Performance Evaluation of Buffer Management Schemes for Implementing ATM Cell Reassembly Mechanism

Gwangman Park*, Sungyeol Kang**, Changhoon Lie***

Abstract

An ATM switching system may be designed so that communications between processors of its control part can be performed via its switching network rather than a separate inter-processor communications (IPC) network. In such a system, there should be interfaces to convert IPC traffic from message format to cell format and vice versa, that is, mechanisms to perform the SAR (Segmentation And Reassembly) sublayer functions. In this paper, we concern the cell reassembly mechanism among them, mainly focused on buffer management schemes.

We consider a few alternatives to implement cell reassembly function block, namely, separated buffering, reserved buffering and shared buffering in this paper. In case of separated and reserved buffering, we employ a continuous time Markov chain for the performance evaluation of cell reassembly mechanism, judicially defining the states of the mechanism. Performance measures such as message loss probability, mean number of messages queued in buffer and average reassembly delay are obtained in closed forms. In case of shared buffering, we compare the alternatives for implementing cell reassembly function block using simulation because it is almost impossible to analyze the mechanism of shared buffering by analytical modeling. Some illustrations are given for the performance analysis of the alternatives to implement cell reassembly function block.

* Electronics and Telecommunications Research Institute, Economic Analysis Section
** Electronics and Telecommunications Research Institute, ATM Services Section
*** Seoul National University, Department of Industrial Engineering

1. Introduction

In the past few years, broadband ISDN has received increasing attention as a communication architecture capable of supporting multimedia applications. The key to a successful B-ISDN is the ability to support a wide variety of traffic with diverse service requirements. B-ISDN should also be able to cope with expected (as well as unexpected) future services in a practical and easily expandable fashion.

Several techniques have been proposed as the switching and multiplexing schemes for B-ISDN. Among them, asynchronous transfer mode (ATM) is expected to be the most promising transfer technology because of its efficiency and flexibility. ITU-T is also actively working on standardizing ATM-based B-ISDN.

In ATM, information from traffic sources are segmented into fixed-size packets called cells. They are statistically multiplexed and then transmitted to their destination. At each destination, the cells are reassembled to restore the original information. The cells for a connection are transmitted in a logical path, labeled by VCI (Virtual Channel Identifier) and VPI (Virtual Path Identifier) which is determined at its connection establishment. One of the most important network elements in an ATM based B-ISDN is the ATM switching system that exchanges various traffic from the sources to the destination.

Quite a few ATM switch architectures have been suggested and many experimental ATM switches are being developed. ATM switching systems are usually composed of many switching modules and each module may be controlled by its own processor. In such an ATM switching system, inter-processor communications (IPCs) are necessary in order to perform switching control functions properly. Furthermore, an ATM switching system can be designed so that its switching network can be used for inter-processor communications without a separate IPC network. In such a switching system, there should be interfaces to convert IPC traffic from message format to cell format and vice-versa, that is, mechanisms to perform the SAR (Segmentation And Reassembly) sublayer functions.

Many studies in packet communications have dealt with packetization buffers and their results may be applied to the design of mechanism which performs the segmentation of messages in ATM environment [1,2]. But, the literature seems to contain little analysis of reassembly buffers. Moors et al.[3] identify a number of critical issues that are particular to the implementation of receivers for ATM-based networks, and propose alternatives for dealing with these issues. Smith et al.[4] concern the virtual channel queue for the broadband terminal adaptor, to convert data packets into ATM cells and vice-versa. They derive an upper bound for the fraction of packets sacrificed by comparing

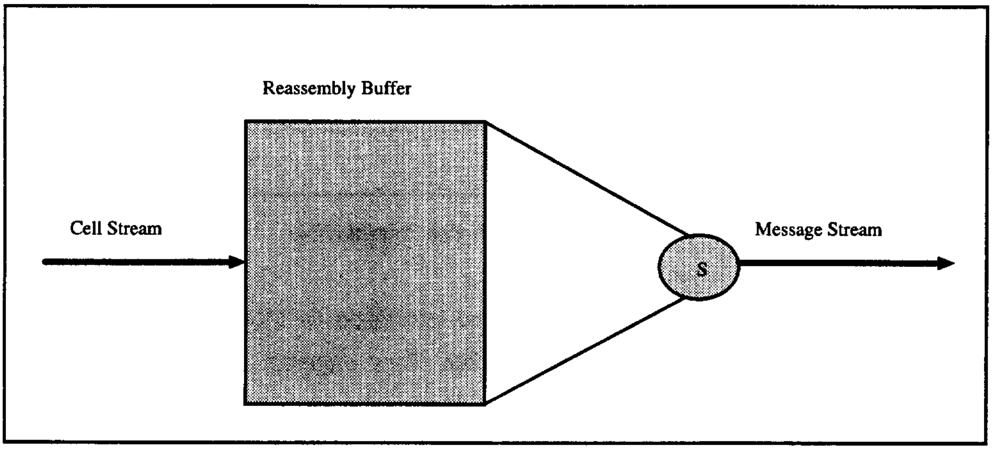

Fig. 1. Functioning of cell reassembly block

the sample paths of the completed work processes for finite and infinite buffers. Heijenk et al.[5] model and analyze the SAR processes. They obtain the first two moments of the interdeparture time distribution, given the first two moments of the interarrival time distribution by extending the analysis of the class of queueing networks considered in QNA(Queueing Network Analyzer), a software package developed at AT&T Bell Labs by Whitt et al. But, these papers mentioned in the above[3, 4, 5] have analyzed reassembly buffers whose buffer management schemes are different from that of this paper.

In this paper, we consider buffering schemes in implementing the cell reassembly mechanism among the SAR sublayer functions and develop their performance evaluation model. Fig. 1 depicts the functioning of the mechanism, which recovers messages by reassembling their multiple constituent cells.

We begin in Section 2 with the description of a few methods to implement cell reassembly mechanism, mainly focused on buffer management. Comparisons of these alternatives are given in Section 3 through analytical modeling and simulation approaches.

2. Methods for implementing cell reassembly function mechanism

If a message transferred using ATM fits within the cell payload, it can be transmitted as a single segment message. Otherwise, the message must be segmented by the source and trasferred in

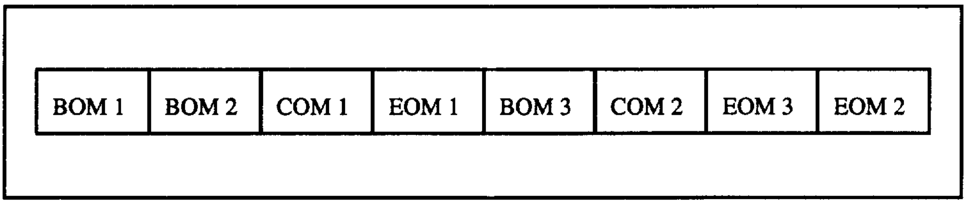

Fig. 2. Possible cell Interleaving

multiple cells. Cells for the message can be classified into Begin Of Message (BOM), Continuation Of Message (COM), and End Of Message (EOM) cells. They represent the first cell, the continuation cell, and the last cell of the message, respectively. These cells are multiplexed with those from other messages, possibly from other sources, requiring transport. In ATM, the occurrence of cells containing message from a specific source is not necessarily periodic. Hence, cells from different messages may be arbitrarily interleaved as shown in Fig. 2, where integer number represents message identifier.

Cell reassembly function mechanism performs the reassembly of a message from its constituent cells. The cells of a message arrive at the function block and wait to be ready for reassembly in buffer until the EOM cell of the message arrives. Once the EOM cell arrives, its constituent cells are assembled in order into original message format. After completing the reassembly of the message, the buffer space assigned for its constituent cells can be reassigned for other messages. According to buffer management scheme, there may exist various alternatives in implementing cell reassembly function block. We here consider three buffering schemes for constructing it, namely, separated buffering, reserved buffering and shared buffering.

2.1. Separated buffering

This scheme assigns dedicated buffer space for each message. Buffer size for a message is designed by considering the maximum length of messages and fixed in this mechanism. When BOM cell of a message arrives at cell reassembly block, it occupies dedicated buffer space reserved individually for each message irrespective of actual message size. So, buffer utilization is very low if the maximum size of messages is large but the great part of messages is small in comparison with the maximum. It is probable that the entire buffer is occupied with many small-size messages, resulting in large unused remaining space in buffer. But, this buffering mechanism is very simple in implementation and buffer management.

2.2. Reserved buffering

This scheme assigns buffer space according to message size when the BOM cell of this message arrives. It is different from separated buffering in that it assigns buffer space necessary for each message but separated buffering always assigns fixed buffer space for every message. Thus, it makes better the buffer utilization than separated buffering does. But, it is conservative in that the decision of whether to accept a message is made at arrival of BOM cell of the message, and cannot account for resources becoming available later. For example, if four cells of buffer space are available as BOM1 arrives in Fig. 2, then all messages shown could be reassembled (if reassembly buffer space was released immediately after each reassembly completed) but this mechanism would prevent the entry of BOM2 resulting in the blocking of message 2.

2.3. Shared buffering

To improve the drawbacks of reserved buffering, common buffer is shared by all messages in shared buffering. In comparison with the other buffering schemes, it clearly improves buffer utilization. But, reassembly deadlock may occur in this mechanism, that is, the common buffer may be saturated with only BOM cells and COM cells without any space for EOM cells of in-progress messages and BOM cells of a new message. Such phenomena do not exist in the previously mentioned mechanisms since they check empty buffer space for the message at the arrival instants of BOM cells. For breakthrough from the deadlock state, it is inevitable to discard cells of an unready message.

The possibility of occurrence of deadlock state may be very small since buffer space should be designed to guarantee prescribed performance level, such as message loss probability. However, if deadlock state occurs, it is desirable to recover from the deadlock quickly and to reduce the possibility of getting into the deadlock state again.

3. Performance analysis of alternatives for implementing cell reassembly function block

In this section, we compare the alternatives for implementing cell reassembly function block. Of course, several factors such as control complexity and required hardware volume in implementing them must be considered together. But, our interests here in comparing the alternatives are mainly placed on their buffer requirements to guarantee some prescribed message loss probabilities.

3.1. Performance analysis of alternatives by analytical approach

We develop queueing models to analyze performance of separated buffering and reserved buffering schemes. The assumptions of these models are the following.

- Arrival process of BOM cells is a Poisson process with rate λ.
- The size of messages (in cells) ranges from 1 to L and a_i denotes the probability that the size of message is i.
- Interarrival time distribution between BOM and EOM cell of a message is exponential with rate β.
- Reassembly time distribution of a message is exponential with rate μ.

3.1.1. Analysis of separated buffering scheme

In this section, we develop performance evaluation model of separated buffering scheme employing a continuous time Markov chain (CTMC). The buffer capacity (in messages) is assumed K_m. We define the states of this CTMC model by (x, y) where x is the number of messages in buffer whose BOM cells have arrived and y is the number of messages in buffer whose EOM cells have arrived $0 \leq y \leq x \leq K_m$.

The state transition rate diagram of this CTMC is shown in Fig. 3. From the state transition rate diagram, we can obtain the balance equations of this CTMC. Manipulating balance equations, we can

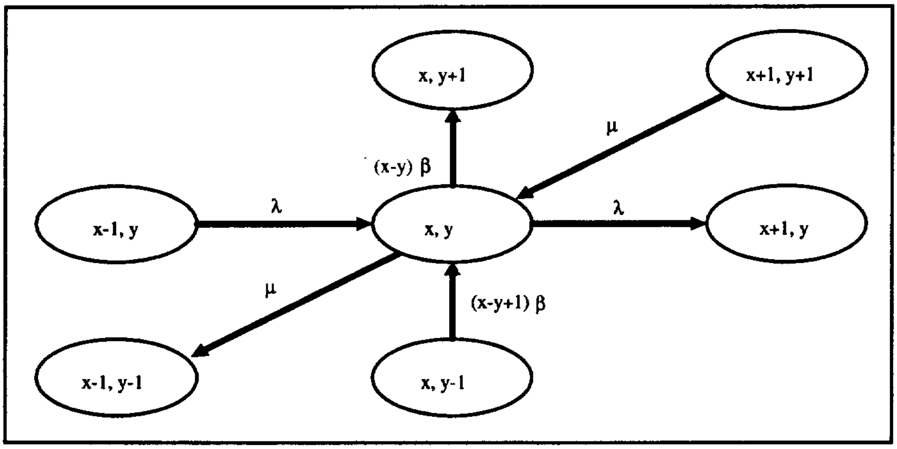

Fig. 3. State transition rate diagram(separated buffering)

get a nice result on the steady-state probabilities as follows.

$$P_{(0,0)} = \left[\sum_{x=0}^{K_m}\sum_{y=0}^{x}\lambda^x / ((x-y)!\beta^{x-y}\mu^y)\right]^{-1} \tag{1}$$

$$P_{(x,y)} = \lambda^x \cdot P_{(0,0)} / ((x-y)!\cdot\beta^{x-y}\cdot\mu^y) \tag{2}$$

Using PASTA (Poisson Arrival See Time Average) property, Little's law and effective arrival theorem, we can get message loss probability P_{loss}, mean number of messages queued in buffer L, and average delay of a message in buffer W (that is, the sojourn time from its arrival through reassembly completion).

$$P_{loss} = \sum_{y=0}^{K_m} P_{(K_m,y)} \tag{3}$$

$$L = \frac{\sum_{x=0}^{K_m} x\cdot(\sum_{y=0}^{x}\lambda^x / ((x-y)!\cdot\beta^{x-y}\cdot\mu^y))}{\sum_{x=0}^{K_m}\sum_{y=0}^{x}\lambda^x / ((x-y)!\cdot\beta^{x-y}\cdot\mu^y)} \tag{4}$$

$$W = L/(\lambda\cdot(1-P_{loss})) \tag{5}$$

3.1.2. Analysis of reserved buffering scheme

In the similar way as in section 3.1.1., we develop a performance evaluation model of reserved buffering scheme employing CTMC. The buffer capacity (in cells) is assumed K_c. We define the states of this CTMC by $(x_1,\ldots,x_L, y_1,\ldots y_L)$ where x_i is the number of messages having size of i cells whose BOM cells have arrived and y_i is the number of messages in buffer having size of i cells and whose EOM cells have arrived, $x_i \geq y_i$ for all i. Thus, the following relationships between x_i and K_c exist.

$$\sum_{i=1}^{L} i\cdot x_i \leq K_c \tag{6}$$

The state transition rate diagram of this CTMC is shown in Fig. 4. We illustrate transition out of the state $(x_1,\ldots,x_L, y_1,\ldots y_L)$ because of the graphical complexity. But, the whole transition rate diagram can be obtained easily. We assume that the order of service for ready message is random.

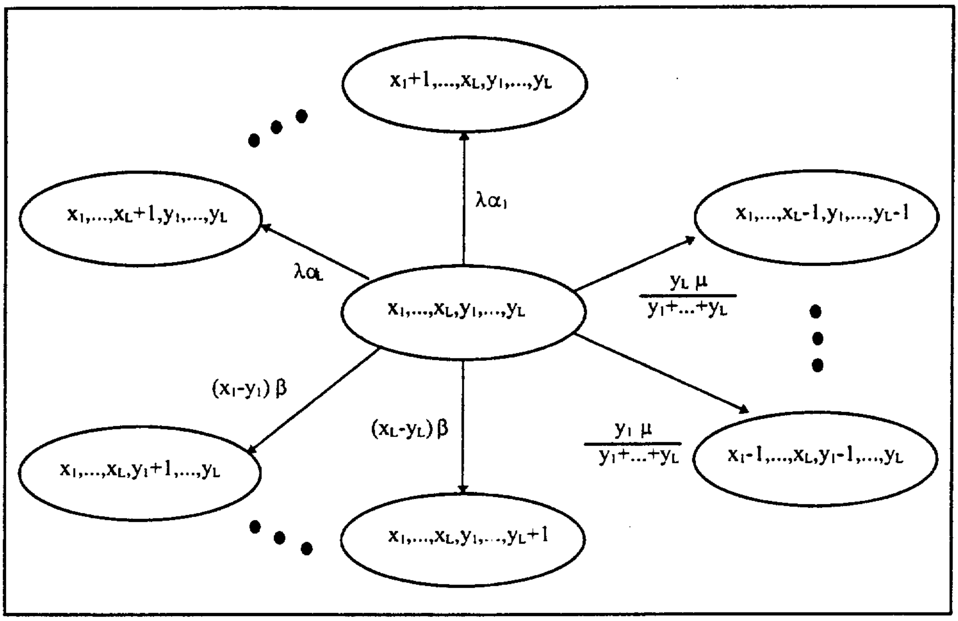

Fig. 4. State transition rate diagram(reserved buffering)

From the state transition rate diagram, we can get a nice result on the steady state probabilities. $P_{(x_1,\ldots x_L,y_1,\ldots,y_L)}$ is obtained by

$$P_{(0,\ldots,0,0,\ldots,0)} = (\sum_{(x_1,\ldots,x_L,y_1,\ldots,y_L)} (y_1+\ldots+y_L)!\prod_{i=1}^{L} (\lambda\alpha_i)^{x_i} / ((x_i - y_i)!\cdot\beta^{x_i-y_i}\cdot\mu^{y_i}\cdot x_i!))^{-1} \quad (7)$$

$$P_{(x_1,\ldots,x_L,y_1,\ldots,y_L)} = (y_1+\ldots+y_L)!(\prod_{i=1}^{L} (\lambda\alpha_i)^{x_i} / ((x_i - y_i)!\cdot\beta^{x_i-y_i}\cdot\mu^{y_i}\cdot y_i!))P_{(0,\ldots,0,0,\ldots,0)} \quad (8)$$

Once $P_{(x_1,\ldots x_L,y_1,\ldots,y_L)}$ is obtained, we can get message loss probability P_{loss}, mean number of messages queued in buffer L and average delay of a message in buffer W. Define P_{b_c} and P_x as

$$P_{b_c} = \sum_{1\cdot x_1+\ldots+L\cdot x_L=b_c}^{(x_1,\ldots,x_L,y_1,\ldots,y_L)} P_{(x_1,\ldots,x_L,y_1,\ldots,y_L)} \quad (9)$$

$$P_x = \sum_{x_1+\ldots+x_L=x}^{(x_1,\ldots,x_L,y_1,\ldots,y_L)} P_{(x_1,\ldots,x_L,y_1,\ldots,y_L)} \quad (10)$$

Then, P_{loss}, L and W is obtained by

$$P_{loss} = \sum_{b_c=K_c-L+1}^{K_c} P_{b_c} \sum_{i=K_c-b_c+1}^{L} \alpha_i \quad (11)$$

$$L = \Sigma_x x \cdot P_x \tag{12}$$

$$W = L / (\lambda \cdot (1 - P_{loss})) \tag{13}$$

3.2. Performance analysis of alternatives by simulation

In this section, we compare the alternatives for implementing cell reassembly function block using simulation. We assume the following for simulation modeling.

- Arrival process of BOM cell is Poisson process with rate λ.
- Interarrival time between two consecutive cells of a message is exponentially distributed with rate β.
- Reassembly time of message is proportional to its size (in cells) and that of one cell is constant μ.
- The size of messages (in cells) ranges from 1 to L.

The distribution of message size is expected to effect buffer utilization. In this paper, two different distributions, truncated geometric and discrete uniform, are considered. The probability density function of truncated geometric distribution is given by

Pr{message size is i cells} = $p \cdot (1-p)^{i-1}/((1-(1-p)^{L})$

where p represents the probability that message size is 1 cell.

Cell reassembly function block have to be designed to guarantee prescribed performance level such as message loss probability. Given traffic condition, buffer requirements vary with buffer management schemes. But it is almost impossible to determine buffer requirements precisely using simulation if performance requirements are stringent, e.g. message loss probability less than 10^{-8}. So, we here use the following procedure for the comparison of the alternatives described above for buffer management.

- Determine the simulation length and the number of replications.
- Execute simulation for the infinite buffer model and obtain a sample on the maximum requirements of buffer at each simulation run.
- For sufficiently large number of replications, we apply the central limit theorem to the distribution of maximum requirements of buffer, that is, assume that the distribution is normal,

and decide the buffer requirements as the 99 percentile of the distribution.

Fig. 5 and Fig. 6 show the buffer requirements obtained from the above procedure under specific traffic condition when the distribution of message size is truncated geometric and discrete uniform, respectively. Fig. 7 shows the buffer requirements of separated and reserved buffering schemes normalized by that of shared buffering scheme. We can see that buffer requirements vary with buffer management schemes as well as traffic load. And we can also see that the buffer requirement for the truncated geometric case is smaller than that for the uniform case although we can not find it directly in this figure because the buffer requirements of separated and reserved buffering schemes are normalized by that of the shared buffering scheme. It is because truncated geometric is left skewed, monotonically decreasing at positive mass points. Fig. 8 shows the buffer requirements as the maximum message size, L, changes. As expected, the buffer requirements increase as L does. Fig. 9 shows the buffer requirements as p, the parameter of truncated geometric distribution, changes. From this figure, we can see that the buffer requirements decrease as p increase. As mentioned in the above, it is because truncated geometric is the more left skewed as p increase.

4. Conclusion

An ATM switching system may be designed so that communications between processors of its control part can be performed via its switching network rather than a separate IPC network. In such a system, there should be interfaces to convert IPC traffic from message format to cell format and vice versa, that is, mechanisms to perform the SAR sublayer functions. In this paper, we have considered three buffer management schemes for implementing the cell reassembly mechanism among them, that is, separated, reserved and shared buffering. We have compared them for buffer requirement through a simulation approach. We observe that some significant reduction of buffer requirements can be achieved with either reserved buffering or shared buffering scheme rather than separated buffering scheme, especially when the great portion of messages are small in comparison with the maximum message size. We have also developed analytical models for the first two schemes which provide us with performance measures such as message loss probability and average delay in reassembly buffer in closed forms.

The cell reassembly function mechanism of SAR sublayer functions is also necessary for signaling messages and user traffic. So, the results in this study may be useful for the performance evaluation and dimensioning of those function blocks as well as that of IPC traffic.

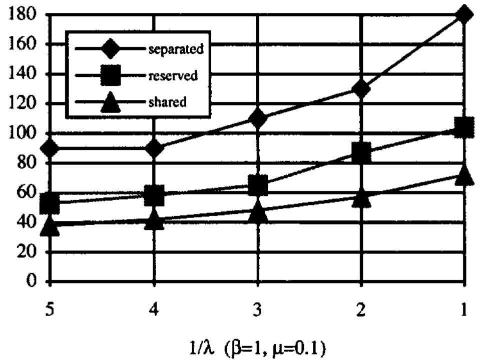

Fig. 5. Comparison of buffering schemes (truncated geometric distribution, L=10, p=0.2)

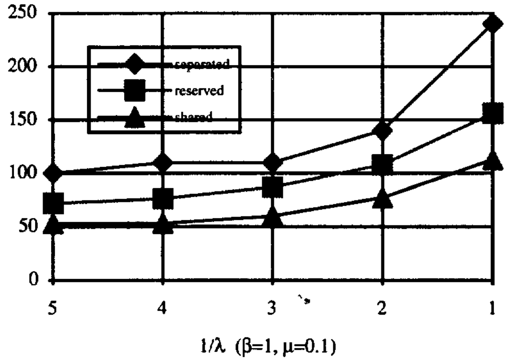

Fig. 6. Comparison of buffering schemes (discrete uniform distribution, L=10)

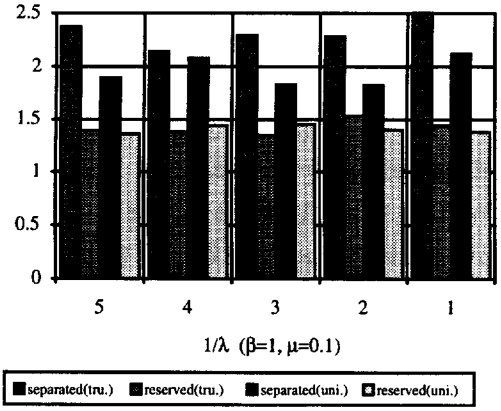

Fig. 7. Buffer requirements normalized by that of shared buffering scheme

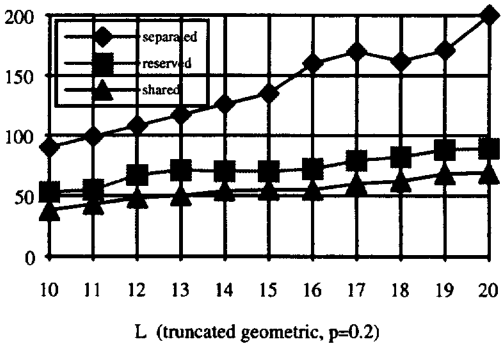

Fig. 8. Buffer requirements as L changes (λ =0.2, β =1, μ =0.1)

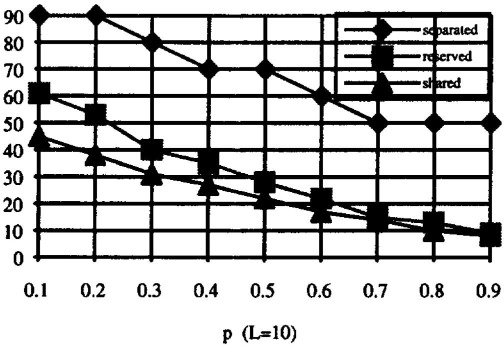

Fig. 9. Buffer requirements as p changes
(λ =0.2, β =1, μ =0.1)

References

[1] J.Chang and R-F Chang, "The behavior of a finite queue with batch Poisson inputs resulting from message packetization and a single server," IEEE Trans. Commun., Vol.COM-32, pp.1277-1284, Dec., 1984.

[2] D.R.Manfield and P.Tran-Gia, "Analysis of finite storage system with batch input arising out of message packetization," IEEE Trans. Commun., Vol.COM-30, pp.456-463, Mar., 1982.

[3] T.Moors and A.Cantoni, "ATM receiver implementation issues," IEEE J. Select. Areas Commun., Vol.11, No.2, pp.254-263, Feb., 1993.

[4] D.E. Smith and H.J.Chao, "Buffer sizing at a host in an ATM network," Proc. of IEEE INFOCOM '92, pp. 536-543, 1992

[5] G.J.Heijenk, M.E.Zarki and I.G.Niemegeers, "Modeling of segmentation and reassembly processes in communication networks," Proc. ITC 14, pp. 513-524, 1994.

[6] R.O.Onvural, Asynchronous Transfer Mode Networks : Performance Issues, Artech House Inc., 1993.

[7] G.Park, S.Kang and Y.Kim, "Buffer management schemes for implementing cell reassembly mechanism and their performance analysis," Proc. of JC-CNSS '94, pp. 69-73, July, 1994.

[8] G.Park, S.Kang and C.Han, "Performance evaluation of a cell reassembly mechanism with individual buffering in an ATM switching system," ETRI Journal, Vol. 17, No. 1, pp. 23-36, April, 1995.

[9] R.B. Cooper, Introduction to Queueing theory, second ed. New York : North Holland, 1981

[10] D.Gross and C.M.Harris, Fundamentals of Queueing Theory, second edition, New York : John Wiley & Sons, 1985.

[11] L. Kleinrock, Queueing Systems Vol.1: Theory, New York : John Wiley & Sons, 1975.