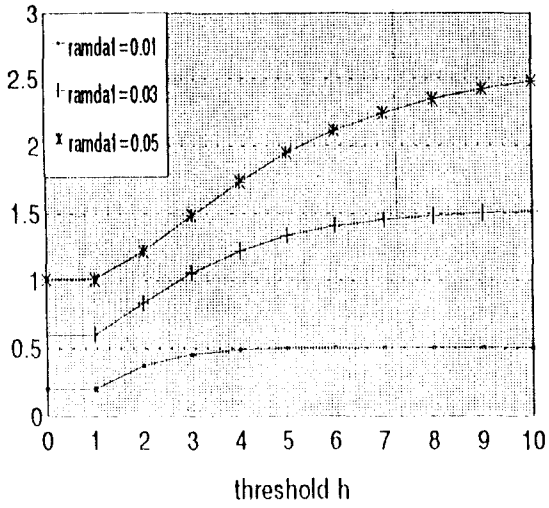


Fig 4. Mean number of NB calls in the system ($n=7$, $c=24$, $b_1=1$, $b_2=8$, $m=3$, $\lambda_2=1/180$, $\mu_1=1/20$, $\mu_2=1/90$)

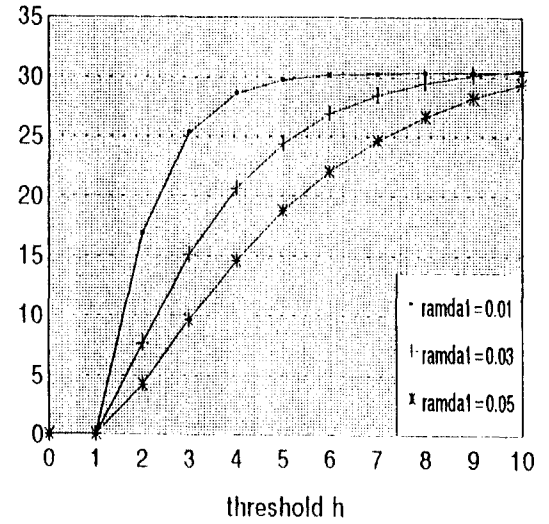


Fig 5. Mean queueing delay of a NB call for $\lambda_1=0.01, 0.03, 0.05$ ($n=7$, $c=24$, $b_1=1$, $b_2=8$, $m=3$, $\lambda_2=1/180$, $\mu_1=1/20$, $\mu_2=1/90$)

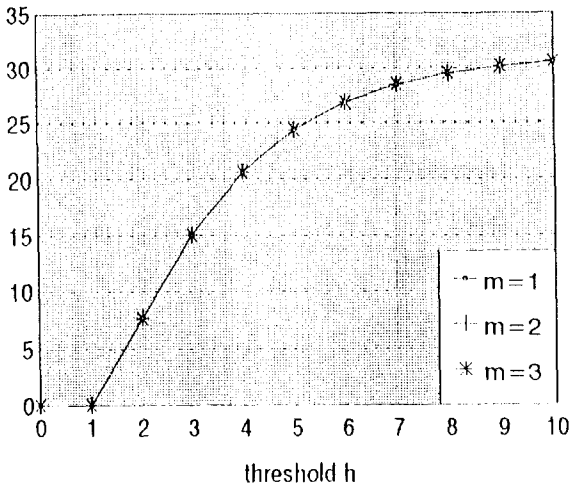


Fig 6. Mean queueing delay of a NB call for $m=1,3,5$ ($n=7$, $c=24$, $b_1=1$, $b_2=8$, $\lambda_1=3/100$, $\lambda_2=1/180$, $\mu_1=1/20$, $\mu_2=1/90$)

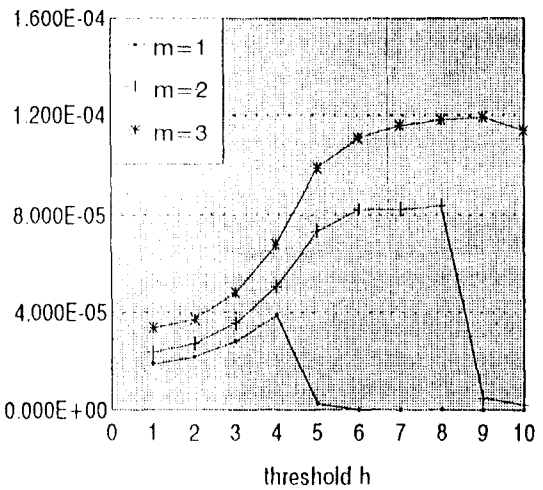


Fig 8. Probability that the number of NB queueing calls excced h for $m=1,3,4$ ($n=7$, $c=24$, $b_1=1$, $b_2=8$, $\lambda_1=3/100$, $\lambda_2=1/180$, $\mu_1=1/20$, $\mu_2=1/90$)

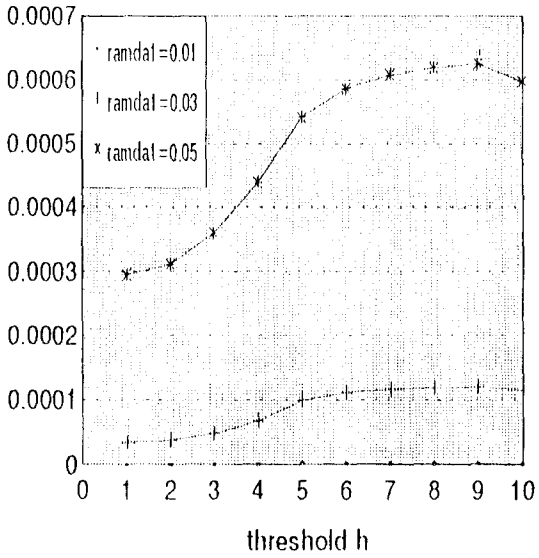


Fig 7. Probability that the number of NB queueing calls excced h for $\lambda_1=0.01,0.03,0.05$ ($n=7$, $c=24$, $b_1=1$, $b_2=8$, $m=3$, $\lambda_2=1/180$, $\mu_1=1/20$, $\mu_2=1/90$)

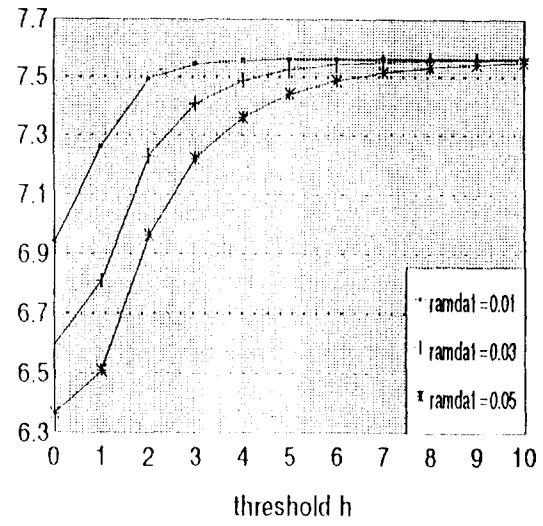


Fig 9. Mean bandwidth for a WB call ($n=7$, $c=24$, $b_1=1$, $b_2=8$, $m=3$, $\lambda_2=1/180$, $\mu_1=1/20$, $\mu_2=1/90$)

Figure 4 gives the mean number of NB calls, $E[N_{nb}]$ in the system as a function of threshold h for various arrrival rate of NB calls λ_1 increases monotonically, and rapidly as λ_1 increases. Note that monotonicity works since as h increases the chance to compress WB calls to its minimum bandwidth m decreases, so that the bandwidths extracted from WB calls and offered to NB traffics are small, thus the number of NB calls in the system increases as a function of h. Figures 5 and 6 shows the results of the mean queueing delay of a NB call obtained by changing λ_1 and m, respectively. The results in Figure 5 are obviously as expected due to the similar arguments mentioned above. Figure 6 illustrates that the effect of the minimum bandwidth requirement of a WB call, m, on the mean queueing delay of a NB call can be negligible for the selected set of parameters in this experiment. In Figure 7-8, we present the performance of the probability of the number of NB calls in the buffer exceeding the threshold h for various values of λ_1 and m. As shown in Figure 7, for a small value of λ_1 such as $\lambda_1=0.01$, this probability becomes zero. Also, from these figures, we see that the monotonicity of this probability as a function of h is not satisfied. But, the value of this probability is very small, namely less than 10^{-4}. Figure 9 indicates the results of the mean bandwidth allocated to each WB call. The curve is monotonically increasing This owes to the fact that the smaller the threshold is the more the chance of congestion among queueing NB calls increases, so that bandwidth of each WB call has more likely to be reduced. From this figure, we also see that the mean bandwidth for a WB call is less than its bandwidth requirement $b_2=8$, so that the difference of these two bandwidths is used to allow more WB calls and to allocate the NB calls in the waiting buffer.

Ⅳ. Conclusion

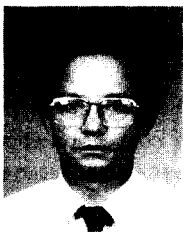
We have proposed a multiplexing scheme for integrating the real time and non-real time traffics with congestion control embedded that is primarily attained by bandwidth reduction strategy operated on real time traffics, assuming that the real time traffics require larger bandwidth than the non-real time traffics. The real time traffics are assumed to have a preemptive priority over the non-real time traffics. In order to resolve the congestion, as the congestion among the non-real time traffics increases, the bandwidths of each real traffic are reduced by a proper amount so that the bandwidths extracted from real time traffics are to be allocated to non-real time traffics. But, the bandwidths of the real time traffic are not allowed to be reduced below a given threshold, in order to guarantee a certain level of quality of the real time traffics when played out at the receiving end. In consequence, the scheme provides us an efficient way for utilizing the costly bandwidth resources, by accommodating as many time traffics as possible with gauranteeing its mimimun bandwidth requirements, and also resloving the congestion encountered among non-real time traffics. The system is described by the continuous-time Markov process, and analyzed by exploiting the matrix geometric method from which various performance measures of interest are derived. Computational issues, which becomes critical as the dimension of the matrix increases, are also discussed. In order to validate the analysis, some numerical computations are provided.

References

1. J. S. Baras, A. J. Dorsey, and A. M. Makowski, "Two Competing Queues with Linear Costs and Geometric Servics Requirement : The μc-rule is Optimal," *Adv. Appl. Prob.*, vol. 17, pp. 186-209, 1985.
2. K. C. Chua, and D. T. Nguyen, "Bit-rate Comression and Restricted Access Strategy for Integrated Services Digital Networks," *Computer Communications*, vol 13, no. 2, pp. 67-72, Mar. 1990.
3. T-C. Hou, and D. M. Lucantoni, "Performance

Analysis of an Integrated Video/Data Transport Mechanism with Built-In Congestion Control," *Proc. of Globecom '88*, pp. 231-238, 1988.

4. B. Kraimeche, and M. Schwartz, "Analysis of Traffic Access Control Strategies in Integrated Service Networks," *IEEE Trans. on Commun.*, vol. 33, no. 10, pp. 1085-1093, Oct. 1985.
5. G. Latouche, "Algorithmic Analysis of a Multiprogramming multiprocessor computer system," *Jour. of the Assoc. for Comp. Mach.*, vol. 28, no. 4, pp. 662-679, Oct. 1981.
6. A. A. Lazar, "Optimal Flow Control of a Class of Queueing Networks in Equilibrium," *IEEE Trans. on Automatic Control*, vol. 28, no. 8, pp. 1001-1007, Aug. 1983.
7. M. F. Neuts, "Markov Chains with Applications in Queueing Theory, which have a matrix geometric invariant probability vector," *Adv. Prob.*, vol. 10, pp. 185-212, 1978.
8. M. Schwartz, "Telecommunication Networks : Protocols, Modelling, and Analysis," Addison-Wesley Publishing Co. 1987.
9. K. Sriram, and D. M. Lucantoni, "Traffic Smoothing Effects of Bit Dropping in a Packet Voice Multiplexer," *IEEE Trans. on Commun.*, vol. 37, no. 7, pp. 703-712, July 1989.



宋 官 浩(Kwan Ho, Song) 正會員
1952년 1월 26일생
1973년 3월~1980년 2월 : 서울대학교 전자공학과(공학사)
1980년 3월~1984년 9월 : 한양대학교 산업대학원 전자공학과(공학석사)
1990년 3월~현재 : 광운대학교 대학원 전자통신 공학과 (박사과정수료)
1979년 11월~1985년 10월 : 금성전선(주) 정보시스템과장
1985년 11월~1987년 11월 : 데이콤(주) 미래연구실장
1987년 12월~현재 : 한국전산원 책임연구원
※ 주관심분야 : OSI 프로토콜, 인터넷프로토콜, 초고속통신망, 컴퓨터통신

李 在 昊(Jae Ho Lee) 正會員
1934년 5월 26일생
1968년 2월 : 광운대학교 통신공학과(공학사)
1978년 2월 : 단국대학교 대학원 전자공학과(공학석사)
1988년 8월 : 경희대학교 대학원 전자공학과(공학박사)
1970년~현재 : 광운대학교 전자통신공학과 교수
1985년~현재 : 광운대학교 통신과학 연구소 소장
1980년~1992년 : 한국 통신학회 이사 역임
1990년~현재 : 한국전산원 전산통신표준화 연구위원회 위원
1980년~현재 : 한국전기통신공사 협회 하도급 분쟁조정위원회
※ 주관심분야 : 데이터 통신, 통신망 제어, 디지털 교환기