論文 83-20-3-10

On Statistical Multiplexing of Data Signals with Dynamic Buffer Control (Buffer의 동적제어에 의한 데이터 신호의 통계적 다중화에 관한 연구)

金 洛 明*殷 鍾 官**
(N. M. Kim and C. K. Un)

要 約

본 논문에서는 데이터 신호의 통계적 다중화에 관한 제문제들이 취급되었다. 먼저 한정된 waiting room을 가진 시스템에서 데이타 신호가 일정한 속도로 출력된다고 가정하고 batch-poisson 분포의 신호가 입력될 때의 queueing model을 검토하였다. 즉, 신호량과 신호의 평균길이를 변수로 하여 buffer의 크기, overflow의 확률 및 buffering으로 야기되는 queueing 지연시간 등을 살펴보았다. 그리고 데이터 신호의 통계적 다중화 시스템에 관한 실제 모델을 제시하여 마이크로프로그래밍을 통한 하드웨어 구현에 기본이 되게 하였다. 제안된 실제 모델은 buffer의 동적제어방식을 응용하여 설계되었으며 효율적인 I/O 프로세스가 되도록 마련하였다. 끝으로, 이 시스템 모델의 성능을 알아보기 위하여 여러 조건에서의 컴퓨터 시뮬레이션을 행하였다. 이 연구로 얻어진 제반 결과는 일반적인 buffer 설계에 지침이 될 수 있다.

Abstract

In this paper various aspects of statistical multiplexing of data signals have been investigated. A queueing model with finite waiting room and batch Poisson arrivals is studied assuming that data signals are transmitted at a constant rate. Using traffic intensity and average burst length as parameters, overflow probabilities and expected queueing delay due to buffering are obtained. Also, a real system model of a statistical multiplexer that can be directly used in micro-programmed hardware realization is proposed. To examine the performance of the system, computer simulation has been done at various conditions. The results obtained can be used in designing a buffer efficiently.

I. Introduction

In recent years, as the demand of data communications is increasing rapidly, the efficient use of communication channel bandwidth is important. One method to solve this problem is efficient multiplexing. In this work, we examine statistical multiplexing of digital data signals with packet transmission strategy. Our major consideration will be given to the buffer behavior that is the most important in the design of a statistical multiplexer. Two important problems are the interrelationship

*正會員, 金星電氣株式會社
(Gold Star Electric Co. Ltd.)
**正會員, 韓國科學技術院 電氣 및 電子工學科
(Dept. of Electrical Science, KAIST)
接受日字 : 1983年 1月 4日

between the overflow probability and the buffer size at various traffic intensities, and the queueing delay due to buffering. Overflow that may be defined as the average fraction of the data rejected results in lost data, causing invalid transmission. To reduce overflow occurrences, the use of a large buffer is desired. But, in this case, the queueing delay can become unacceptable.

From measurement of several operating computer systems, Fuchs and Jackson[1] have found that the burst arrivals can be approximated by the Poisson distribution, and that the burst length can be approximated by the geometrical distribution. Based on this result, studies on the buffer behavior have been done by several researchers[2],[3]. Although from these study results the buffer behavior is known fairly well, a buffer control model that can be directly used for hardware realization needs to be investigated.

In this paper we first analyze the buffer behavior of batch Poisson arrivals and a single constant output with a fixed queueing buffer. Next, we propose a first-in first-out (FIFO) dynamic buffer control method and a real system model of a statistical multiplexer that can be directly used in micro-programmed hardware realization. Also, we analyze the proposed model with the input traffic that is typical in data communication system, and verify the result by simulation. The results obtained can be used in designing a statistical multiplexer efficiently.

II. Description of Statistical Multiplexing System

It is known that the data structure of the user-to-computer traffic during a call has only 5% activity duration[4]. This indicates that during a call the system is in its idle state for a large amount of time. In such an environment, using the asynchronous time division multiplexing (ATDM) technique (statistical multiplexing is another name of ATDM.) instead of the synchronous time division multiplexing (STDM) technique gives a great advantage in channel utilization. The basic idea of statistical multiplexing is the introduction of a meassage switching technique in place of a circuit switching technique. In statistical multiplexing, the user whose message have arrived first gets the top priority in obtaining service. If the user (or terminal) has no message to transmit, it becomes disconnected from service. Hence, each user is granted access to the channel only when he has a message to transmit. Schematic diagrams of STDM and ATDM are shown in Fig. 1.

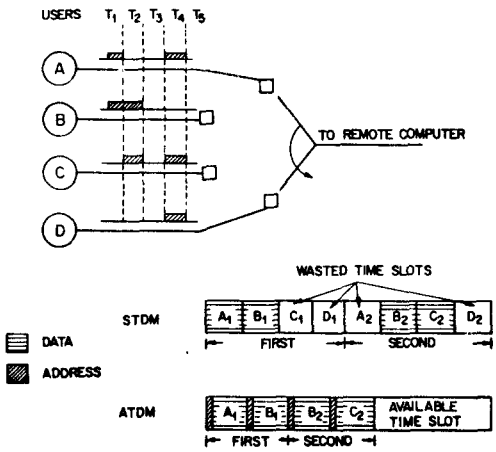

Fig. 1. Schematic diagram of STDM and statistical multiplexing.

As mentioned previously, the burst length can be approximated by a geometric distribution function and the burst arrival by a Poisson distribution function in data communication systems as follows:

$$f_X(\ell) = \theta \cdot (1-\theta)^{\ell-1}, \ \ell=1,2,..., \quad (1)$$

and

$$f_Y(n) = \exp(-\lambda) \cdot \lambda^n / n!, \ n=0,1,2,..., \quad (2)$$

where X represents the burst length and Y the number of bursts arriving during unit service interval.[1] Using (1) and (2), we can deduce the marginal probability mass function $\pi(j)$, which

[1] A unit service interval means the time required to transmit a character on output channel.

describes the probability of the arrival of j characters during a service interval as

$$\pi(j) = \begin{cases} \sum_{k=1}^{j} \binom{j-1}{k-1} \cdot (\lambda\theta)^k \cdot (1-\theta)^{j-k} \\ \qquad \cdot \exp(-\lambda)/k!,\ j=1,2,\ldots,\infty, \\ \exp(-\lambda),\ j=0. \end{cases} \tag{3}$$

Analysis of statistical multiplexing system requires the knowledge of buffer status that indicates the amount of data storage in the buffer. Since the buffer status is changed for every input or output process performed, the state equation have natually the form of a recursive equation.

Let us assume that the length of a queueing buffer is N, and that the probability that there are exactly n characters in the queueing buffer at the instant when one unit service has been finished is represented by P_n. Then, we can deduce the basic equation describing the buffer behavior. To obtain P_1, we consider the following three possible cases; the case that there is no character arrived with the previous state P_2, the case that there arrived one character with the previous state P_1, and lastly the case that there arrived one character with the previous state P_0. Thus, a state equation for P_1 can be written as

$$P_1 = \pi(0) \cdot P_2 + \pi(1) \cdot P_1 + \pi(1) \cdot P_0, \tag{4}$$

where $\pi(k)$ represents the number of characters arrived during a unit service interval. Generalizing this result, we can obtain the following N recursive equations:

$$P_n = \pi(0) \cdot P_{n+1} + \sum_{i=1}^{n} (n-i+1) \cdot P_i + \pi(n) \cdot P_0, \quad n=0,1,2,\ldots,N-1. \tag{5}$$

In addition, since the length of a queueing buffer is constrained to be N, we have the following normalizing equation:

$$\sum_{i=1}^{N} P_i = 1. \tag{6}$$

Using (3), (5) and (6), we can obtain the equilibrium state probabilities, P_0, P_1, ..., and P_{N-1}.

Now, let α be the average data departure rate and $\beta(\triangleq \lambda/\theta)$ be the average data arrival rate. Since the transmission of data initiates at any time when data storage at the queueing buffer is not zero, we can have

$$\alpha = 1 - P_0. \tag{7}$$

In general, overflow can be defined as the ratio of the average number of data rejected and the total number of arriving data. Hence, the overflow probability P_{of} may be written as

$$P_{of} = 1 - \frac{\alpha}{\beta}. \tag{8}$$

In addition, the traffic intensity, a measure of congestion of input stream, can be expressed in terms of the data arrival rate β and unit service interval μ. That is,

$$\rho \triangleq \frac{\beta}{\mu}. \tag{9}$$

III. Modeling of Statistical Multiplexing System

1. General

In a statistical multiplexing system with packet transmission strategy, an addressing scheme is needed to identify each transmitting message. Also, to handle messages arriving randomly, an efficient buffer control algorithm is required. Although these extra requirements result in a little overhead in channel utilization and some increase in system complexity, these effects are minor compared with the large gain in channel utilization through statistical multiplexing.

A statistical multiplexing system is composed largely of several line buffers, a queueing buffer, and a buffer control unit. Data signals generated by users arrive first at line buffers to be served by the queueing buffer. Without any priority problems, data arrived are served according to the first-in first-out (FIFO) strategy. That is, if any one of the users

gets first a certain amount of the contents of his line buffer, he is immediately granted the queueing buffer service. A queueing buffer is the central part of the multiplexing system under consideration. In managing the queueing buffer, it is most important to consider how to partition its capacity among all users. A block diagram of the statistical multiplexing system is shown in Fig. 2.

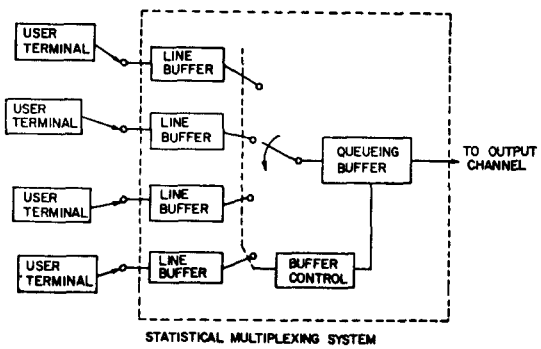

Fig. 2. Construction of a statistical multiplexing system.

2. Dynamic Buffer Control

As a buffer control algorithm, we use a variable size partitioning algorithm based on dynamic buffer control. This algorithm uses adaptive distribution of the space in a queueing buffer memory according to the amount of data to be transmitted to each destination. Thus, the maximum space that can be assigned to each destination depends on the buffer occupancy at the moment and on the number of active otputs. Let the space to be assigned to the i-th output be represented by S_i. Then, S_i can be written as

$$S_i = \frac{M}{m} + k_i, \qquad i = 1,2,\ldots,m,$$

where

$$\frac{M}{m} + 1 \leqq k_i \leqq M - \frac{M}{m} - m + 1. \tag{10}$$

In (10) M represents the size of a queueing buffer memory in blocks and m stands for the number of destinations. The buffer memory can be assigned to each user adaptively from minimum one block to maximum M-m+1 blocks.

3. Description of a Real System Model

Modeling of a statistical multiplexing system involves four subsystem modeling processes;

(i) modeling of a line buffer,
(ii) design of the buffer control unit,
(iii) modeling of a queueing buffer,
(iv) modeling of the input/output (I/O) process.

Among these subsystems, the buffer control unit is the most important, which controls all the signal flows among sub-blocks in the system with dynamic FIFO service.

A functional block diagram for the statistical multiplexing system is shown in Fig. 3. A line buffer accepts all the data from each user

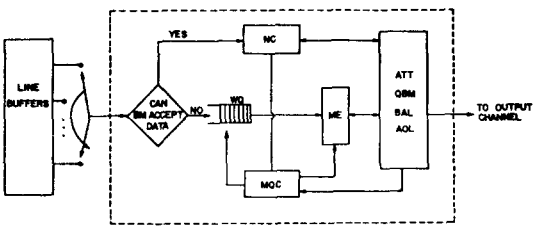

Fig. 3. Functional block diagram of statistical multiplexing system.

without any interrupt. The status of the line buffer is continuously checked and the information of the contents of the line buffer is fed back to the buffer control unit. The buffer control unit is composed of a network controller, a message queueing controller, and a message editor. The network controller performs functions such as transmission control, device control, and message routing. The message queueing controller performs queueing and scheduling for transmission. Lastly, the message editor does text coding, message header interpretation and linkage control. From the information of line buffer state, the controller decides whether the data in the line buffer should be transmitted to the queueing buffer or not.

The queueing buffer system consists of

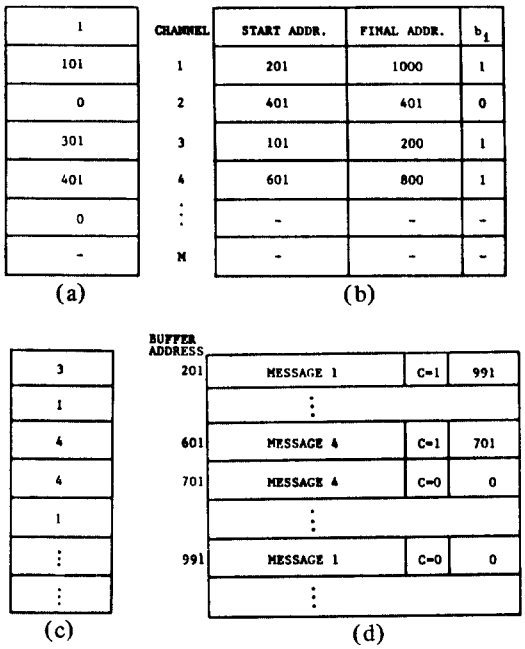

Fig. 4. Sub-block of buffer system.
(a) BAL (b) ATT (c) AOL (d) QBM

four sub-blocks; an address translation table (ATT), a queueing buffer memory (QBM), block available list (BAL), and arrival order list (AOL) (see Fig. 4). The ATT translates the message destination addresses to the physical addresses of the QBM. The QBM is a waiting place for the message before getting the output service for transmission. It is divided into many fixed-sized blocks. Here, the size of a block is defined as a multiple of a single packet size. A single block consists of three elements; the message, a block continuation bit, and a linkage pointer. The block continuation bit indicates whether the block is the last block of the message or not. If the block is not the last block, the linkage pointer following the block continuation bit indicates the physical address of the next block of the message. Control of storage and transmission of data is done by the dymaic buffer control algorithm for which the memory partition method was discussed previously. The BAL is a storage where all the states of the WBM are written. In the BAL, address of the unoccupied blocks in the QBM are always available and immediately updated as input and output processes are performed. Finally, for the FIFO service, the AOL is prepared in the system. In the AOL, destinations of arriving message blocks are written in the order of arrival at the time when the blocks are stored in the QBM. Then, during the output process, the FIFO service is done by deciding the order of service in the order listed in the AOL.

The input process is composed of two procedures; choosing a channel for service and transferring the data for the chosen channel from the line buffer to the QBM. The choice of a channel is made by the scanning operation. Data generated by each user is received by the system and first stored in a line buffer without any interruption. If any one of the line buffers has its contents exceeding a certain threshold, the channel connected to the line buffer is selected through the scanning operation.

On detecting a channel for service, the controller checks if there is any space to store in the QBM. The controller selects one block among available blocks listed in the BAL, and adopts it as the first block address. Now, the controller looks up the ATT. In this case, there can be two possible states. First, if the buffer status bit for the destination of the data is 0 (no queueing message for the same destination), the controller writes the selected address on the list of the first block address of the ATT. The data read from the line buffer is stored in the QBM starting at the first block address. After storing one block of data in the QBM, it is checked whether the block is the last block of data or not. If the block is the last block, the controller sets the block continuation bit to 1, and search for the BAL for another space to serve the remaining data. Also, the controller writes the destination of the stored block on the list of the AOL. searches for the last block of data in the QBM, and then sets the block continuation bit to 1. Next, the BAL is searched. In this case if there is no list of empty blocks, the data should wait in the waiting quenue. When a new address is selected, the controller writes

the selected address on the linkage pointer of the last block of data in the QBM. After Through the above procedure, the messages are served to reach the last block of each message. When the data just served is the last block of data, the block continuation bit is set to 0. And the last block address of the data is written on the list of the last block address of the ATT. Second, if the buffer status bit for the destination of the data is 1 (some messages are enqueued now for the same destination), then the data should be stored in the QBM right after the enqueued data. Therefore, one should first identify the last block address of the enqueued data from the ATT. After the identification, the controller the process, storage of data in the QBM follows. This storage is also done in the same manner as illustrated in the first case. Fig. 5 shows the overall flow chart of the queueing buffer service described above.

The output process performs the function of transmitting data in the QBM on an output channel at a fixed transmission rate. This process is "unlocked" at anytime when the QBM is not empty. Whether the QBM is empty or not can be easily checked by the buffer status bit in the ATT. The output process can be initiated if any one of buffer status bits is 1.

The output process is performed as follows. First, the controller examines the AOL to know which destination should be linked first. After the determination of the destination, the controller reads the first block address overflow probabilities in Fig. 7. This type of errors can be eliminated if a large size of data base were used. Still another reason for discrepancy is due to the assumption that the maximum message length is infinite, whereas in our simulation we put some limitation on the

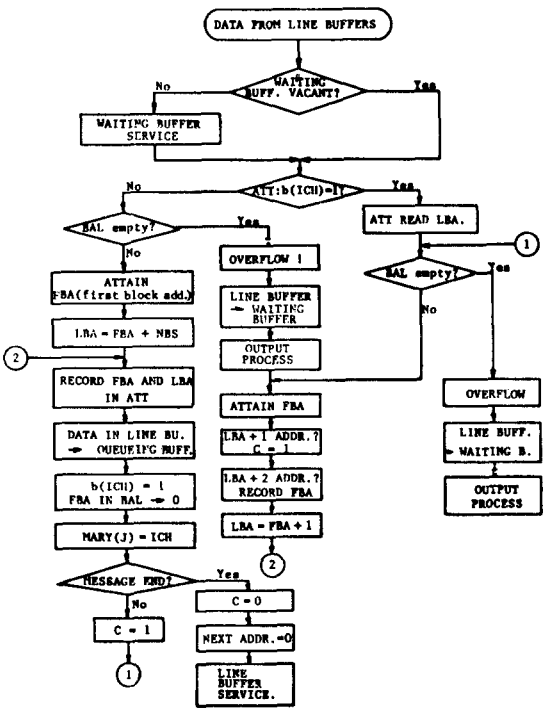

Fig. 5. Flow diagram of queueing buffer service.

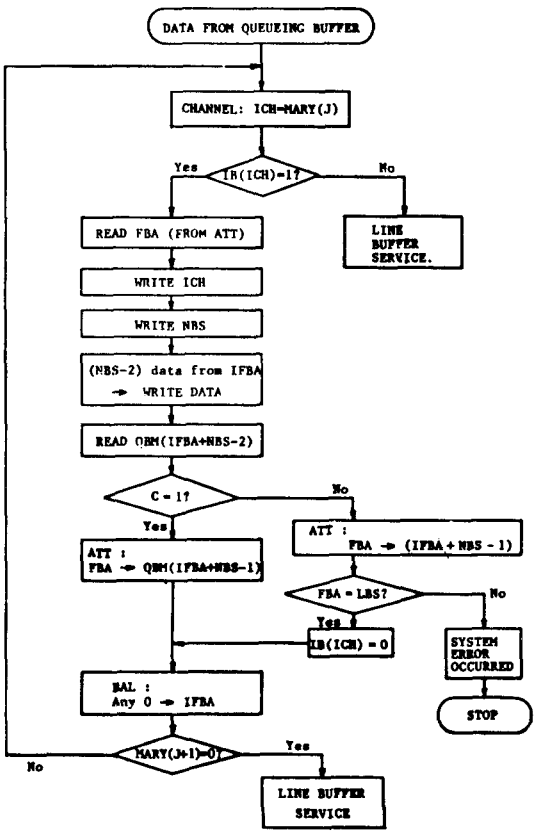

Fig. 6. Flow diagram of output process.

from the ATT. Then, transmission of the data in the block initiates. Now that the block can be used for storage of other data, the first block address of the transmitted data is written in the BAL. As the last step, the controller updates the first block address of the ATT by writing the contents of the linkage pointer on the list of the first block address of the ATT. A flow diagram describing the above procedure is shown in Fig. 6.

IV. Computer Simulation and Discussion

One of the most important aspects in the performance of a statistical multiplexing system is the time delay occurring during the passage of data through the system. Generally, the time delay results mostly from the waiting time in the queueing buffer. Therefore, the size of a queueing buffer directly affects the delay time.

Our computer simulation was aimed at the performance evaluation of the proposed real system model of a statistical multiplexer. Also, we studied on the performances of the real system model in comparison with those of the analytic model described in Chapter 2.

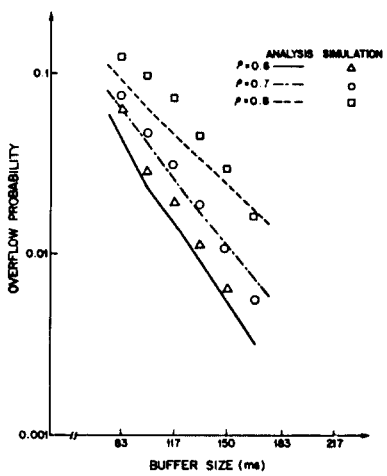

Fig. 7. Buffer overflow probability vs. buffer size with the traffic intensity ρ as a parameter.

Here, we examined carefully the relationship between the size of a queueing buffer and the overflow probability, and the queueing delay resulting from buffering. For the input data base, we employed Martin's model[5] that is known to model accurately the characteristics of real data traffic.

Fig. 7 shows the relationship between the size of the queueing buffer and the overflow probability. Overflow probability depends on the size of a queueing buffer, the traffic intensity, and the average length of messages. Thus, with a fixed buffer size, the overflow probability increases with the increase of traffic intensity. With a fixed buffer size, the overflow probability increases with the increase of traffic intensity. The small discrepancy between analysis and simulation has resulted largely from the assumption in analysis that the occurrence of overflow is negligible. In fact, overflows often occur, especially when the traffic intensity is high and/or the buffer size is small. Another cause of error is the limitation of computational capacity. This has affected dominantly the lower values of

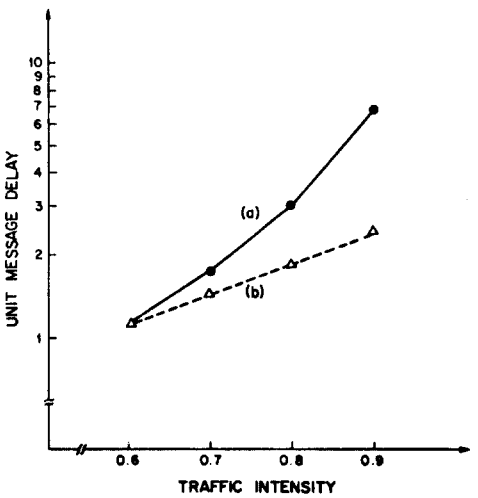

Fig. 8. Queueing delay vs. traffic intensity.
(a) Average burst delay
(b) Normalized average packet delay

Note : 1 packet $= \frac{1}{6}$ average message length.

message length.

Fig. 8 illustrates the relationship between the queueing delay due to buffering and the traffic intensity of input data. In the figure, the curve (a) represents the burst delay and the curve (b) represents the packet delay. As seen in the figure, the average burst delay and packet delay increase as the traffic intensity increases. And, if the traffic intensity is fixed, the delay increases as the average message length increases. In addition, it can be confirmed from the figure that packet switching generally induces less delay than message switching.

V. Conclusions

In this work a statistical multiplexing system of data signals has been examined and analyzed at various conditions. In addition, a real system model of the statistical multiplexer of data signals that can be directly used in hardware realization through microprogramming has been proposed. The results obtained can be used as a guide to efficient design of a statistical multiplexer.

References

[1] E. Fuchs and P.E. Jackson, "Estimates of distribution of random variables for certain computer communication traffic models," *Proc. ACM Symp. Probl. in the Optimization of Data Commun. Syst.*, (Pine Mountain, Ga., Oct. 13-16, 1969), pp. 201-227.

[2] N.M. Dor, "Guide to the length of buffer storage required for random input and constant output rates," *IEEE Trans. Elec. tron. Comput.*, vol. EC-16, pp. 638-684, Oct. 1967.

[3] W.W. Chu and L.C. Liang, "Buffer behavior for mixed input traffic and single constant vol. COM-20, pp. 230-235, Apr. 1972.

[4] P.E. Jackson and C.D. Stubbs, "A study of multi-access computer communication systems," *AFIPS Conf. Proc.*, 34 (1969(, pp. 491-504.

[5] Martin, *Systems Analysis for Data Transmission.* Englewood Cliffs, N.J.: Prentice Hall, Inc., pp. 515-540, pp. 641-676, 1972.