THE BELL SYSTEM TECHNICAL JOURNAL

DEVOTED TO THE SCIENTIFIC AND ENGINEERING
ASPECTS OF ELECTRICAL COMMUNICATION

Volume 58 February 1979 Number 2

A Queuing Model for a Hybrid Data Multiplexer

By R. R. ANDERSON, G. J. FOSCHINI, and B. GOPINATH

(Manuscript received March 3, 1978)

There are several instances in a data network where a communication line is shared by two or more types of data. In this paper, we analyze the performance of a buffer used to multiplex two types of data. Sporadic short messages, like inquiries from terminals, share the same channel as relatively steady synchronous data, like trunk traffic or long messages from computer data bases. To the authors' knowledge, previous studies have been limited to an ad hoc approximation to the probability distribution of interest. We solve for the equilibrium distribution of number of units of data in the buffer. The delay distribution easily follows. Numerical results are also presented which can be used as a guide to determining how much of each type of traffic can be sustained simultaneously.

I. INTRODUCTION

We analyze the performance of a buffer that is used to multiplex two types of data. Sporadic short bursts of data, like inquiries from terminals, share the same communication channel with relatively steady streams of data like digitized voice, lengthy messages from computers, data bases, or traffic from a busy trunk. A line-switched network, one that provides a dedicated channel for each connection, is preferred for lengthy steady messages. A packet-switched network, on the other hand, is efficient for messages that are short and bursty. In a packet-switched network, messages are forwarded from node to node in the form of packets of data that include addressing information. In such a network, there is no necessity for a dedicated channel for each connection.

A data network could accommodate both types of traffic by dividing the transmission facilities into two fixed parts—one part exclusively for line-switched traffic, the other for packet-switched traffic. The subframe switching concept introduced in Ref. 1 is an example of a temporal division of capacity. The model developed here can be used for the analysis of such a system. In any system where a resource such as a transmission line is shared between two or more types of users, the performance guaranteed to each individual type of customers has to be met. Packet delay and the probability of losing packets are the two measures of performance we consider.

Kummerle[2] proposed a model for multiplexing line and packet-switched data and derived, using an ad-hoc approximation, formulas relating the performance measures to the traffic intensity and transmission capacity. A similar problem was analyzed using a diffusion approximation in Ref. 3. In Ref. 4, an $M/D/N$ model is used for an approximate analysis. Reference 5 looks at a related continuous-time problem where the arrival mechanism (rather than the service) has a periodic component. An integral equation is derived that can be solved using Wiener-Hopf techniques. In this paper, we formulate a model for the multiplexer and solve it exactly. We then describe the computational method used to derive the numerical results and display them to illustrate the tradeoff involved in performance and line utilization.

II. DESCRIPTION OF THE MULTIPLEXING SYSTEM

The sources of data that are connected to a generic node in the network are divided into two groups—synchronous sources and asynchronous sources. Both these sources generate messages randomly. However, when a synchronous source generates a message, the message is generated at a constant rate and is much longer than the messages generated by asynchronous sources. To describe the system, we use an example. Let sources A, B in Fig. 1 be two synchronous sources transmitting at ½ and ⅙ the line (marked LINE in Fig. 1) rate, respectively. Packets are assumed to be of fixed size, and the unit of time is normalized to be the time required for the line to transmit one packet. The output of source A is assembled into packets by the line buffers as shown in Fig. 1. Then the output of this buffer, connected to source A, will produce one packet every two units of time. As soon as these packets are ready, they are transmitted by the line even if packets from asynchronous terminals, marked T in Fig. 1, are waiting to be transmitted in the packet buffer. Similarly, the output of the line buffer connected to source B produces a packet every six units of time. However, the output of B is so synchronized that A and B packets do not have to be transmitted at the same time on the line. The asyn-

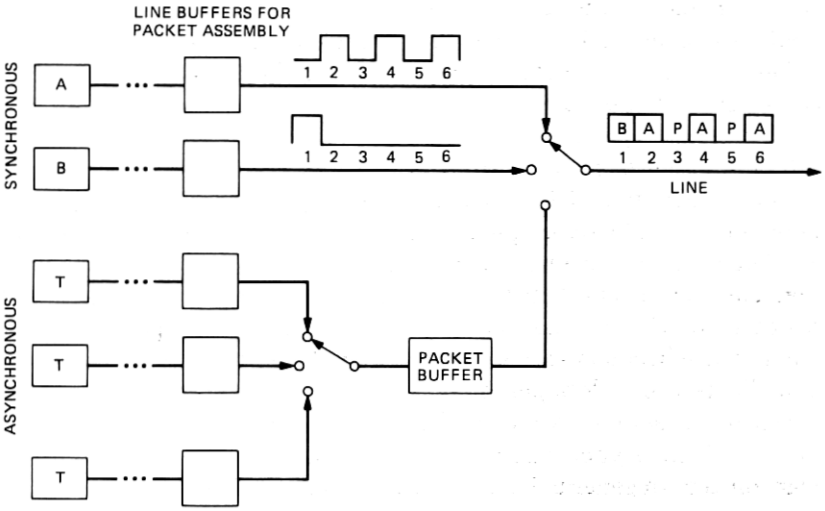


Fig. 1—Example of multiplexing system.

chronous sources have their own line buffers doing the packet assembly, but the output of these buffers are not necessarily synchronous with the line. As soon as these sources produce packets, they are inserted into the packet buffer and there await transmission on the output line. Notice that this synchronous method of transmitting sources A and B allows us to discard addressing information in all except the first packets of a message from these sources. In practice, if a packet from an asynchronous source arrives at the buffer when it is full, the packet is lost. In order to guarantee satisfactory performance for the asynchronous sources, we must keep the probability of such a loss small (requirements in the $10^{-5} - 10^{-7}$ range are typical). It is mathematically convenient to work with an infinite rather than a finite buffer model and solve for the probability that queue size exceeds a given threshold. The probability that queue size in such a buffer exceeds a level B is an upper bound for the probability that a finite buffer of size B overflows.

III. THE MATHEMATICAL MODEL

The model considered here applies to systems where there are one or more packet buffers (associated with asynchronous sources) and, of course, many synchronous sources. The packet buffers and synchronous sources may be served by the line (it is available to accept a packet for transmission) in any fixed periodic pattern. The distribution of the number of packets in any given packet buffer is only influenced by the asynchronous traffic connected to it and the pattern in which the line becomes available to it. During the time slots that the line is not available to the given packet buffer, it is immaterial whether another packet buffer or a synchronous source is being served.

Hence, the mathematical model described below analyzes a single-server queue in discrete time with the server being absent according to a fixed periodic pattern.

The number of packets in the packet buffer at the end of nth unit of time is denoted by b_{n+1}. The number of packets that the asynchronous sources collectively generate in the nth unit of time is denoted by x_n. The sequence of integers $\{x_n\}$ is assumed to be samples of independent, identically distributed, random variables. The frame length denoted by M is the period of the pattern of serving packet buffers and the synchronous sources. In the example of Fig. 1, the frame length is 6. There are 6 slots per frame and slots 1, 2, 4, 6 are dedicated to synchronous sources A and B. Slots 3 and 5 are used for transmitting packets from asynchronous sources whenever there are any to be transmitted. In general, let $\mathcal{S}$ denote the set of indices of slots in which the given packet buffer is not served. In the example of Fig. 1, $\mathcal{S} = \{1, 2, 4, 6\}$. Then b_{n+1}, the number of packets in the buffer at the end of the $(n + 1)^{\text{st}}$ unit of time, is given by

$$b_{n+1} = (b_n - u_n)^+ + x_n, \tag{1)*$$

where

$$u_n = \begin{cases} 0 \text{ if } n \equiv \mathcal{S}(M) \\ 1 \text{ otherwise.} \end{cases}$$

Whenever $n \equiv \mathcal{S}(M)$, no packets from the packet buffer can be transmitted in the nth unit of time. Therefore, for such an n,

$$b_{n+1} = b_n + x_n.$$

On the other hand, if $n \not\equiv \mathcal{S}(M)$, a packet from the packet buffer can be transmitted in the nth unit of time (if there were any left at the end of the $(n - 1)$st time interval). Hence,

$$b_{n+1} = (b_n - 1)^+ + x_n$$

if $n \not\equiv \mathcal{S}(\mathrm{M})$. Because of the time-varying nature of u_n, it is clear that $\{b_n\}$ itself has no stationary distribution. However, the vector process $\mathbf{b}_m = (b_{mM+1}, b_{mM+2}, \cdots b_{(m+1)M})^t$ indexed by m has a stationary distribution because of the periodic nature of u_n. We will show that we can find a relationship between the marginal distributions of the components of b_m that uniquely specify the equilibrium distribution of the b_{mM+i}, $i = 1, 2, \cdots M$.

The process $\{b_n\}$ is Markov process with the state space being the

* $x^+ = x$ if $x > 0$, $x^+ = 0$ if $x \leq 0$, $n \equiv \mathcal{S}(\mathrm{M})$ if $n \equiv i \bmod M$ for some $i \in \mathcal{S}$.

nonnegative integers. The transition probability matrix P_n at time n is determined by whether $n \equiv \mathscr{S}(\mathrm{M})$ or not; $P_n = P_m$ if $n \equiv m \bmod M$. Let P_i denote P_{mM+i}, $i = 1, 2, \cdots, M$. For each i, the process b_{mM+i}, indexed by m, is again a Markov process with the associated transition probability matrix

$$Q_i = P_{i-1}P_{i-2}\cdots P_1 P_M P_{M-1}\cdots P_{i+1}P_i.$$

It is easily shown that the Markov chain associated with Q_i is aperiodic and irreducible. We will now show that, if the average number of packets arriving in the packet buffer is less than the number of slots available for transmission during a frame, this Markov chain is positive recurrent and hence, for each i, b_{mM+i} has a limiting stationary distribution as $m \uparrow \infty$. Let J denote the number of slots in a frame during which no packets from the packet buffer can be transmitted.

Lemma 1: For each $i = 1, 2, \cdots, M$ the Markov chain associated with Q_i is positive recurrent if

$$MEx_n < M - J.$$

Proof: Repeated application of (1) shows that, for $m = 0, 1, 2, \cdots$,

$$b_{(m+1)M+i} = (\cdots((b_{mM+i} - \delta_1)^+ + x_{mM+i} - \delta_2)^+ \cdots + x_{(m+1)M+i-2} - \delta_M)^+ + x_{(m+1)M+i-1},$$

where $\delta_j = 1$ if $j \not\equiv \mathscr{S}(M)$ and $\delta_j = 0$ otherwise. Therefore, if $b_{mM+i} \geq M$, then

$$b_{(m+1)M+i} = b_{mM+i} + \left\{ \sum_{\ell=0}^{M+1} x_{mM+i+\ell} \right\} - (M - J);$$

hence,

$$E(b_{(m+1)M+i} \mid b_{mM+i} = j) = b_{mM+i} + MEx_n - (M - J)$$

for all $j \geq M$. Let $Q_i(\ell, j)$be the (ℓ, j)th element of Q_i; then, if $MEx_n < M - J$

$$\sum_{\ell=0}^{\infty} Q_i(\ell, j)\ell < j \qquad \text{for} \quad j \geq M.$$

Hence, using Theorem 2 of Ref. 6, the Markov chain corresponding to Q_i is positive recurrent.

IV. CALCULATION OF STEADY-STATE DISTRIBUTION

In this section, we show how to calculate the steady-state distribution of the components of $\mathbf{b}_m$. As we mentioned earlier, the distribution of b_n has no limiting value as n tends to ∞. However, we showed that the distribution of b_{mM+i} for $i = 1, 2, \cdots, M$ approaches a limiting value when $MEx_n < M - J$, which is obviously the condition for stability of the queuing process. Let $\phi_i(s)$ denote the generating function corresponding to the limiting distribution of b_{mM+i} for $i = 1, 2, \cdots, M$. Then

$$\phi_i(s) = \lim_{m \to \infty} Es^{b_{mM+i}} \; i = 1, 2, \cdots, M.$$

Calculating the generating functions of both sides of (1) and then letting m tend to ∞, we can derive equations for $\phi_i(s)$. Let $\chi(s) = Es^{x_n}$, and note that $\chi(s)$ factors out on the right-hand side of (1) since x_n is independent of b_n. Then we have for $i = 0, \cdots, M - 1$,

$$\begin{aligned} \phi_{i+1}(s) &= \phi_i(s)\chi(s) \qquad \text{for} \quad i \equiv S(M) \\ \phi_{i+1}(s) &= [s^{-1}\phi_i(s) + (1 - s^{-1})p_{io}]\chi(s) \qquad \text{for} \quad i \not\equiv S(M), \end{aligned} \tag{2}$$

where

$$p_{i0} = \lim_{n \to \infty} \Pr\{b_{nM+i} = 0\} \quad \text{and} \quad \phi_0 = \phi_M. \tag{3}$$

We can write the above equations (2) in matrix form as follows. Let ϵ_i, δ_i be s and 0 respectively if $i \equiv S(M)$. For $i \not\equiv S(M)$, let $\epsilon_i = \delta_i = 1$. Then

$$\begin{bmatrix} \phi_1(s) \\ \phi_2(s) \\ \cdot \\ \cdot \\ \cdot \\ \phi_M(s) \end{bmatrix} = \frac{\chi(s)}{s} \begin{bmatrix} 0 & 0 & \cdots & 0 & \epsilon_M \\ \epsilon_1 & 0 & \cdots & 0 & 0 \\ 0 & \epsilon_2 & \cdots & 0 & 0 \\ & & & \cdot & \\ \cdot & & & \cdot & \\ \cdot & & & \cdot & \\ \cdot & & & 0 & 0 \\ 0 & 0 & \cdots & \epsilon_{M-1} & 0 \end{bmatrix} \begin{bmatrix} \phi_1(s) \\ \phi_2(s) \\ \\ \\ \phi_M(s) \end{bmatrix}$$

$$+ (1 - s^{-1})\chi(s) \begin{bmatrix} p_{M,0}\delta_M \\ p_{1,0}\delta_1 \\ \cdot \\ \cdot \\ \cdot \\ p_{M-1,0}\delta_{M-1} \end{bmatrix} \tag{4}$$

Using the symbol $\boldsymbol{\phi}$ for the vector $(\phi_1, \cdots, \phi_M)^t$, $\mathbf{p}$ for $(\delta_M p_{M,0}, \cdots, \delta_{M-1}p_{M-1,0})^t$, $A(s)$ for the matrix consisting of entries either 1, 0, or s, and I for the identity matrix, we can rewrite (4) as

$$\left[I - \frac{\chi(s)}{s} A(s) \right] \boldsymbol{\phi}(s) = (1 - s^{-1})\chi(s)\mathbf{p}. \tag{5}$$

For every s such that the matrix on the left is invertible, let $B(s)$

denote its inverse. Represent $B(s)$ by its rows $B_i(s)$ for $i = 1, 2, \cdots,$ M. Then we can derive the following equation for B_1.

$$B_1 = \frac{s^{M-J}}{s^{M-J} - \chi^M(s)}$$
$$\times \left[1, \prod_{j=2}^{M} \left[\frac{\epsilon_j}{s} \right] \chi^{M-1}(s), \cdots, \prod_{j=M-1}^{M} \left[\frac{\epsilon_j}{s} \right] \chi^2(s), \frac{\epsilon_M}{s} \chi(s) \right]. \qquad (6)$$

The other rows of B can be recursively computed as follows: Let $\ell_i = \left(0, 0, \cdots, \overset{i}{1}, 0, \cdots, 0 \right)$ then

$$B_{i+1} = \ell_{i+1} + \chi(s) \frac{\epsilon_i}{s} B_i \qquad i = 1, 2, \cdots, M-1. \qquad (7)$$

Since $\chi(1) = 1$, $s^{M-J} - \chi^M(s) = 0$ for $s = 1$. It can be shown by Rouché's theorem that $s^{M-J} - \chi^M(s)$ has $M - J - 1$ distinct roots strictly inside the unit disk.[7] Each of the generating functions ϕ_i satisfies

$$\phi_i(s) = (1 - s^{-1}) \chi(s) B_i \mathbf{p} \qquad (8)$$

for every s for which $B_i(s)$ is well defined ($B_i\mathbf{p}$ is the product of B_i and $\mathbf{p}$ viewed as matrices). Since $\phi_i(s)$ is analytic on the unit disk, the representation (8) extends to the roots of $s^{M-J} - \chi^M(s)$ that lie within the unit disk if $\mathbf{p}$ is such that the singularities of $[s^{M-J} - \chi^M(s)]^{-1}$ are removed. We illustrate this in detail by using an important special case of the model presented here. This is the case when, out of the M slots, the last J slots are needed for line switched or steady traffic. Only the first $M - J$ slots are used for transmitting asynchronous traffic. In this case, the matrix $A(s)$ has the form:

$$\begin{bmatrix}
0 & 0 & \cdots & & & \cdots & 0 & s \\
1 & 0 & \cdots & & & \cdots & 0 & 0 \\
0 & 1 & \cdots & & & \cdots & 0 & 0 \\
\vdots & & \ddots & & & & & \\
0 & & & 1 & & \cdots & 0 & 0 \\
0 & & \cdots & s & & \cdots & 0 & 0 \\
0 & & \cdots & & s & \cdots & 0 & 0 \\
\vdots & & & & & \ddots & & \\
0 & & \cdots & & & \cdots & s & 0
\end{bmatrix}$$

and the vector $\mathbf{p} = (0, p_{1,0}, p_{2,0}, \cdots, p_{M-J,0}, 0, \cdots, 0)^t$.

Let $\xi_1 = 1, \xi_2, \xi_3, \cdots, \xi_{M-J}$ denote the roots of $s^{M-J} - \chi^M(s)$ that are in the unit disk. At these roots, the first row of B, excluding the scalar multiplier in (6), can be expressed as

$$\left[1, \frac{\chi^{M-1}(\xi_i)}{\xi_i^{M-J-1}}, \cdots, \chi^J(\xi_i), \chi^{J-1}(\xi_i), \cdots, \chi^2(\xi_i), \chi(\xi_i)\right].$$

Hence a choice of **p** that will cancel the singularities of the scalar multiplier in (6) satisfies the following equation

$$\begin{bmatrix} 1 & \chi(\xi_1)/\xi_1 & \chi^2(\xi_1)/\xi_1^2 & \cdots & \chi^{M-J-1}_{(\xi_1)}/\xi_1^{M-J-1} \\ 1 & \chi(\xi_2)/\xi_2 & \chi^2(\xi_2)/\xi_2^2 & \cdots & \chi^{M-J-1}_{(\xi_2)}/\xi_2^{M-J-1} \\ \cdot & & & & \\ \cdot & & & & \\ \cdot & & & & \\ 1 & \cdots & & & \end{bmatrix} \begin{bmatrix} p_{M-J,0} \\ \cdot \\ \cdot \\ \cdot \\ p_{2,0} \\ p_{1,0} \end{bmatrix} = \begin{bmatrix} p \\ 0 \\ 0 \\ \\ \\ 0 \end{bmatrix}, \qquad (9)$$

where $\;= \sum_{i=1}^{M-J} p_{i,0}$. Except for the first equation, the relations in (9) express the fact that the numerator of (8) vanishes at $1, \xi_2, \cdots \xi_{M-J}$. The vector $(p_{1,0}, p_{2,0}, \cdots p_{M-J,0})^t$ appears reversed in (9), so the coefficient matrix can be written as a Vandermonde matrix, which we denote $V = V(\xi_1, \xi_2, \cdots \xi_{M-J})$. The nonsingularity of V follows from the distinctness of the $\{\xi_j\}_1^{M-J}$

Therefore, if the scalar p is known, the vector **p** is determined uniquely from (9). Differentiating (2) with respect to s and letting s approach 1 gives $p = (M - J) - MEx_n$. Hence **p** and then $\phi(s)$ can be determined uniquely from (5).

Theorem 1: *Let $MEx_n < (M - J)$, then there always exists constants $\{p_{i,0}\}_{i=1}^M$ such that the functions $\{\phi_i(s)\}_{i=1}^M$ are analytic in the unit circle. Moreover, these constants are uniquely determined by the condition $\phi_1(1) = 1$.*

In the general case, the components of **p** that are zero depend on the set $\mathscr{S}$. Let $i_1 < i_2 < \cdots < i_{M-J}$ represent the indices not included in $\mathscr{S}$. By definition, $\delta_j = 0$ whenever $j \neq$ some i_m. Hence the unknown constants in eq. (8) are $p_{i_m,0}$, $m = 1, 2, \cdots, M - J$. Once again, using the arguments above, we can arrive at the $(M - J)$ equations that the $p_{i_m,0}$ satisfy in order that $\phi_i(s)$ have no singularities inside the unit circle. These correspond to (9) and will be denoted by (9′); however, the matrix appearing on the left-hand side of (9′) is no longer Vandermonde. Let V′ be the matrix associated with (9′):

$$\sum_{m=1}^{M-J} V'_{1m}\, p_{i_m,0} = \rho$$

$$\sum_{m=1}^{M-J} V'_{\ell m}\, \rho\, {}_{i_m,0} = 0 \qquad \text{for} \quad M - J \geq \ell > 1 \tag{9′}$$

and the elements $V'_{\ell m}$ are

$$V'_{\ell m} = \left[\prod_{j=i_m}^{M} \frac{\epsilon_j}{s} \right] \chi(s)^{M-i_m+1} \qquad m = 1, 2, \cdots, M.$$

There must be at least one solution for these equations whenever $MEx_n < M - J$, since the invariant distribution corresponding to Q_i, which exists by Theorem 1, satisfies (9′) and $\phi_i(s)$ has no singularities inside the unit circle. Moreover, we will show that any solution of (9′) gives the unique invariant distribution corresponding to Q_i. Corresponding to any solution of (9′), we can find functions $\hat{\phi}_i(s)$ from (8) such that the associated sequences $\{\hat{p}_{ij}\}$ with $\sum_{j=0}^{\infty} \hat{p}_{ij} s^j = \hat{\phi}_i(s)$ are absolutely summable, since $\hat{\phi}_i(s)$ will have no singularities inside the unit disk. Inspection of (2) shows that the vectors $\hat{\pi}_i = \{\hat{p}_{ij}\}_{j=0}^{\infty}$ satisfy

$$\hat{\pi}_{i+1} = P_i \hat{\pi}_i \qquad i = 1, 2, \cdots, M - 1$$

and

$$\hat{\pi}_1 = P_M \hat{\pi} M.$$

Hence, from the definition of Q_i,

$$\hat{\pi}_i = Q_i \hat{\pi}_i.$$

Since the Markov chain corresponding to Q_i is positive recurrent, the (ℓ, m) element of Q_i^n, the nth power of Q_i, tends to $p_{i\ell}$ the ℓth component of the invariant distribution corresponding to Q_i. From the above equation,

$$\hat{p}_{i\ell} = \sum_m Q_i^n(\ell, m)\, \hat{p}_{im}.$$

Since the sequence $\{\hat{p}_{ij}\}$ is absolutely summable taking limits of both sides and interchanging limits, we have

$$\hat{p}_i \ell = p_i \ell \sum_m \hat{p}_{im}.$$

However, from the first of equation of (9′) we can show that $\sum_{m=0}^{\infty} \hat{p}_{im} = 1$. Hence $\hat{p}_{i\ell} = p_{i\ell}$, the unique invariant density corresponding to Q_i. Hence we have shown that (9′) has a unique solution whenever $MEx_n < M\text{-}J$. Q.E.D.

Remark: Usually, the queue size β at a "random" time is of interest. The generating function of β, denoted by $\psi((s)$, is the average of $\phi_i's$, i.e.,

$$\psi(s) = M^{-1} \sum_1^M \phi_i(s).$$

V. SOME SPECIAL CASES

In the special case $J = 1$, $M = 2$, it is easy to express $\phi_i(s)$ in closed form. For Poisson arrivals, we have

$$\begin{bmatrix} \phi_1(s) \\ \phi_2(s) \end{bmatrix} = \frac{e^{\lambda(s-1)}(s-1)(1-2\lambda)}{s - e^{2\lambda(s-1)}} \begin{bmatrix} e^{\lambda(s-1)} \\ 1 \end{bmatrix}.$$

So

$$\psi(s) = \frac{(1-2\lambda)}{2} \frac{e^{\lambda(s-1)}(s-1)}{s - e^{2\lambda(s-1)}} (e^{\lambda(s-1)} + 1).$$

The probability of an empty buffer is

$$p_0 = \psi(0) = \frac{(1-2\lambda)}{2}(1 + e^{\lambda})$$

and, of course,

$$\lim_{\lambda \to (1/2)} p_0 \text{ and } \lim_{\lambda \to 0} p_0 = 1.$$

The mean buffer content

$$\bar{\beta} = \psi(s)\Big|_{s=1} = \frac{3}{2}\lambda + \frac{2\lambda^2}{1-2\lambda}.$$

As is intuitively obvious,

$$\lim_{\lambda \uparrow (1/2)} \bar{\beta} = \infty \text{ and } \lim_{\lambda \downarrow 0} \bar{\beta} = 0.$$

For the variance of the buffer content, we have

$$\begin{aligned} \text{Var}(\beta) &= \psi'(\text{s}) + \psi'(\text{s}) - (\psi'(\text{s}))^2|_{s=1} \\ &= \frac{5}{8}\lambda^2 + \frac{13}{6}\frac{\lambda^3}{1-2\lambda} + \frac{2\lambda^2}{(1-2\lambda)^2} + \bar{\beta}^2 - \bar{\beta}, \end{aligned}$$

and again

$$\lim_{\lambda \uparrow (1/2)} \text{Var}(\beta) = \infty$$

while

$$\lim_{\lambda \downarrow 0} (\beta) = 0.$$

Another simple case is when $M \to \infty$ with J fixed, so the relative time when the server is absent tends to zero. Asymptotically, the system behaves like a discrete time $M|\mathrm{D}|1$ queue with time quantum equal to one service time. The analysis of this queue is given in the excellent survey paper of Ref. 8. The generating function of β is

$$\left[\frac{s-1}{s-\chi(s)}\right]\chi(s)(1-\lambda).$$

So, for the Poisson case,

$$p_0 = (1-\lambda),$$

$$\bar{\beta} = \lambda\left[1 + \frac{\lambda}{2(1-\lambda)}\right],$$

and

$$\operatorname{Var} \beta = \lambda^2 + \frac{4}{3}\left[\frac{\lambda^3}{1-\lambda}\right]\frac{\lambda^4}{2(1-\lambda)^2} + \bar{\beta} - \bar{\beta}^2.$$

A related queuing problem is introduced in Ref. 7. They undertake a discrete time analysis of the waiting room occupancy in a situation where a shuttle visits every M time units, whereupon up to a maximum of K occupants are removed. If less than K occupants confront the arriving shuttle, all are removed. As in our analysis, the arrivals are arbitrary i.i.d. variables. The generating function $\xi(s)$, of the equilibrium density of waiting room occupancy as seen by the arriving shuttle, is determined (see Ref. 9 for more detail). In the special case where $K = 1$, if we set $J = M - 1$ in our analysis, then $\xi(s)$ is the same as $\phi_0(s)$.

Another variation of the process analyzed in the last section occurs when the synchronous packets are also queued and hence are subject to delay and loss. Let the resulting buffer process be denoted by b'_n. Then

$$b'_{n+1} = (b'_n - 1)^+ + x_n + (1 - u_n).$$

Starting with $\mathrm{b}_0 = b'_0 = 0$, we can show by induction that for each arrival stream realization, b_n and b'_n agree to within one packet, that is, with probability one $|\, b_n - b'_n \,| \le 1$ uniformly in n. Thus the analysis in the previous section aids in estimating the jitter suffered by the synchronous input stream. Such jitter considerations, which are basic to the emerging topic of packetized speech, will not be explored here.

VI. EXTENDING THE RESULTS

6.1 Obtaining delay densities from the buffer densities

So far in this paper, we have focused on the problem of obtaining the buffer density; however, in many applications the density of delay is also of importance. It is reasonable to expect several system requirements to be in effect, as, for example

$$\begin{aligned}&\Pr[\text{buffered packets} > 32] < 10^{-7}\\&\Pr[\text{waiting time} > 500 \text{ ms}] < 10^{-1}\\&E[\text{waiting time}] < 250 \text{ ms.}\end{aligned}$$

In some applications, satisfying the first objective obviates the other two. However, when the delay requirements are more crucial, it is easy to obtain the delay density numerically as it is simply related to $\phi(s)$, as we now show.

Since we are using a discrete time model, we consider all arrivals to occur at integer times. The discrete time model of arrivals can be considered to arise from a continuous-time arrival process in which all arrivals on $]n, n+1]$ are associated with a time of arrival $n+1$. For the purpose of computing delay, it is essential to retain an order relationship for the arrivals at time n.

The packets to be served ahead of a typical arrival y in $]n, n+1]$ can be partitioned into three groups:

(*i*) The packets already in the buffer at time n.

(*ii*) The number of asynchronous packets arriving in $]n, n+1]$ ahead of y, denoted by x_n^*. For example, if x_i is Poisson, an arrival occurs according to a uniform distribution so the generating function of x_n^* is

$$\chi^*(z) = \int_0^1 e^{\lambda\tau(z-1)}\, d\tau = \frac{e^{\lambda(z-1)}-1}{\lambda(z-1)}.$$

(*iii*) The synchronous packets arriving during the transmission of asynchronous packets in the system before the transmission of y.

Let E_n denote the sum of (*i*) and (*ii*) above. Note the generating function of E_n is the product of the generating functions of b_n and x_n^*. The delay caused by interruptions, as mentioned in (*iii*) above, can be derived easily from the nature of $\mathscr{S}$ and depends only on E_n and the slot in the frame that corresponds to n. For a "random" arrival, the delay density is found by averaging over the densities corresponding to the M slots. Of course, an additional delay unit must be included to account for the service of the packet whose delay distribution is being calculated. In conclusion, when delay performance is a dominant consideration, the delay density functions can also be computed from $\phi(s)$.

Once the delay density and/or the density of buffered packets is computed for the parameter range of interest, the network designer can determine the tradeoffs among

Delay and/or buffer size.
Trunk capacity.
Average packet arrival rate.
Percentage of transmission facilities devoted to line switching (or batch service).

Such information, along with market and revenue forecasts and resource cost estimates, allows the designer to determine an optimum packet service-line service mix for the projected environment.

6.2 Extension to batch arrival model

The process $\{x_n\}_{-\infty}^{\infty}$ of packet arrivals on the nth time interval can be considered to be realized from an underlying message arrival process $\{m_n\}_{-\infty}^{\infty}$ where each message contains a random number of packets $\{l_n\}_{-\infty}^{\infty}$. So both $\{m_n\}_{-\infty}^{\infty}$ and $\{l_n\}_{-\infty}^{\infty}$ are processes of i.i.d variables independent of each other and with (different) underlying densities with values in the nonnegative integers. If $M(z)$ and $L(z)$ are the corresponding generating functions of m_n and l_n, then $M(L(z))$ is the generating function of x_n.

In some applications, fast switching may be employed so that a fraction of a packet is the basic unit switched (e.g., a byte in a system employing fixed-size 1024-bit packets). To analyze such situations, one can take the basic time quantum in the mathematical model to be the time required to transmit a subpacket. Then a packet corresponds to a message and a subpacket to a packet in the above discussion. For the example cited, $L(z) = z^{128}$.

So the model is accommodating whether we view arrivals as single packets or batches of packets. The most useful case is when the arrivals are Poisson and a geometric number of packets is associated with each arrival. In this case,

$$M(z) = \exp \lambda(z-1) \quad \text{and} \quad L(z) = z(1-p)/(1-pz).$$

So

$$\chi(z) = \exp \lambda[(z-1)/(1-pz)].$$

The mean number of arrivals per time slot is $\lambda(1-p)^{-1}$. For purposes of obtaining the delay density in the Poisson case, we need $\chi^*(z)$ which, by a straightforward integration, is

$$\chi^*(z) = \frac{(1-pz)}{\lambda(z-1)} e^{\lambda(z-1)/(1-pz)} - 1.$$

6.3 A more general problem

Here our objective is to point out a seemingly more complex queuing situation where our methods still apply. The type of queuing situation we discuss plays an important role in the analysis of subframe packet switching in Ref. 2.

Consider the situation in Fig. 2. The server visits the various queues according to some fixed periodic pattern. The input to various queues are independent. During every visit, the server completes servicing one job if there are any jobs waiting. Each input stream represented by an arrow can have batch arrivals according to any arbitrary distribution function (it is permissible for different distributions to correspond to different inputs). The question is: What is the buffer and delay distribution as perceived by one of the inputs (indicated by the only dashed arrow inputting the starred queue)?

A little reflection reveals that, from the perspective of one input, the problem is no different than the one we have already solved. The equivalence would be immediate if the dashed arrow denoted the sole

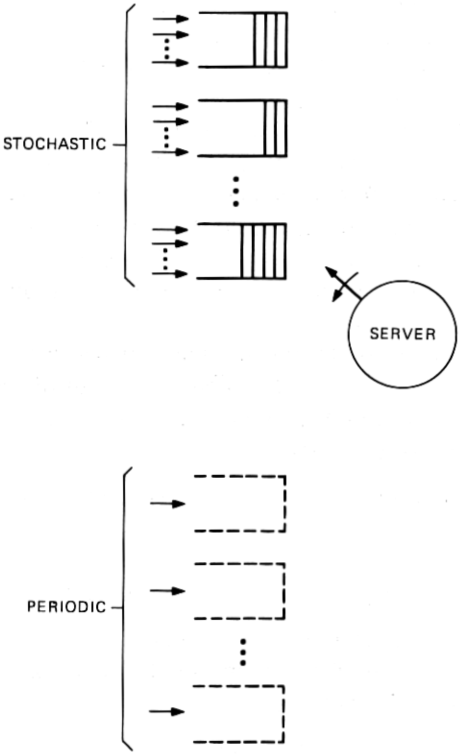


Fig. 2—Generalized system.

input to its queue. After all, from the perspective of the input in question, the structure of what the server does while he visits the unstarred queues is irrelevant to the buffer and delay performance. All that matters is the periodic pattern of the service availability and unavailability to the starred queue. When the starred queue has other inputs, there is no real complication. To obtain the buffer density, simply replace the parallel inputs by a single input stream choosing a single batch density representing the various batch densities with their corresponding frequencies. A similar method accommodates delay with the adjustment that the time the batch, whose delay is being computed, spends in service is represented as a random choice from the distribution of batch sizes from the dotted input.

Paul Lue of Bell Laboratories in Holmdel, who had an earlier version of this paper, has formulated a further generalization involving multiple periodic parallel servers for Fig. 2. He plans to report his successful analysis in the future.

VII. DISCUSSION OF NUMERICAL WORK

7.1 The computer program

For input distributions which have a generating function of the form

$$\chi(s) = e^{[\lambda(s-1)/1-ps]},$$

a program for obtaining the equilibrium density of buffer size and delay as a function of (λ, M, J, p) was developed. The core of the program is the computation of ($\phi_1(s)$, $\phi_2(\mathrm{s})$, $\cdots$, $\phi_M(s)$) and a generating function inversion routine.

Double precision was used throughout the program since at the present time overflow probabilities of the order of 10^{-7} are of interest. The determination of the $\phi_m(s)$ includes finding the location of the "apparent" poles in the unit disk. While a straightforward search of the disk for poles does the trick, the Newton Raphson routine suggested in Ref. 2 is preferred for its speed.

For inverting a generating function, we employ a fast Fourier transform (FFT) program. The use of the FFT in this situation stems from the observation that, if we replace s by $e^{i\omega}$, the generating function is then a Fourier series on the boundary of the unit disk. The Fourier coefficients are the probabilities of interest. In using the FFT, the generating function is represented by its values at the discrete sample points

$$\{e^{\sqrt{-1}(2\pi\ell/L)}\}_{\ell=0}^{L},$$

where L is taken to be sufficiently large to obtain the accuracy required.

As indicated in the previous sections, once the $\phi_m(s)$ are known, the analytical determination of the delay distribution is also possible. Programmed implementation of the procedure for obtaining the delay distributions from the $\{\phi_n\}_1^M$ is straightforward.

Usually the program output routine is set only to provide the densities (buffer or delay) averaged over an entire frame as the more refined intraframe densities are of secondary interest.

7.2 A peculiarity of the numerical data

With each irreducible fraction $r = p/q$ in (0, 1), associate a frame of size q in which the first p time slots for asynchronous data are followed by $q - p$ time slots for synchronous data (see Fig. 3). Let $r_n = p_n/q_n$ be such that $\lim_{n\to\infty} \mathrm{r}_n = 1/2$ and $\lim_{n\to\infty} q_n = \infty$. Mean buffer size $\bar{\beta}$ and mean delay $\bar{\alpha}$ are discontinuous functions of r. Indeed, the buffer size and delay of those packets arriving in the first half of the last $q_n - p_n$ slots are going to infinity since they receive no service in the second half of the last $q_n\, p_n$ slots.

The above argument points out that two frame organizations can be arbitrarily close in terms of the relative number of time slots devoted to packet switching yet the mean delay and buffer sizes of both systems can differ by an arbitrarily large number. The preceding discussion also shows that interpolation of statistics to an intermediate r value is a perilous calculation. However, interpolation to an r point from points with the same denominator (frame size) can be reasonable.

7.3 Numerical results

The program for determining the distribution of β was exercised for numerous cases with $M \leq 16$ and $\rho_{\text{eq}} \triangleq \lambda M/(M - J)(1 - p) < 1$, the latter inequality being required for stability. For illustrative purposes, Tables I and II summarize the statistics associated with a wide variety of examples. The parameters for Tables I and II differ only in that, in I, single packet messages are assumed, while, in II, the messages are of random (geometric) size with mean five (i.e., $p = 0.2$). The 10^{-x} headers

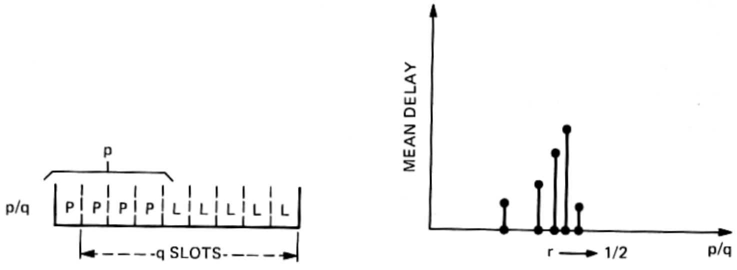


Fig. 3—Discontinuity of buffer and delay statistics.

Table I—Buffer size and delay statistics for cases where all slots but one in a frame are allotted for asynchronous data

		$\lambda = 0.2$						$\lambda = 0.4$						$\lambda = 0.6$					
		Mean	Var	10^{-2}	10^{-5}	10^{-6}	10^{-7}	Mean	Var	10^{-2}	10^{-5}	10^{-6}	10^{-7}	Mean	Var	10^{-2}	10^{-5}	10^{-6}	10^{-7}
$M = 2$	Buffer	0.433	0.497		7	9	10	2.200	5.65		27	32	37			Unstable			
$J = 1$	Delay	1.17	3.05	5				4.50	22.4	22									
$M = 3$	Buffer	0.328	0.352		6	7	8	0.974	1.38		12	15	17	4.83	23.6				
$J = 1$	Delay	0.63	0.793	5				1.42	2.97	8				6.98	50.2	34			
$M = 4$	Buffer	0.297	0.315		5	6	7	0.800	1.02		10	12	14	2.34	5.80		27	32	38
$J = 1$	Delay	0.48	0.572	4				0.99	1.67	6				2.89	10.1	15			
$M = 5$	Buffer	0.282	0.297		5	6	7	0.729	0.897		9	11	13	1.84	3.72		21	25	29
$J = 1$	Delay	0.41	0.487	4				0.82	1.27	5				2.06	5.55	11			
$M = 6$	Buffer	0.272	0.285		5	6	7	0.690	0.832		9	11	12	1.62	2.97		19	22	26
$J = 1$	Delay	0.36	0.436	4				0.721	1.08	5				0.70	4.02	10			

of the columns refer to an upper bound on the probability that the random variable (buffer size or delay) takes a value larger than the number entered in the column. For example, in the first subtable

$$\Pr[\text{delay} > 5 \text{ packets}] < 10^{-2}$$

and

$$\Pr[\text{buffer size} > 11 \text{ packets}] < 10^{-7}.$$

We use "packets" as the unit of delay since this is readily converted to time, because in our model we assumed that the duration of one time slot is the transmission time for one packet.

To emphasize that more refined statistics are easily obtained, Table III presents the intraframe details for a specific case.

Figures 4 and 5 stem from Tables I and II and are used below in a pair of hypothetical examples we include to show how a designer could make use of the available numerical capability.

Example I. Consider a situation in which a 56-kb/s trunk is available for transmission of 1024 bit packets. The packet arrival process is Poisson, and one packet is associated with each arrival. A 32-packet buffer is available. It is required that the probability of a lost packet not exceed 10^{-6}. If $\lambda = 0.4$ (23 packets per second), the question is how

Table II—Second-order statistics for buffer and delay for cases where all slots but one in a frame are allotted for asynchronous data. Multiple packet messages of mean size five (p^{-1})

		$\frac{\lambda}{1-p} = 0.2$		$\frac{\lambda}{1-p} = 0.4$		$\frac{\lambda}{1-p} = 0.6$	
		Mean	Var	Mean	Var	Mean	Var
$M = 2$	Buffer	3.072	35.4	11.9	13.75	Unstable	
$J = 1$	Delay	12.3	132.7	176.2	227.8		
$M = 3$	Buffer	2.02	21.6	6.53	86.5	12.7	223.5
$J = 1$	Delay	8.85	76.5	13.1	146.5	11.6	230.8
$M = 4$	Buffer	1.74	18.1	5.17	65.5	12.0	176
$J = 1$	Delay	7.58	61.5	11.2	114	14.8	203
$M = 5$	Buffer	1.60	16.5	4.59	56.7	10.6	151
$J = 1$	Delay	6.95	53.1	10.1	97.9	14.5	177
$M = 6$	Buffer	1.52	15.6	4.27	51.9	9.76	137
$J = 1$	Delay	6.58	48.2	9.45	88.2	13.8	161

Table III—Intraframe buffer size statistics (frame size six, server absent for four slots, $\lambda = 0.2$, $\rho = 0.6$)

	Time Slot						
	1	2	3	4	5	6	$M\|D\|1$
Mean	1.361	0.850	0.561	0.761	0.961	1.161	1.05
Var	1.712	1.331	.912	1.112	1.312	1.512	1.43
10^{-6}	15	15	14	14	15	15	

much capacity can we devote to line switching. Figure 4 shows that the answer is 50 percent. If, at a subsequent date, we have that λ has increased to 0.6, then only 25 percent of capacity can be devoted to line switching.

Figure 5 shows that, in the case λ = 0.6, the 99-percent delay is about 250 ms, while the mean is about 55 ms.

Example II: The ability to compute the density of β (J, M, λ, p)

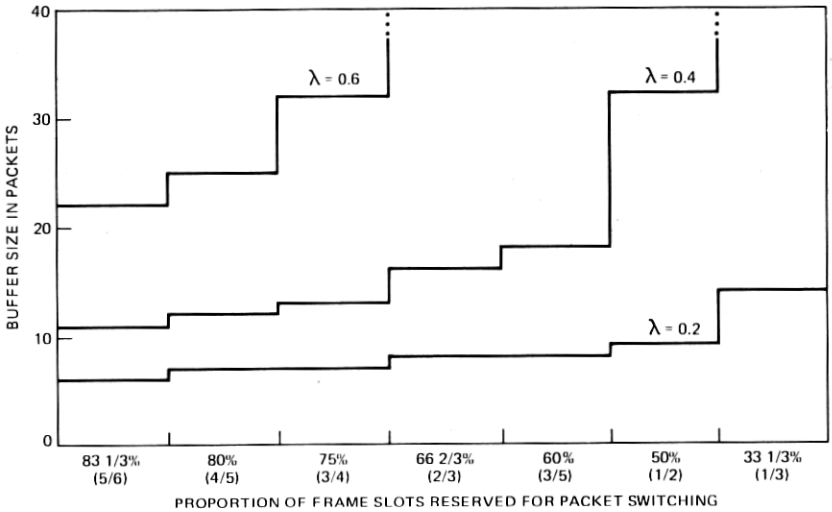


Fig. 4—10^{-6} loss threshold for various mixes of packet and line switching.

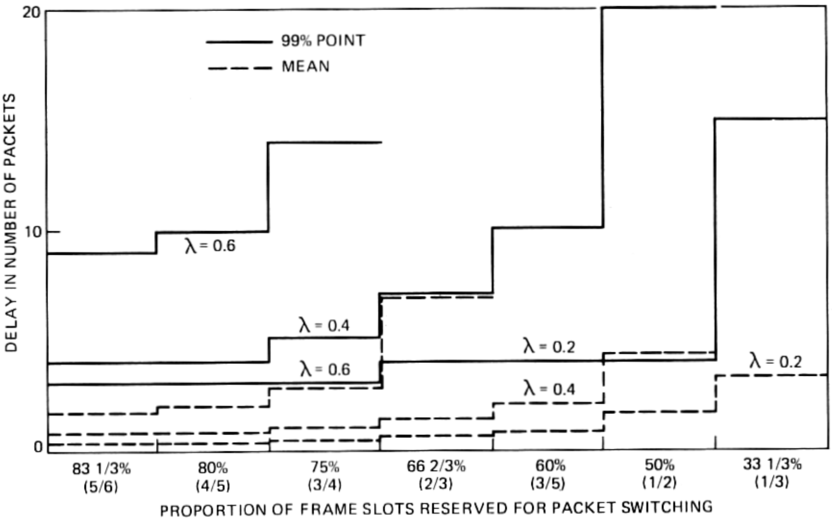


Fig. 5—Delay (mean and 99-percent point) for various mixes of packet and line switching.

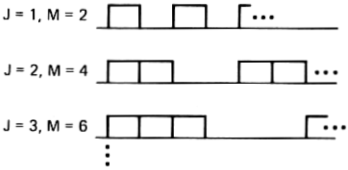
also offers useful information relating to switching considerations. To see this, consider a hypothetical situation in which a designer wishes to devote 50 percent of capacity to packet switching and the remaining for line switching. The question of organizing the frame arises. Any of the patterns shown in Fig. 6a will suffice. Intuitively, one would expect buffer occupancy statistics to degrade as one moves down the list. Yet the reduced load on the processor for attending to switching among the two types of customers would make the longer frames attractive. For $\lambda = 0.4$, the degradation is illustrated in Fig. 6b. The numerical capability reported here enables one to determine the optimal operating point on the basis of projected switching costs and sensitivity of revenue to performance.

This example and the discussion in Section 7.2 point out that the *amount* of trunk capacity devoted to asynchronous traffic of a specified intensity is not enough information for one to determine the buffer and delay distributions. The details of frame organization can be essential for obtaining accurate statistics.

7.4 The role of an $M|D|1$ model in computations

Previous analysis of such systems used simulation or approximation to obtain buffer and delay statistics. Yet simulations are usually too expensive for obtaining the extremal statistics preferred by the requirements engineer. On the other hand, the accuracy of "approximations" such as using an $M|D|1$ model could not be appraised. It is reasonable to require that the $M|D|1$ model have the same utilization $\rho_{\text{eq}} = [M/(M{-}J)]\,\lambda$ and the same throughput as the hybrid model. Indeed, ρ_{eq} and throughput uniquely determine an $M|D|1$ model.

With the hybrid multiplexor solution in hand, one can evaluate the above nonexact methods in the parameter range of interest. While a thorough exploration of this issue is beyond the scope of this paper, we shall include some comparisons that were made for the $J = 1$ case. For $M = 2$, the errors range as high as 28 percent and then decrease as M increases, as one would expect. It appears that, insofar as the tail probabilities which only register order of magnitude are concerned,



(a)

	J/M 1/2	2/4	3/6	4/8	5/10
MEAN	2.200	2.265	2.351	2.450	2.557
VAR	2.126	2.173	2.243	2.341	2.453

(b)

Fig. 6—(a) A sequence of possibilities for attaining a 50-percent mix ($J = M/2$). (b) Central moments of buffer size vs frame length for 50-percent mix ($J = M/2$).

one might as well use $M|D|1$ formulas. Of course, if a much more finely resolved graph is meaningful, the $M|D|1$ approximation may no longer suffice. In initial stages of the evaluation of computer networks, it is unusually unrealistic to expect to know the arrival rate to a tighter tolerance than 10 percent, and so the $M|D|1$ analysis of overflow for the 1 out of M cases provides a useful simplified model.

Nonetheless, there are parameter ranges where the $M|D|1$ approximation is useless. To see this, fix λ and take $J = M/2$. Consider what happens as M increases. Note the $M|D|1$ approximation has $\rho_{\text{eq}} = 2\lambda$ and the throughput is independent of M. For the hybrid model, the packets arriving in the third quarter of a frame must wait out at least the last quarter before they are eligible for service. So, as $M \rightarrow \infty$, the mean buffer size and mean delay increase without bound and the error in using an $M|D|1$ approximation goes to infinity. This example is by no means pathological, as it addresses precisely those cases that arise in the switching study mentioned in Example II. The dotted line in Fig. 7 gives the $M|D|1$ result.

The $M|D|1$ model is useful in comparing the hybrid system with a system providing separate dedicated facilities for synchronous and asynchronous data. For example, Fig. 7 compares the performance between hybrid and dedicated implementations, and an $M|D|1$ model is used to provide numbers for the latter. With reference to Example II in Section 7.3, the dotted line of Fig. 7 indicates the average delay performance of a competitive system using dedicated trunks. In Fig. 7 we see a region where dedicated trunks of a given capacity do not perform as well as a hybrid system that devotes the same capacity to asynchronous traffic.

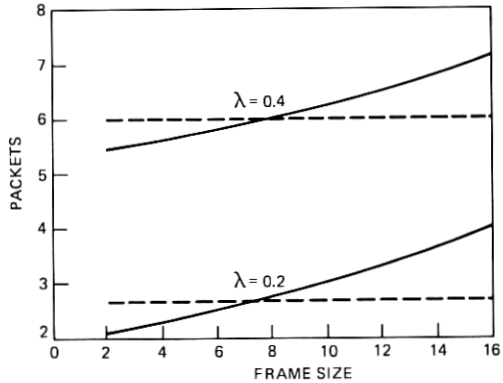


Fig. 7—Mean delay for various realizations of a 50-percent packet service rate.

REFERENCES

1. G. J. Foschini, B. Gopinath, and J. F. Hayes, unpublished work.
2. K. Kummerle, "Multiplexor Performance for Integrated Line and Packet-Switched Traffic," ICCC Stockholm Second International Conference on Computer Communication, Stockholm, 1974, pp. 507–515.
3. M. J. Fischer, "Analysis and Design of Loop Service Systems Via a Diffusion Approximation," Defense Communications Engineering Center Report.
4. M. J. Fisher and T. C. Harris, "A Model for Evaluating the Performance of an Integrated Circuit- and Packet-Switched Multiplex Structure," IEEE Trans. Commun., *COM-24* (February 1976), pp. 195–202.
5. Izzet Sahin and U. Narayan Bhat, "A Stochastic System with Scheduled Secondary Inputs," Operations Research, *19* (1971), pp. 436–446.
6. F. G. Foster, "On the Stochastic Matrices Associated with Certain Queueing Problems," Ann. Math. Statist. *24* (1953), pp. 355–360.
7. P. E. Boudreau, J. S. Griffin, and M. Kac, "An Elementary Queueing Problem," American Mathematical Monthly, October 1962, pp. 713–724.
8. H. Kobayashi and A. G. Konheim, "Queueing Models for Computer Communications System Analysis," IEEE Trans. Commun. (Special Issue on Computer Communications), *COM-25*, No. 1 (January 1977), pp. 2–28.
9. A. G. Konheim and R. Meister, "Service in a Loop System," J. Ass. Comput. Mach., *19*, No. 1 (January 1972), pp. 92–108.