AT&T Bell Laboratories Technical Journal
Vol. 63, No. 8, October 1984
Printed in U.S.A.

Analysis of a TDMA Network With Voice and Data Traffic

By M. L. HONIG*

(Manuscript received May 16, 1983)

An analysis of an integrated voice-data network with Demand Assignment Time Division Multiple Access (TDMA) is presented using the following model: (1) voice calls that cannot be serviced are blocked, whereas requests to transmit data messages are queued; (2) no traffic boundaries are assumed, i.e., any new traffic arrival may be assigned to any unassigned time slot; (3) message lengths are exponentially distributed with the mean voice message length assumed to be much larger than the mean data message length; (4) traffic requests are generated according to two independent Poisson processes; and (5) time slot assignments are made instantaneously and no priorities are assumed. Such a model applies to a single-channel TDMA network in which voice and data traffic arrivals are serviced on a first-come first-served basis. An approximate analysis, based upon physical insight, is presented that yields the blocking probability for voice messages, the mean number of queued data requests, and the mean value of the peaks of the data queue process. Comparisons with simulation results indicate that the analytical results are very accurate. Performance curves are presented and compared with analogous results for TDMA networks that handle only one traffic type.

I. INTRODUCTION

The popularity of integrated voice-data networks has motivated numerous analyses of associated network queueing models.[1–7] This paper presents an analysis of a voice-data network using Demand Assignment Time Division Multiple Access (DA/TDMA). This work

* AT&T Bell Laboratories; present affiliation Bell Communications Research, Inc.

evolved from a study of a multichannel DA/TDMA protocol that handles different traffic types, and that has received much attention in the context of satellite communications.[8,9] These networks typically have a few hundred megabits of capacity and can be used to carry real-time digital voice traffic in addition to data traffic. Analysis in Ref. 10 indicates that a large multichannel TDMA network (i.e., a network with more than 100 communicating traffic nodes) behaves much like a single-channel TDMA network with an equivalent number of time slots per frame. We therefore concentrate on the simpler single-channel network and attempt to characterize its performance when used to handle both real-time voice and data traffic. The results in this paper carry over to the multichannel case when the number of traffic nodes in the network is large.[10]

A TDMA protocol divides the broadcast channel into a series of time slots of identical width. A prespecified number of time slots forms a TDMA frame that continually repeats itself. A demand assignment protocol assumes that when a traffic source has a message to transmit, it must first send a message to a central controller indicating that it wishes to transmit a message to a specified destination address. The central controller assigns specific time slots to each received request on a noninterfering basis. Only one time slot per frame is assigned to each request. Once a time slot is assigned, it remains assigned to the same traffic source for the duration of the message. We therefore assume that each data message consists of a variable number of packets, where the length of a packet is the number of bits per time slot. Voice calls are assigned a dedicated time slot for the duration of the call. (Full-duplex voice traffic actually requires two time slots per frame, one for each direction.) Figure 1 shows an example in which there are four time slots per frame. The numbers in each slot specify the source and destination addresses. The controller has assigned slot 1 to the source-destination pair 1-2. Since the message generated by source 1 requires more than one time slot, source 1 also uses the first time slot in the succeeding frame. The number of time slots per frame and number of bits per time slot are design parameters that can vary from system to system. Notice, however, that an additional constraint might be that the number of frames per second and number of bits

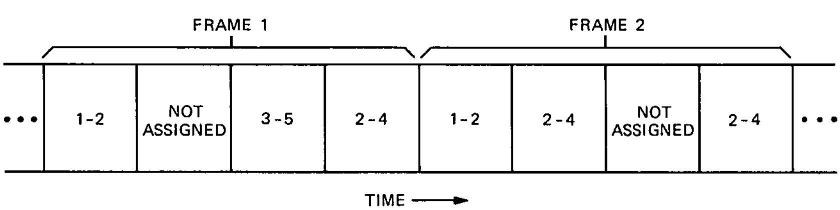


Fig. 1—Assignment of source-destination pairs to time slots.

per time slot must be selected so that each time slot represents a 64-kb/s circuit, which is necessary to provide toll-quality PCM voice transmission.

As the source traffic intensity increases, so does the probability of not being able to assign a new voice or data request because all slots have already been assigned. New voice requests that cannot be assigned are blocked, whereas unassigned data requests enter a queue. As slots become free, requests from the queue are then assigned. The network model analyzed here is different from the models used in Refs. 1 through 7 in one or more of the following respects: (1) Each channel time slot can be assigned to either traffic type. In particular, no moving boundaries[1,6,7] are assumed that partition the time slots in each frame into a section reserved for voice traffic and a section reserved for data traffic. (2) The duration of each message measured in number of time slots, or equivalently, the number of TDMA frames, is an exponentially distributed random variable. Furthermore, the mean voice message length is assumed to be at least an order of magnitude larger than the mean data message length. (3) No priorities are assumed, so that traffic is serviced on a first-come first-served basis.

To simplify the analysis, approximations based upon physical insight are used. Results are expressions for voice blocking probability, mean number of queued data requests, and the mean value of the peaks of the data queue process as functions of the number of time slots and traffic parameters. Comparisons with simulation results indicate that these analytical results are quite accurate.

The next section describes the network queueing model in detail. Section III presents the analytical results, and Section IV presents performance curves and compares them with analogous curves for systems having only one input traffic type.

II. NETWORK QUEUEING MODEL

The TDMA network is modeled as a c-server queueing system, where c is the number of time slots per TDMA frame. We assume voice and data traffic requests to be generated according to two independent Poisson processes with respective arrival rates λ_v and λ_d. We therefore implicitly assume an infinite source model. Service times (or message lengths) in both cases are assumed to be exponentially distributed, with service rates μ_v for voice messages and μ_d for data messages. In particular, the mean number of time slots required for a voice message is $1/\mu_v$. Notice that in practice the service distribution must be discrete, since messages can only last an integral number of time slots. The continuous distribution assumed here is a good approximation to a "discrete" exponential distribution as long as the typical message length is a relatively large number of time slots (i.e., > 10).

Each voice or data arrival demands one time slot per frame for the duration of the message. The rates (i.e., bits per second) at which both voice and data messages access the channel are therefore identical. Voice messages that cannot be assigned a time slot immediately are blocked (i.e., disappear), whereas unassigned data requests enter a queue. To simplify the analysis, we assume that time slot assignments occur instantaneously, rather than at the beginning of the next frame. In particular, as soon as a request is generated, it is immediately assigned, assuming unassigned slots exist. This approximation is reasonable as long as the average message lengths are much greater than one time slot. Given that there are queued data requests, they are assigned as soon as time slots are relinquished by traffic already assigned. Voice messages that arrive while data requests are queued are therefore blocked. As either the voice or data traffic intensity increases, the data traffic tends to grab enough time slots to empty any queue that may appear, thereby depriving voice traffic of available time slots. One therefore expects that under high traffic intensities, voice blocking probability is quite high, whereas the mean data queue length (i.e., number of queued requests) is moderate.

An exact analysis of the model just described can be performed by extracting the associated embedded Markov chain. In this case, the two-dimensional state defining the embedded Markov chain is (d, v), where d and v represent the number of data and voice messages, respectively, in the system. Transition probabilities are easily determined in terms of the traffic parameters and number of time slots, thus a solution for the steady-state probability distribution $p(d, v)$ can be theoretically obtained. If we assume that the data message queue can be arbitrarily large, however, the number of states in the Markov chain becomes infinite. As the dimension of the problem increases, the amount of numerical computation required to obtain $p(d, v)$ increases, which in turn causes further propagation of round-off errors. The exact analysis just outlined was attempted.[11] However, it was not successful due to finite word-length effects. The approximate analysis in the next section is therefore proposed as a simple alternative.

III. ANALYTICAL RESULTS

We start by deriving an approximate expression for voice blocking probability. In steady state, the minimum number of time slots per frame needed to ensure that the system remains stable (i.e., the number of queued data requests does not become infinite with probability one) is

$$\left\lfloor \frac{\lambda_d}{\mu_d} \right\rfloor + 1,$$

where $\lfloor x \rfloor$ denotes the largest integer less than x. If the number of slots is less than this amount, then the system would be unstable even with no additional voice traffic. If the number of slots is greater than this amount, then the system must be stable since unassigned voice requests disappear, and because voice requests cannot preempt queued data requests. All queued data requests must therefore be assigned before any voice messages can be assigned.

The number of time slots per frame available for data messages is a random process that varies according to how many time slots are assigned to voice traffic. Assuming that the mean service time for voice messages ($1/\mu_v$) is orders of magnitude greater than the mean service time for data messages ($1/\mu_d$), the voice "state", i.e., number of voice-occupied slots per frame, varies much more slowly than the data state, which is the total number of data messages present in the system. It is therefore a good approximation to assume that the time spent in each voice state is much longer than the time it takes the number of data requests present in the system to reach steady-state behavior. This "steady-state approximation" is the basis for the analysis that follows. Using this approximation, it follows that if the number of voice messages in the system is greater than or equal to

$$v_0 \equiv c - \left\lfloor \frac{\lambda_d}{\mu_d} \right\rfloor, \tag{1}$$

the normalized data traffic intensity conditioned on the number of voice-occupied slots, $\lambda_d/[(c - v_0)]\mu_d$, is greater than one, and data requests become queued with probability one. Since the number of queued data requests is assumed to reach steady-state behavior, this queue cannot be emptied until a voice message relinquishes a time slot. The steady-state approximation therefore implies that the number of voice messages in the system is never greater than v_0. Computer simulations of the queueing model considered have verified that the probability of the voice state v becoming greater than v_0 is indeed very small when μ_v is much less than μ_d. The "competition" of voice and data traffic for available time slots can therefore be expected to produce intermittent queue "spikes," representing times at which the voice state $v = v_0$. During this time, the data queue process experiences a "transient instability."

Given that v time slots are assigned to voice traffic, the probability that an incoming voice message is blocked is equal to the probability that the number of data requests in the system, d, is greater than or equal to $c - v$. Using the steady-state approximation, this is simply the steady-state probability that a queue exists in an $M/M/c-v$ queueing system and is given by[12]

$$p(d \geq c - v \mid v) = \frac{p_0}{(c-v)!} \left(\frac{\lambda_d}{\mu_d}\right)^{c-v} \frac{1}{1 - (\lambda_d/\mu_d)}, \tag{2}$$

where

$$p_0 = \left[\sum_{k=0}^{c-v-1} \frac{1}{k!} \left(\frac{\lambda_v}{\mu_v}\right)^k + \frac{1}{(c-v)!} \left(\frac{\lambda_v}{\mu_v}\right)^{c-v} \frac{1}{1 - (\lambda_v/\mu_v)} \right]^{-1}. \tag{3}$$

The blocking probability for voice traffic is therefore

$$P_B = \sum_{v=0}^{v_0} p(d \geq c - v \mid v) p(v), \tag{4}$$

where $p(v)$ is the probability that v time slots are assigned to voice traffic.

Consider now a blocking system with v_0 servers and one Poisson input. Given $v < v_0$, let ϕ_v denote the probability that a new arrival can be served (i.e., even though all servers are not busy, a newly arriving request is blocked with probability $1 - \phi_v$). Assuming exponential service times, the probability that v servers are busy is known to be[12]

$$p(v) = \frac{\frac{1}{v!} \left(\frac{\lambda_v}{\mu_v}\right)^v \prod_{j=0}^{v-1} \phi_j}{\sum_{i=0}^{v_0} \left[\frac{1}{i!} \left(\frac{\lambda_v}{\mu_v}\right)^i \prod_{j=0}^{i-1} \phi_j \right]}. \tag{5}$$

This exactly describes the voice-data system considered, where the "entrance" probability,

$$\phi_v = p(d \geq c - v \mid v), \tag{6}$$

and is given by (2). Substituting (2) and (5) into (4) therefore gives the desired result. Notice that because the arrival processes are Poisson, the blocking probability P_B is equal to the probability of being in a blocking state (i.e., all time slots are busy), which is equal to the probability that data requests are queued.

An analogous argument can be applied to compute the mean number of queued data requests. Denoting this queue length as q, we have

$$E(q) = \sum_{v=0}^{v_0} p(v) E(q \mid v), \tag{7}$$

where $E(q \mid v)$ is the mean number of queued data requests given v assigned voice messages. Assuming $\mu_v \ll \mu_d$ implies that $E(q \mid v)$ is approximately equal to the mean queue length for an $M/M/c-v$ queueing system,[12] i.e.,

$$E(q \mid v) \approx \frac{p_0}{c-v}\left(\frac{\lambda_v}{\mu_v}\right)^{c-v} \frac{(c-v)\lambda_v\mu_v}{[(c-v)\mu_v-\lambda_v]^2}, \qquad 0 \le v < v_0, \qquad (8)$$

where p_0 is given by (3). If $v = v_0$, this expression no longer applies, however, since the system becomes unstable. To approximate $E(q \mid v_0)$, note that when $v = v_0$, the mean data queue length increases approximately at rate $\lambda_d - (c - v_0)\mu_d$ until a voice message relinquishes its time slot. At this point the queue starts to empty at rate $(c - v_0 + 1)\mu_d - \lambda_d$. As the queue empties, more time slots may be relinquished by voice messages, causing the queue to empty at a faster rate. Suppose that we assume

$$E(q \mid v_0) \approx \frac{1}{t_0}\int_0^{t_0} E[q(t)]dt, \qquad (9)$$

where t_0 is the duration of the queue spike and $q(t)$, $0 \le t \le t_0$, is the queue length as a function of time (given that v increased from $v_0 - 1$ to v_0 at $t = 0$). Furthermore, we assume that $E[q(t)]$ is piecewise linear (fluid flow approximation).[4] Then it is shown in Appendix A that

$$E(q \mid v_0) \approx \frac{1}{t_0}\left\{\frac{1}{2}\frac{\lambda_d-\mu_d}{(v_0\mu_v)^2} + \frac{\bar{q}_{i_0}^2}{2[(c-v_0+i_0+1)\mu_d-\lambda_d]} + \sum_{j=1}^{i_0}\frac{1}{2(v_0-j)\mu_v}[\bar{q}_{j-1}+\bar{q}_j]\right\}, \qquad (10)$$

where

$$t_0 = \frac{\bar{q}_i}{(c-v_0+i_0+1)\mu_d-\lambda_d} + \sum_{j=0}^{i_0}\frac{1}{(v_0-j)\mu_v}, \qquad (11)$$

$$\bar{q}_i = \frac{\lambda_d-\mu_d}{v_0\mu_v} - \sum_{j=1}^{i}[(c-v_0+j)\mu_d-\lambda_d]\frac{1}{(v_0-j)\mu_v}, \qquad (12)$$

and

$$i_0 = \max\{i \mid \bar{q}_i > 0\}. \qquad (13)$$

Substituting (5), (8), and (10) into (7) therefore gives the approximate mean queue length.

The expressions for voice blocking probability and mean data queue length presented thus far have been found to be quite accurate when compared with simulation results. To gain further insight into the behavior of the system, however, we now attempt to characterize the transient instabilities, or queue spikes, which occur when $v = v_0$.

3.1 *Analysis of transient instabilities*

Figure 2 illustrates the problem under consideration. The queue spikes appear whenever the voice state is equal to v_0. Not shown are queues that occur when the voice state v is less than v_0. The height of each spike is denoted as d_M, the duration of each spike is denoted as τ_w, and the time between spikes is denoted as T. Figure 2 is not meant to indicate a typical sample function for the data message queue process. Significant queues appear when $v < v_0$; however, the purpose of the following analysis is to determine whether the transient instabilities shown in Fig. 2 cause serious performance degradation.

We begin by computing the distribution of the peak value of each spike. Let $p(\tilde{d}, t)$ denote the probability that $\tilde{d}$ data messages are *queued* at time t, given that $v = v_0$. At $t = 0$ we assume $\tilde{d} = 0$. The following equations can be derived in a straightforward manner,[12]

$$\frac{d}{dt} p(\tilde{d}, t) = \lambda_d p(\tilde{d} - 1, t) - [(c - v_0)\mu_d + \lambda_d] p(\tilde{d}\ t) + (c - v_0)\mu_d p(\tilde{d} + 1, t) \quad \text{for} \quad \tilde{d} > 0 \tag{14a}$$

and

$$\frac{d}{dt} p(0, t) = -\lambda_d p(0, t) + (c - v_0)\mu_d p(1, t). \tag{14b}$$

Solving (14) gives the probability that the maximum queue length is equal to $\tilde{d}$ given the time until the first voice departure is t. (Recall that as soon as v decreases from v_0 to $v_0 - 1$, the mean queue length decreases.) We know, however, that the time until the first voice departure is exponentially distributed with parameter $v_0\mu_v$, so that

$$\begin{aligned} q_M(d_M) &\equiv \Pr\{\text{maximum queue length} = d_M\} \\ &= \int_0^\infty v_0\mu_v e^{-v_0\mu_v t} p(d_M, t) dt \\ &= v_0\mu_v Q_M(v_0\mu_v, d_M), \end{aligned} \tag{15}$$

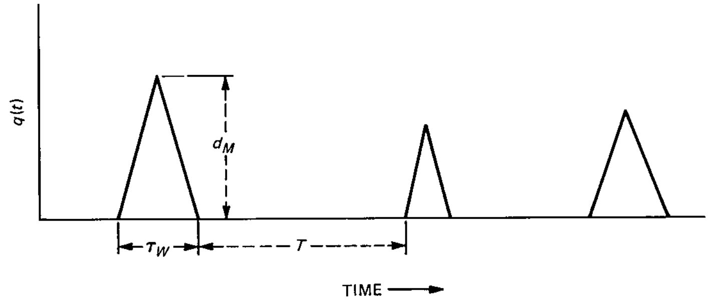


Fig. 2—Transient instabilities in the data queue process.

where

$$Q_M(s, d_M) = \int_0^\infty e^{-st} p(d_M, t)dt \tag{16}$$

is the Laplace transform of $p(d_M, t)$. Equations (14a) and (14b) are standard "birth-death" equations.[12] The Laplace transform, $Q_M(s, \tilde{d})$, can be computed directly from (14) and is given by

$$Q_M(s, d_M) = \frac{r_2^{d_M}(s)}{(c - v_0)\mu_d[r_1(s) - 1]}, \tag{17}$$

where

$$r_1(s) = \frac{1}{2(c - v_0)\mu_d} \{s + \lambda_d + (c - v_0)\mu_d + \sqrt{[s + \lambda_d + (c - v_0)\mu_d]^2 - 4(c - v_0)\mu_d\lambda_d}\} \tag{18a}$$

and

$$r_2(s) = \frac{1}{2(c - v_0)\mu_d} \{s + \lambda_d + (c - v_0)\mu_d - \sqrt{[s + \lambda_d + (c - v_0)\mu_d]^2 - 4(c - v_0)\mu_d\lambda_d}\}. \tag{18b}$$

We therefore have

$$q_M(d_M) = q_M(0) r_2^{d_M}(v_0\mu_v), \tag{19}$$

where

$$q_M(0) = \frac{v_0\mu_v}{(c - v_0)\mu_d[r_1(v_0\mu_v) - 1]}. \tag{20}$$

Since

$$\sum_{d_M=0}^{\infty} q_M(d_M) = 1, \tag{21}$$

it follows that

$$q_M(0) = 1 - r_2(v_0\mu_v). \tag{22}$$

The distribution of the maximum of the queue spike is therefore geometric with parameter $r_2(v_0\mu_v)$. Consequently,

$$E(d_M) = \frac{r_2(v_0\mu_v)}{1 - r_2(v_0\mu_v)} \tag{23}$$

and

$$E[d_M - E(d_M)]^2 = \frac{r_2(v_0\mu_v)}{[1 - r_2(v_0\mu_v)]^2}. \tag{24}$$

Notice that as μ_v decreases relative to μ_d, $E(d_M)$ increases.

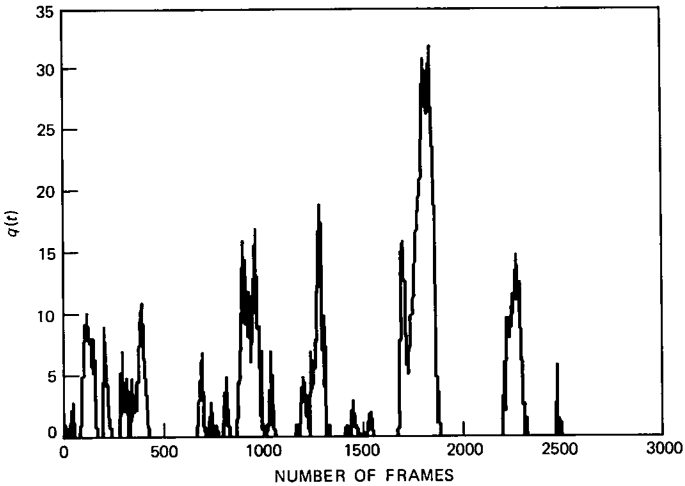


Fig. 3—Sample path for queue process. Every tenth sample is shown.

A sample path of the number of queued data requests versus time is shown in Fig. 3. (Every 10th sample is plotted.) Queues periodically build and empty, which suggests that as an approximation, the mean value of the peaks of these buildups is given by (23). This approximation becomes more accurate as μ_v decreases relative to μ_d. Note, however, that (23) gives the mean value of the peaks of the queue process with initial conditions $v = v_0$ and $\tilde{d} = 0$. It is possible that the mean value of the peaks of the queue process, assuming $v < v_0$, is larger than that predicted by (23). [In some cases, the conditional mean queue length given by (8) for $v = v_0 - 1$ is in fact larger than $E(d_M)$.] A better approximation to the mean value of the peaks of the queue process can be obtained by computing the conditional means assuming $v = 0, 1, \cdots, v_0$, and then using the distribution $p(v)$ given by (5) to form a weighted average. As the present analysis is concerned with evaluating the performance degradation caused by transient instabilites, this computation was not performed.

We now compute the mean duration of the queue spike. Denoting this quantity as $\bar{\tau}_w$, it is apparent that

$$\bar{\tau}_w = \bar{\tau}_{w,1} + \bar{\tau}_{w,2}, \tag{25}$$

where $\bar{\tau}_{w,1}$ is the mean time it takes the queue to reach its maximum value given $v = v_0$, and $\bar{\tau}_{w,2}$ is the mean time it takes the queue to empty. From the previous discussion,

$$\bar{\tau}_{w,1} = \frac{1}{v_0 \mu_v}. \tag{26}$$

Let $\bar{\tau}_{\tilde{d},v}$ denote the mean time it takes to reach state $\tilde{d} = 0$ (i.e., no data message queue) given an initial state of $\tilde{d}$ queued data messages and v voice-occupied time slots. We have the following transition equation,

$$\bar{\tau}_{\tilde{d},v} = \frac{1}{\sigma(v)} + p_v \bar{\tau}_{\tilde{d}-1,v} + q_v \bar{\tau}_{\tilde{d}+1,v} + w_v \bar{\tau}_{\tilde{d},v-1} \tag{27a}$$

with initial condition

$$\bar{\tau}_{0,v} = 0, \tag{27b}$$

where

$$\sigma(v) \equiv v\mu_v + (c - v)\mu_d + \lambda_d \tag{28}$$

and $1/[\sigma(v)]$ is the mean amount of time spent in state $(\tilde{d}, v)$ before a state transition occurs, and

$$p_v = \frac{(c - v)\mu_d}{\sigma(v)}, \tag{29a}$$

$$q_v = \frac{\lambda_d}{\sigma(v)}, \tag{29b}$$

and

$$w_v = \frac{v\mu_v}{\sigma(v)} \tag{29c}$$

are, respectively, the probabilities of going to states $(\tilde{d} - 1, v)$, $(\tilde{d} + 1, v)$, and $(\tilde{d}, v - 1)$ from state $(\tilde{d}, v)$. Equation (27) is a two-dimensional difference equation that is nonlinear in v. Notice, however, that we desire

$$\begin{aligned} \bar{\tau}_{w,2} &= \sum_{\tilde{d}=0}^{\infty} q_M(\tilde{d}) \bar{\tau}_{\tilde{d},v_0-1} \\ &= [1 - r_2(v_0\mu_v)] \sum_{\tilde{d}=0}^{\infty} r_2^{\tilde{d}}(v_0\mu_v) \bar{\tau}_{\tilde{d},v_0-1} \\ &= [1 - r_2(v_0\mu_v)] D\left[\frac{1}{r_2(v_0\mu_v)}, v_0 - 1\right], \end{aligned} \tag{30}$$

where

$$D(z, v) = \sum_{\tilde{d}=0}^{\infty} z^{-\tilde{d}} \bar{\tau}_{\tilde{d},v} \tag{31}$$

is the partial z-transform of $\bar{\tau}_{\tilde{d},v}$. An iterative method for computing $D(1/[r_2(v_0\mu_v)], v_0 - 1)$ is discussed in Appendix B.

To compute the mean time between unstable periods, we first define $\bar{T}_v$ as the expected value of the first passage time it takes to go from an initial voice state $v < v_0$ to voice state v_0. The mean time between unstable periods is then

$$\bar{T} = \sum_{v=0}^{v_0-1} p^*(v)\bar{T}_v, \tag{32}$$

where $p^*(v)$ is the probability that the voice state is v at the end of a queue spike (i.e., when $\tilde{d}$ returns to zero). The computation of $p^*(v)$ is similar to the computation of $\bar{\tau}_w$. In particular, let $p(v \mid v_1, \tilde{d})$ denote the probability of v voice-occupied time slots at the end of an unstable period given an initial state $(v_1, \tilde{d})$. The following transition equation is easily obtained,

$$p(v \mid v_1, \tilde{d}) = q_v p(v \mid v_1, \tilde{d} + 1) + p_v p(v \mid v_1, \tilde{d} - 1) + w_v p(v \mid v_1 - 1, \tilde{d}), \tag{33a}$$

where q_v, p_v, and w_v are defined by (29). The initial conditions are

$$p(v \mid v - 1, \tilde{d}) = 0 \tag{33b}$$

and

$$p(v \mid v_1, 0) = \begin{cases} 1 & \text{if } v = v_1 \\ 0 & \text{otherwise.} \end{cases} \tag{33c}$$

In analogy with (27), (33) is a two-dimensional difference equation that is nonlinear in v. As before, we desire

$$\begin{aligned} p^*(v) &= [1 - r_2(v_0)] \sum_{\tilde{d}=0}^{\infty} r_2^{\tilde{d}}(v_0\mu_v) p(v \mid v_0 - 1, \tilde{d}) \\ &= [1 - r_2(v_0)] D\left[\frac{1}{r_2(v_0\mu_v)}, v \mid v_0 - 1\right], \end{aligned} \tag{34}$$

where

$$D(z, v \mid v_1) = \sum_{\tilde{d}=0}^{\infty} z^{-\tilde{d}} p(v \mid v_1, \tilde{d}). \tag{35}$$

An iterative method for computing $D(z, v \mid v_1)$ is discussed in Appendix B.

The mean time between unstable periods, given the initial starting state, can be approximated by again assuming voice traffic service times are very long relative to data traffic service times. For each voice state, we assume that the data traffic exhibits steady-state behavior. This leads to the following difference equation,

$$\bar{T}_v = t_v + r_v \bar{T}_{v-1} + s_v \bar{T}_{v+1}, \tag{36}$$

where

$$t_v = \frac{1}{\lambda_v \phi_v + v\mu_v} \tag{37a}$$

is the mean time spent in voice state v before going to state $v - 1$ or $v + 1$, ϕ_v is the "entrance" probability for an incoming voice message and is equal to the probability that a data message queue exists, and

$$r_v = \frac{v\mu_v}{\lambda_v \phi_v + v\mu_v} \tag{37b}$$

and

$$s_v = \frac{\lambda_v \phi_v}{\lambda_v \phi_v + v\mu_v} \tag{37c}$$

are, respectively, the probabilities of going from voice state v to state $v - 1$ and from voice state v to state $v + 1$. In Appendix C we show

$$\bar{T}_v = \frac{1}{\lambda_v} \sum_{j=v}^{v_0-1} \left\{ \left(\frac{\mu_v}{\lambda_v}\right)^j \frac{j!}{\gamma_j} \left[1 + \sum_{m=0}^{j-1} \left(\frac{\mu_v}{\lambda_v}\right)^{m-j} \frac{\gamma^{j-m-1}}{(j-m)!} \right] \right\}, \tag{38}$$

where

$$\gamma_j = \prod_{m=0}^{j} \phi_m. \tag{39}$$

Therefore, computation of $\bar{T}_v$ and $p^*(v)$ by way of (38) and the method given in Appendix B, respectively, yields the mean time between unstable periods. The relative frequency, or probability, that the system is in an unstable state ($v = v_0$) is approximated by

$$p_u = \frac{\bar{\tau}_w}{\bar{\tau}_w + \bar{T}}, \tag{40}$$

where $\bar{\tau}_w$ and $\bar{T}$ are given by (25) and (32), respectively. Notice that this expression should be approximately equal to the value of $p(v_0)$ obtained using (5).

This completes the presentation of analytical results. These results are used in the next section to evaluate the performance of an integrated voice-data TDMA network, and to demonstrate the improvement over analogous TDMA networks that handle only one traffic type.

IV. PERFORMANCE RESULTS

The objective of this section is to demonstrate how the integrated voice-data network described in Sections I and II performs as a function of (1) input traffic intensity, (2) traffic blend (ratio of voice traffic intensity to total traffic intensity), and (3) system size, as measured by the number of time slots. In all cases, half-duplex voice traffic is assumed. We expect the full-duplex case, where each voice message requests two time slots, to exhibit similar behavior. Denoting the voice traffic intensity as $\rho_v = \lambda_v/(c\mu_v)$, and the data traffic intensity as $\rho_d = \lambda_d/(c\mu_d)$, where c is the number of time slots, the total normalized offered load is defined as

$$\rho = \rho_v + \rho_d, \tag{41}$$

and the traffic blend is

$$r = \frac{\rho_v}{\rho_v + \rho_d}. \tag{42}$$

Initial traffic parameters are selected to produce a preselected value of r, and the input traffic intensity ρ is varied between zero and one by multiplying both λ_v and λ_d by a constant. The service rates used are $\mu_v = 0.001$ and $\mu_d = 0.025$, which corresponds to a mean voice message length of 1000 time slots and a mean data message length of 40 time slots. In practice, the mean voice message is much greater than 1000 time slots; however, the analytical results in the last section become more accurate as μ_v decreases relative to μ_d.

Figures 4a, 4b, and 4c show voice message blocking probabilities computed by means of (4) versus the offered load for systems with 20, 100, and 500 time slots, respectively. In each case, curves are shown for three different traffic blends. A few randomly selected points from Figs. 4a, 4b, and 4c were compared with results obtained by a computer simulation of the queueing model under consideration. In each case the approximate result and the computer-simulated result were nearly identical. Also shown are plots of blocking probabilities versus load produced by systems that handle only (half-duplex) voice traffic and which have rc time slots, where c is the number of time slots in the voice-data network. These curves are computed directly from (5), where $\phi_j = 1$ for $j < rc$.

At high traffic intensities, blocking probabilities for the integrated systems are consistently higher than those for the analogous single traffic systems. In this region, data queues often form, causing data messages to grab time slots relinquished by voice messages. In contrast, at lower traffic intensities, data message queues are less likely, so that voice messages often have access to additional time slots. At low traffic intensities, integrated systems therefore always exhibit superior performance when compared with analogous voice-only systems. In each comparison, there appears to be a unique traffic intensity, ρ^*, where both systems give the same blocking probability. Notice that as the traffic blend r decreases, ρ^* increases. This is due to the fact that the probability of a data message queue existing at a fixed, normalized traffic intensity decreases as the number of time slots allocated for

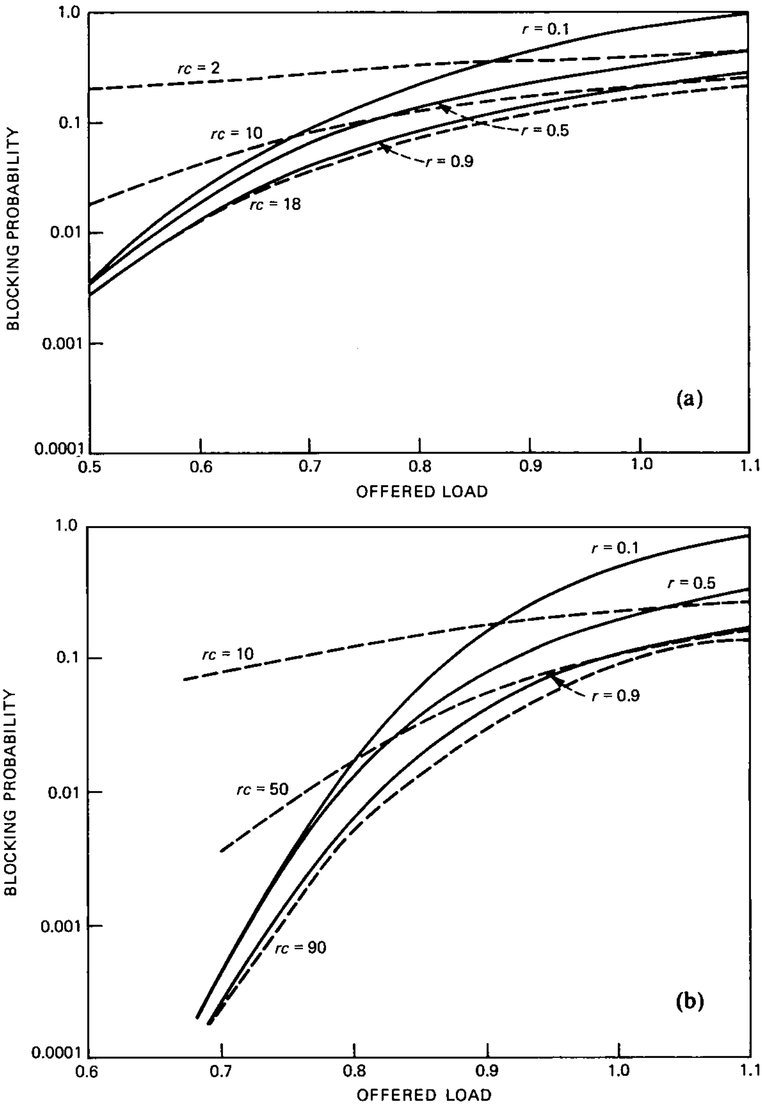


Fig. 4—Voice message blocking probability versus offered load for integrated systems with: (a) 20 slots and single traffic systems with 2, 10, and 18 slots; (b) 100 slots and single traffic systems with 10, 50, and 90 slots.

data traffic increases. As r decreases, voice messages therefore often have access to additional time slots not used by data traffic. At a fixed traffic intensity, as r decreases, the blocking probability produced by the integrated system should therefore decrease, relative to the blocking probability produced by the analogous voice-only system. A final observation is that as the traffic intensity decreases, blocking proba-

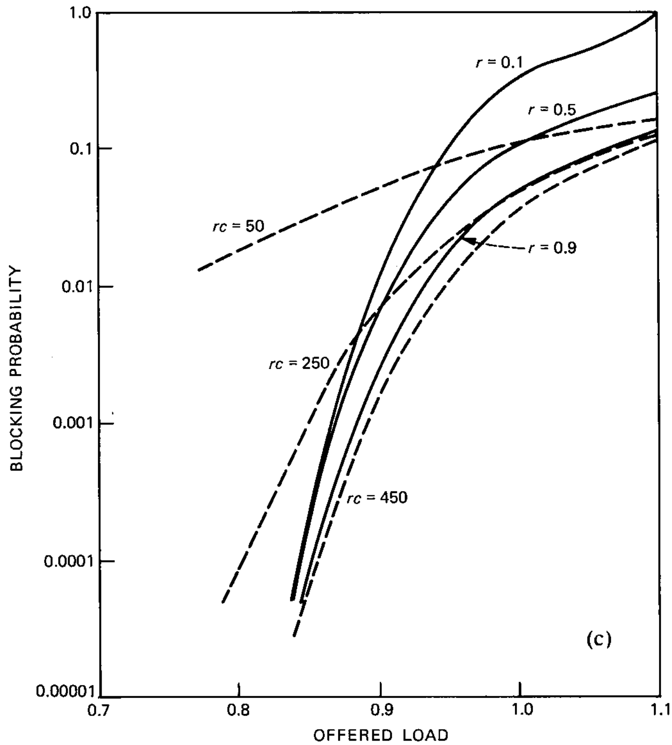


Fig. 4(c)—Voice message blocking probability versus offered load for integrated systems with 500 slots and single traffic systems with 50, 250, and 450 slots.

bilities obtained using the integrated systems become insensitive to the traffic blend. This is in contrast to the analogous voice-only systems, which result in a much wider variation in blocking probabilities as the number of time slots is varied.

Figures 5a, 5b, and 5c show plots of mean data message queue length (number of queued data requests), computed by means of (7), (8), and (10), versus normalized offered load for systems with 20, 100, and 500 time slots, respectively. Curves are again shown for three different traffic blends. Also plotted is the mean number of queued data requests produced by a system handling data traffic only with c time slots. Close agreement was again found between randomly selected points from these figures and computer simulation results. At high traffic intensities, the variation between curves is caused by the different traffic loads, at which the mean data queue length approaches infinity. In particular, the single traffic curve has its asymptote at $\rho = \rho_d = 1$. In contrast, because queued data messages can grab relinquished voice-occupied time slots, the integrated traffic curves have asymptotes at $\rho_d = 1$, which corresponds to $\rho = 1.1$, $\rho = 2$, and $\rho = 10$ for $r = 0.1$,

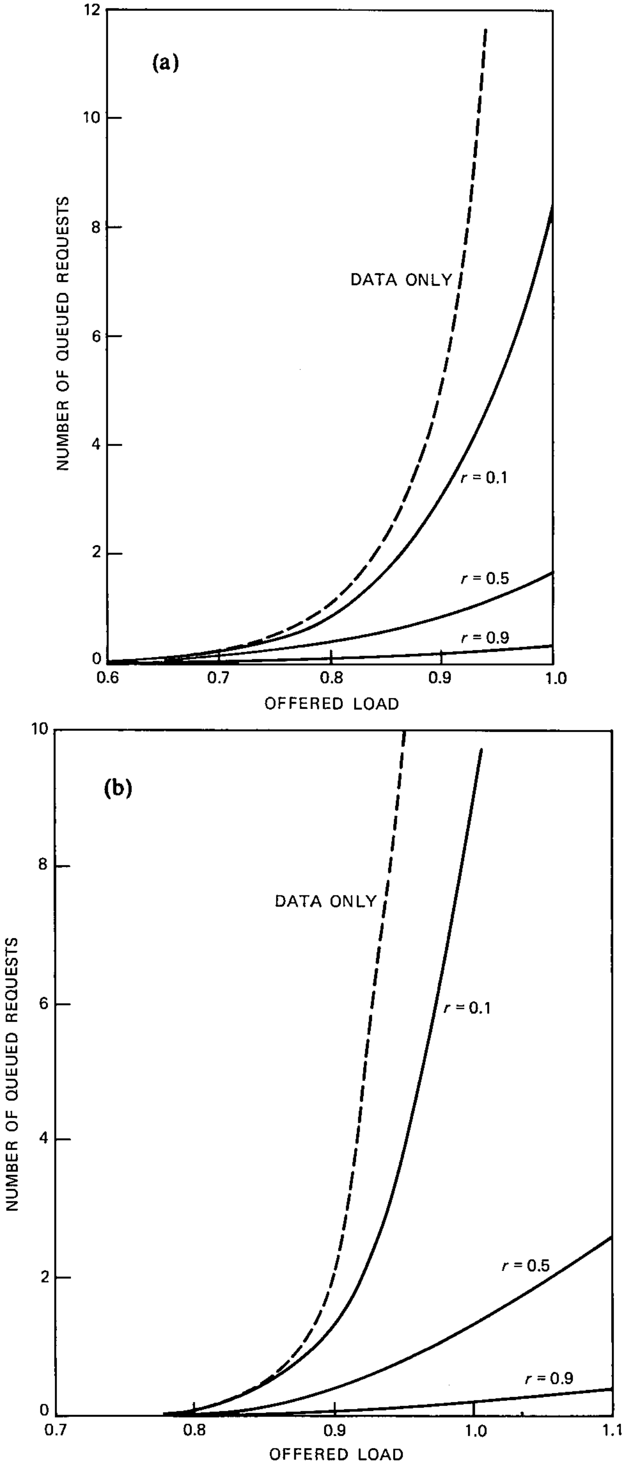


Fig. 5—Mean data message queue length versus offered load for integrated systems and a single traffic system with: (a) 20 slots, and (b) 100 slots.

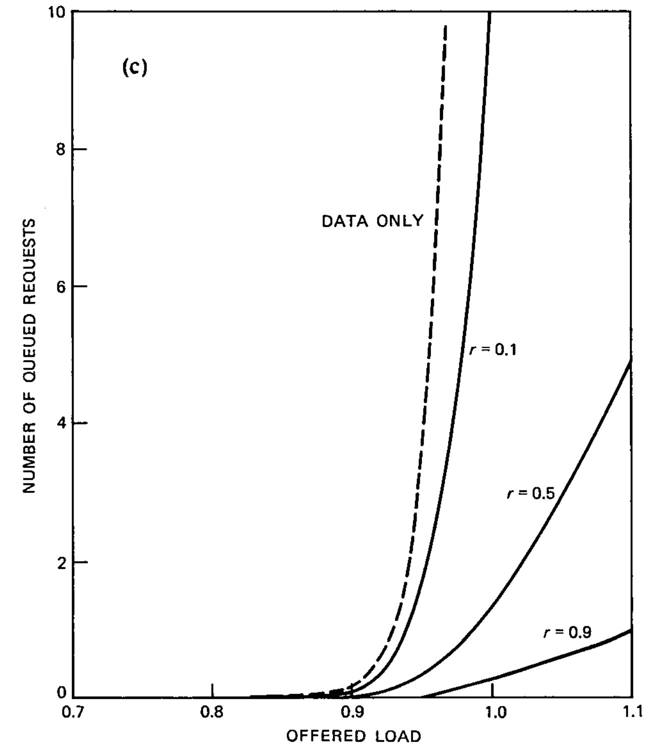
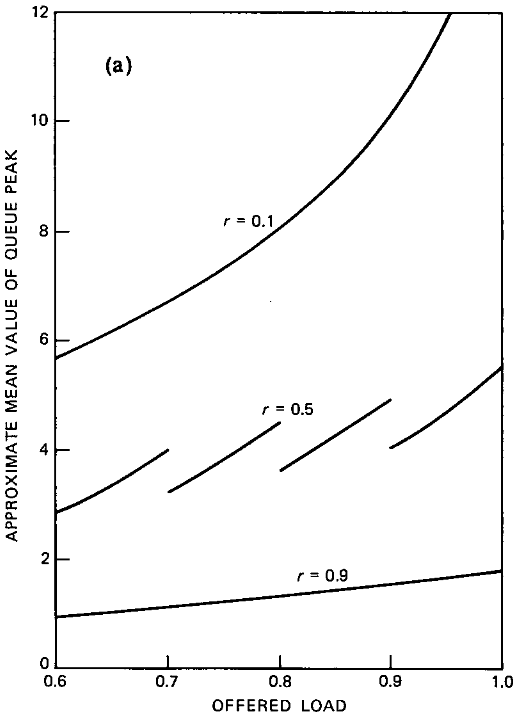


Fig. 5(c)—Mean data message queue length versus offered load for integrated systems and a single traffic system with 500 slots.

$r = 0.5$, and $r = 0.9$, respectively. Mean queue length in these high traffic intensity regions is not of interest, however, since the corresponding voice blocking probability is near unity.

Figures 6a, 6b, and 6c show plots of $E(d_M)$ given by (23) versus offered load, which indicates the mean value of the maximum of queue buildups. The discontinuities in Fig. 6a are caused by discontinuous changes in the voice instability state v_0 as a function of traffic load. As an example, for the case $c = 20$ and $r = 0.5$ shown in Fig. 6a, v_0 changes from 14 to 13 as the traffic intensity ρ increases from $0.7 - \epsilon$ to 0.7, where ϵ is small. Discontinuities were observed in all curves shown in Fig. 6; however, in most cases these discontinuities were hardly noticeable. In particular, as the number of slots c increases, $E(d_m)$ becomes less sensitive to changes in v_0. A comparison of the results in Figs. 6a, 6b, and 6c with simulated sample paths of the data-message queue process indicates that the results presented here are typically about 10 to 25 percent smaller than the actual peaks observed, indicating that the peaks that occur when $v < v_0$ are often greater than those that occur when $v = v_0$. As μ_v decreases relative to μ_d,



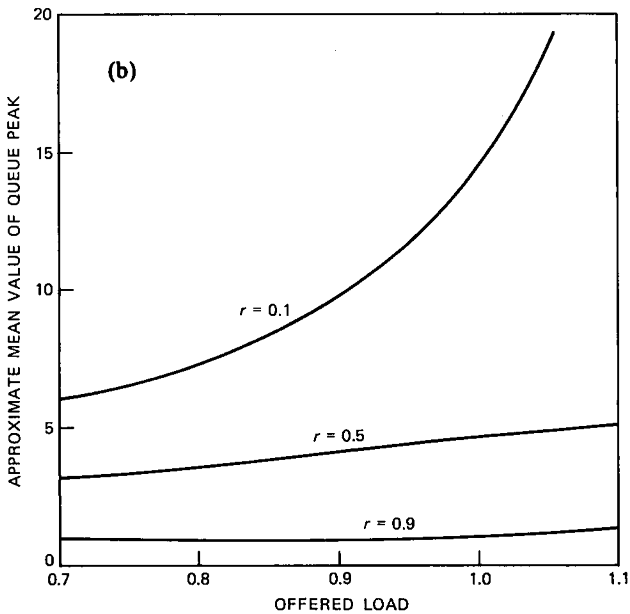


Fig. 6—Approximate mean value of queue peaks versus offered load from eq. (23) for ntegrated systems with: (a) 20 slots and (b) 100 slots.

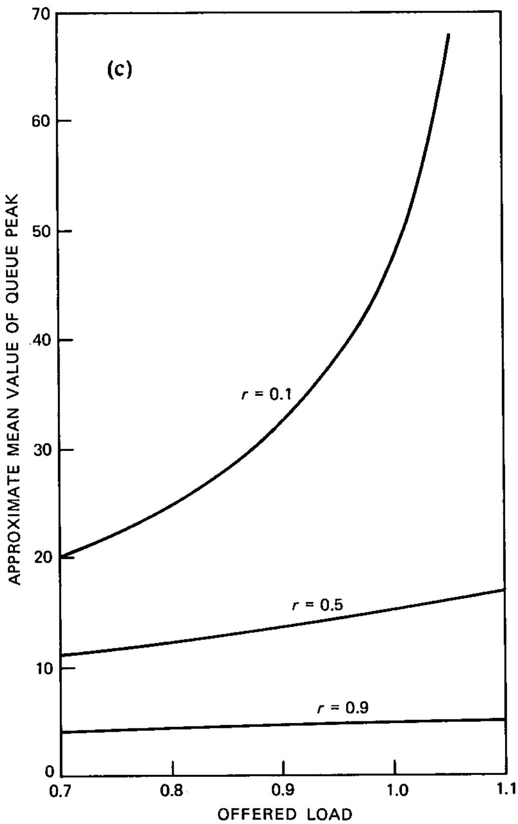


Fig. 6(c)—Approximate mean value of queue peaks versus offered load from eq. (23) for integrated systems with 500 slots.

however, $E(d_M)$ given by (23) should give a more accurate indication of the mean value of these peaks.

For each point computed in Figs. 4 through 6, the probability of being in an "unstable" state $p(v_0)$, the mean duration of the unstable state $(\bar{\tau}_w)$, and the mean time between unstable states $(\bar{T})$ were calculated by means of the results in Section 3.1. A few representative points are listed in Table I. The probability of being in an unstable state and the resulting queues that form are typically too small to cause significant degradation in system performance.

V. CONCLUSIONS

Results obtained from the analysis of an integrated voice-data TDMA protocol indicate that voice message blocking probability in the integrated system is insensitive to the blend of traffic at low input traffic intensities. As the traffic intensity increases, the blocking

Table I—Mean duration of unstable state ($\bar{\tau}_w$ in number of frames), mean time between unstable states ($\bar{T}$ in number of frames), and the probability of being in an unstable state [$p(v_0)$] for a few representative cases (assuming $r = 0.5$)

Time Slots	Load (ρ)	$\bar{\tau}_w$	$\bar{T}$	$p(v_0)$
20	0.6	129.2	$2.05 \cdot 10^5$	$6.3 \cdot 10^{-4}$
20	1.0	191.8	2898	0.062
100	0.7	44.8	$1.85 \cdot 10^9$	$2.4 \cdot 10^{-8}$
100	1.0	60.8	$4.15 \cdot 10^4$	0.0015
500	0.85	$6.66 \cdot 10^6$	$4.12 \cdot 10^{19}$	$1.6 \cdot 10^{-13}$
500	1.0	$2.88 \cdot 10^7$	$5.87 \cdot 10^{13}$	$4.9 \cdot 10^{-7}$

probability of the integrated system increases, relative to the blocking probability of the analogous voice-only system. The traffic intensity at which the two blocking probability curves intersect is a function of the traffic blend. Mean queue length in the integrated system displays a wide variation with traffic blend at high traffic intensities, due to the variation in traffic intensity at which instability occurs. At offered loads of 0.7 to 0.8, very good performance can be achieved (i.e., a blocking probability <0.01 and a mean queue length near zero) with moderately sized systems (~100 slots/frame). Finally, the data message queues that form during the unstable transients are moderate for the cases examined, and the frequency at which these transients occur is in most cases quite small. As the number of time slots per TDMA frame increases, results presented here show significant improvements in system performance. This is an important observation since most practical networks are much larger than those considered here (i.e., greater than 1000 time slots per frame).

VI. ACKNOWLEDGMENT

The author thanks S. M. Barta and B. E. Simon for many helpful conversations concerning this work.

REFERENCES

1. M. J. Fischer 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.
2. C. J. Weinstein, M. L. Malpass, and M. J. Fisher, "Data Traffic Performance of an Integrated Circuit- and Packet-Switched Multiplex Structure," IEEE Trans. Commun., *COM-28* (June 1980), pp. 873–8.
3. M. Schwartz and B. Kraimeche, "Comparison of Channel Assignment Techniques for Hybrid Switching," Proc. 1982 IEEE ICC, Philadelphia, PA, June 1982, pp. 2F.3.1–5.
4. D. P. Gaver and J. P. Lehoczky, "Channels That Cooperatively Service a Data Stream and Voice Messages," IEEE Trans. Commun., *COM-30* (May 1982), pp. 1153–62.
5. M. J. Ross and O. A. Mowafi, "Performance Analysis of Hybrid Switching Concepts for Integrated Voice/Data Communications," IEEE Trans. Commun., *COM-30* (May 1982), pp. 1073–87.

6. N. Janakiraman, B. Pagurek, and J. E. Neilson, "Performance Analysis of an Integrated Switch with Fixed or Variable Frame Rate and Movable Voice/Data Boundary," IEEE Trans. Commun., *COM-32,* No. 1 (January 1984), pp. 34–9.
7. A. G. Konheim and R. L. Pickholtz, "Analysis of Integrated Voice/Data Multiplexing," IEEE Trans. Commun., *COM-32,* No. 2 (February 1984), pp. 140–7.
8. R. Cooperman and W. G. Schmidt, "A Satellite Switched SDMA/TDMA System for Wideband Multibeam Satellites," ICC Conf. Rec., Seattle, Washington, June 1973.
9. D. O. Reudink and Y. S. Yeh, "A Scanning Spot Beam Satellite System," B.S.T.J., *56,* No. 8 (October 1977), pp. 1549–60.
10. S. M. Barta and M. L. Honig, "Analysis of a Demand Assignment TDMA Blocking System," AT&T Bell Lab. Tech. J., *63,* No. 1 (January 1984), pp. 89–114.
11. B. E. Simon, unpublished work.
12. T. L. Saaty, *Elements of Queuing Theory with Application,* New York: McGraw-Hill, 1961.

APPENDIX A

We wish to show (10) using the approximation (9). In addition, we assume that $E[q(t)]$ is piecewise linear. (This would be true, for instance, if the data message queue length $q(t)$ were allowed to assume negative values.) At time $t = 0$ we assume that the state of the system is $(v_0,\ c - v_0)$. Let $t_v(i)$ denote the mean time it takes $i + 1$ voice messages to relinquish their time slots. Then

$$t_v(i) = \sum_{j=0}^{i} \frac{1}{(v_0 - j)\mu_v}, \tag{43}$$

and

$$E[q(t)] \approx \begin{pmatrix} q_M v_0 \mu_v t & 0<t<t_v(0) \\ E\{q[t_v(i)]\} - [(c - v_0 + i + 1)\mu_d - \lambda_d][t - t_v(i)], & t_v(i)<t<t_v(i+1) \end{pmatrix}, \tag{44}$$

where q_M is the peak value of $E[q(t)]$ and is given by

$$q_M = \frac{\lambda_d - (c - v_0)\mu_d}{v_0 \mu_v}. \tag{45}$$

Notice that for $t > t_v(0)$,

$$\begin{aligned} E\{q[t_v(i)]\} \equiv \bar{q}_i &= \bar{q}_{i-1} \\ &\quad - [(c - v_0 + i)\mu_d - \lambda_d][t_v(i) - t_v(i-1)] \\ &= q_M - \sum_{j=1}^{i} [(c - v_0 + j)\mu_d - \lambda_d] \frac{1}{(v_0 - j)\mu_v}. \end{aligned} \tag{46}$$

To calculate the area under $E[q(t)]$, we must first compute

$$t_0 \equiv \inf\{t \mid E[q(t)] = 0 \quad \text{and} \quad t > 0\}. \tag{47}$$

Letting

$$i_0 = \max\{i \mid \bar{q}_i > 0\}, \tag{48}$$

it follows that

$$t_v(i_0) < t_0 < t_v(i_0 + 1). \tag{49}$$

For $t_v(i_0) \le t \le t_0$,

$$E[q(t)] = \bar{q}_{i_0} - [(c - v_0 + i_0 + 1)\mu_d - \lambda_d][t - t_v(i_0)], \tag{50}$$

and setting $\bar{q}_{i_0} = 0$ gives

$$t_0 = \frac{\bar{q}_{i_0}}{(c - v_0 + i_0 + 1)\mu_d - \lambda_d} + t_v(i_0). \tag{51}$$

A plot of $E[q(t)]$, $0 \le t \le t_0$, assuming three voice departures ($i_0 = 2$) is shown in Fig. 7. If we use (9), it follows that

$$E(q \mid v_0) \approx \frac{1}{t_0} \sum_{j=0}^{i_0+1} A_j, \tag{52}$$

where A_j is the area of region R_j. It is apparent that

$$A_0 = \frac{1}{2} \frac{q_M}{t_v(0)} = \frac{1}{2} \frac{\lambda_d - (c - v_0)\mu_d}{(v_0 \mu_v)^2} \tag{53}$$

and that

$$A_{i_0+1} = \frac{1}{2} [t_0 - t_v(i_0)]\bar{q}_{i_0}$$
$$= \frac{\bar{q}_{i_0}^2}{2[(c - v_0 + i_0 + 1)\mu_d - \lambda_d]}. \tag{54}$$

Finally, regions $R_1, \cdots, R_{i_0}$ are trapezoids with upper-boundary $E[q(t)]$, so that for $1 \le j \le i_0$,

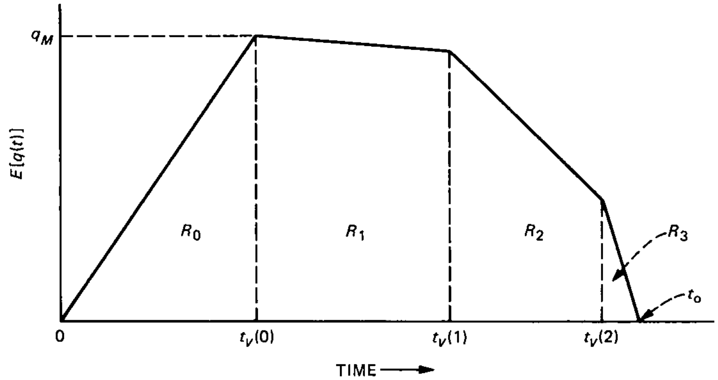


Fig. 7—Mean queue length versus time during transient instability.

$$A_j = \frac{1}{2}[t_v(j) - t_v(j-1)](\bar{q}_{j-1} + \bar{q}_j)$$
$$= \frac{1}{2(v_0 - j)\mu_v}(\bar{q}_{j-1} + \bar{q}_j). \tag{55}$$

Substituting (51) and (53) through (55) into (52), and using (43) and (46) gives (10).

APPENDIX B

We are interested in computing $D\{1/[r_2(v_0\mu_v)], v_0 - 1\}$ and $D\{1/[r_2(v_0\mu_v)], v \mid v_1\}$, where $D(z, v)$ and $D(z, v \mid v_1)$ are given, respectively, by (31) and (35) and $r_2(v_0\mu_v)$ is given by (18b). Multiplying both sides of (27a) by $z^{-\tilde{d}}$ and summing from $\tilde{d} = 1$ to infinity gives, after algebraic manipulation,

$$D(z, v) = \frac{(z-1)[v\mu_v D(z, v-1) - \lambda_d \bar{\tau}_{1,v}] + 1}{z(1-z)s(v, z^{-1})}, \tag{56}$$

where

$$s(v, z) = \mu_d(c - v)z^2 - [(c-v)\mu_d + \lambda_d + v\mu_v]z + \lambda_d, \tag{57}$$

which has real roots

$$r_k(v) = \frac{1}{2(c-v)\mu_d}\{\sigma(v) \pm \sqrt{\sigma^2(v) - 4(c-v)\mu_d\lambda_d}\}, \tag{58}$$

where $k = 1$ corresponds to "+", $k = 2$ corresponds to "−", and $\sigma(v)$ is given by (28). Notice that the roots $r_k(v_0)$ given by (58) are identical to the roots $r_k(v_0\mu_v)$ given by (18). For convenience, we therefore refer to the roots $r_k(v_0\mu_v)$ as $r_k(v_0)$, for $k = 1, 2$. From Rouche's Theorem,[12] it follows that $r_2(v) < 1$ and $r_1(v) \geq 1$. The z-transform, $D(z, v)$, must be analytic outside the unit circle, and hence $\bar{\tau}_{1,v}$ in (56) must be selected to cancel the pole at $1/[r_2(v)]$. In particular, $s(v, z^{-1})$ has roots $1/[r_1(v)] < 1$ and $1/[r_2(v)] > 1$, so that

$$\bar{\tau}_{1,v} = \frac{v\mu_v}{\lambda_d} D\left[\frac{1}{r_2(v)}, v - 1\right] + \frac{r_2(v)}{[1 - r_2(v)]\lambda_d}. \tag{59}$$

As an example, suppose that we assume the data message queue empties with probability one before $k + 1$ voice messages relinquish their time slots. Once the voice state becomes $v = v_0 - k - 1$, the voice service rate $\mu_v = 0$. Substituting these values for μ_v and v in (56) gives

$$D(z, v_0 - k - 1)$$
$$= \frac{z[1 - \lambda_d(z-1)\bar{\tau}_{1,v_0-k-1}]}{(z-1)[1 - r\,(v\ \ - k - 1)z][1 - r\,(v\ \ - k - 1)z]}, \tag{60}$$

where from (58),

$$r_1(v_0 - k - 1) = 1 \quad \text{and} \quad r_2(v_0 - k - 1) = \frac{\lambda_d}{(c - v_0 + k + 1)\mu_d}. \tag{61}$$

Selecting $\bar{\tau}_{1,v_0-k-1}$ to cancel the pole at $1/[r_2(v_0 - k - 1)]$ and simplifying gives

$$D(z, v_0 - k - 1) = \frac{z}{(z-1)^2} \frac{1}{\mu_d(c - v_0 + k + 1) - \lambda_d}, \tag{62}$$

which has inverse transform

$$\bar{\tau}_{\tilde{d},v_0-k-1} = \frac{\tilde{d}}{\mu_d(c - v_0 + k + 1) - \lambda_d}. \tag{63}$$

Equation (62) constitutes an initial condition for (56), which can be iterated numerically using (59). In particular, assuming no more than k voice messages can relinquish their time slots after the queue begins to empty, $D(z, v_0 - k - 1)$ for $z = 1/[r_2(v_0)]$ and $z = 1/[r_2(v_0 - k)]$ is calculated from (62). The value of $\bar{\tau}_{1,v_0-k}$ is subsequently computed from (59) and is used in (56) to compute $D(z, v_0 - k)$ for $z = 1/[r_2(v_0)]$ and $z = 1/[r_2(v_0 - k + 1)]$. Equation (59) is subsequently used to compute $\bar{\tau}_{1,v_0-k+1}$, which is used to compute $D(z, v_0 - k + 1)$, and so forth until $D\{1/[r_2(v_0)], v_0 - 1\}$ is computed.

To compute $D(z, v \mid v_1)$ given by (35) at $z = 1/[r_2(v_0)]$, we multiply both sides of (33a) by $z^{-\tilde{d}}$ and sum from $\tilde{d} = 1$ to infinity to get

$$D(z, v \mid v_1 + 1) = \frac{-v\mu_v D(z, v \mid v_1) + \lambda_d p(v \mid v_1 + 1, 1) + v\mu_v \delta_{v,v_1} - [\sigma(v) - \lambda_d z]\delta_{v,v_1+1}}{zs(v, z^{-1})}, \tag{64}$$

where δ_{ij} is the Kronecker delta. Using the condition (33b) gives the boundary condition

$$D(z, v_1 \mid v_1 - 1) = 0. \tag{65}$$

Because $D(z, v \mid v_1)$ must be analytic outside the unit circle, $p(v \mid v_1, 1)$ is selected to cancel the pole at $z = 1/[r_2(v_0)]$. This implies that

$$p(v \mid v_1, 1) = \frac{v\mu_v}{\lambda_d} \left\{ D\left[\frac{1}{r_2(v)}, v \mid v_1 - 1\right] - \delta_{v,v_1-1} \right\} + \frac{\sigma(v) r_2(v) - \lambda_d}{\lambda_d r_2(v)} \delta_{v,v_1}. \tag{66}$$

To compute $D(z, v \mid v_0 - 1)$ at $z = 1/[r_2(v_0)]$ for $v = v_0 - 1, v_0 - 2, \cdots, v_0 - k - 1$, where k is the maximum number of voice departures allowed, the boundary condition (65) is first used in (66) to get

$$p(v_0 - j \mid v_0 - j, 1) = \frac{\sigma(v_0 - j) r_2(v_0 - j) - \lambda_d}{\lambda_d r_2(v_0 - j)}, \tag{67}$$

which is used in (64) to compute

$$D(z, v_0 - j \mid v_0 - j) = \frac{\lambda_d p(v_0 - j \mid v_0 - j, 1) - [\sigma(v_0 - j) - \lambda_d z]}{z s(v_0 - j, z^{-1})},$$

where j is initially one and ranges from one to $k + 1$. This expression is evaluated at $z = 1/[r_2(v_0 - j)]$ and substituted into (66) to obtain $p(v_0 - j \mid v_0 - j + 1, 1)$, which is used in (64) to compute $D(z, v_0 - j \mid v_0 - j + 1)$ at the appropriate values of z. This procedure continues until the value of $D\{1/[r_2(v_0)], v_0 - j \mid v_0 - 1\}$ is obtained, whereupon j is incremented and the procedure starts over again. In this way the values $D\{1/[r_2(v_0)], v_0 - j \mid v_0 - 1\}$ for $j = 1, 2, \cdots, k + 1$ are generated systematically.

APPENDIX C

We wish to show that (38) is the solution to (36). We first rewrite (36) as

$$\bar{T}_{v+1} = \left(1 + \frac{v\mu_v}{\lambda_v \phi_v}\right) \bar{T}_v - \frac{v\mu_v}{\lambda_v \phi_v} \bar{T}_{v-1} - \frac{1}{\lambda_v \phi_v}. \tag{68}$$

Letting

$$y_v = \bar{T}_v - \bar{T}_{v-1}, \tag{69}$$

(68) can be rewritten as

$$y_{v+1} = \frac{v\mu_v}{\lambda_v \phi_v} y_v - \frac{1}{\lambda_v \phi_v}, \quad \text{for} \quad 0 \le v \le v_0, \tag{70}$$

which can be iterated to give

$$y_{v+1} = \left(\frac{\mu_v}{\lambda_v}\right)^{k+1} \frac{v!\gamma_{v-k-1}}{(v - k - 1)!\gamma_v} y_{v-k} - \frac{1}{\lambda_v} \sum_{m=0}^{k} \left(\frac{\mu_v}{\lambda_v}\right)^m \frac{v!\gamma_{v-m}}{(v - m)!\gamma_v}, \tag{71}$$

where γ_v is given by (39). Using the initial condition,

$$y_1 = \bar{T}_1 - \bar{T}_0 = -\frac{1}{\lambda_v \phi_0}, \tag{72}$$

and substituting $k = v - 1$ in (71) gives

$$y_{v+1} = -\frac{1}{\lambda_v} \left(\frac{\mu_v}{\lambda_v}\right)^v \frac{v!}{\gamma_v} \left[1 + \sum_{m=0}^{v-1} \left(\frac{\mu_v}{\lambda_v}\right)^{m-v} \frac{\gamma_{v-m-1}}{(v - m)!}\right]. \tag{73}$$

From (69),

$$\bar{T}_v = \sum_{j=1}^{v} y_j + \bar{T}_0, \tag{74}$$

which has boundary condition

$$\bar{T}_{v_0} = \sum_{j=1}^{v_0} y_j + \bar{T}_0 = 0. \tag{75}$$

This implies that

$$\bar{T}_0 = -\sum_{j=1}^{v_0} y_j, \tag{76}$$

so that

$$\bar{T}_v = -\sum_{j=v+1}^{v_0} y_j. \tag{77}$$

Combining (73) and (77) gives (38).

AUTHOR

Michael L. Honig, B.S. (Electrical Engineering), 1977, Stanford University; M.S. and Ph.D. (Electrical Engineering), 1978 and 1981, respectively, University of California, Berkeley; Bell Laboratories, 1981–1982; AT&T Information Systems, 1983. Present affiliation Bell Communications Research, Inc. At AT&T Bell Laboratories and AT&T Information Systems Mr. Honig worked on modulation, coding, and echo cancellation of voiceband data signals, performance analysis of local area networks, and office information systems. He is now a member of the Communications Principles Research group in the Applied Research Area of Bell Communications Research, Inc. Member, IEEE, Tau Beta Pi, Phi Beta Kappa.