AT&T BELL LABORATORIES TECHNICAL JOURNAL
Vol. 63, No. 2, February 1984
Printed in U.S.A.

Integrated Voice/Data Services on Fasnet

By J. W. MARK* and J. O. LIMB†

(Manuscript received July 8, 1983)

Fasnet is a high-speed local area network with a pair of unidirectional communication channels. While individual stations control their own access to the medium, global scheduling of the medium for various types of traffic is controlled centrally. Centralized scheduling permits a slot to be endowed with a type, so it is feasible to integrate synchronous and asynchronous traffic services on Fasnet. In this paper we describe a policy for admitting new voice calls into the system and an approximate analysis of the queueing behaviour of data packets in an integrated voice and data services environment. Queueing behaviour can also be studied by means of simulation, which is, however, an expensive undertaking, particularly at high transmission rates, for example, at 100 Mb/s. The approximate method described in this paper allows the performance evaluation for high-speed local area networks.

I. INTRODUCTION

Packet-switching technology has enhanced the potential for complete integration of different types of traffic on a common transmission facility, more particularly within a local environment where signal propagation delays are relatively small. Over the past decade a number of papers have been published on the integration of voice and data.[1-8]

We will address the problem of integrated voice and data services using a specific local area network (LAN), Fasnet,[9] as the backbone support. The techniques explored and evaluated here would carry over

* University of Waterloo, Waterloo, Ontario, Canada. † AT&T Bell Laboratories; present affiliation Bell Communications Research, Inc.

to other related systems (e.g., certain implicit token-passing systems and some ring networks) with small modifications.

Fasnet is a dual-channel, packet-switched LAN. Channel time is partitioned into contiguous slots, each of a duration long enough to transmit one packet of data plus the necessary overhead to allow for access control, addressing, error checking, etc. A voice source, engaged in a conversation, may be silent (either listening to the other party in the conversation or in a short pause between speech spurts) or active (producing a speech spurt). An active voice source is partitioned into segments, each of length T seconds. Each segment of voice is mapped into one information packet. The length of an information packet is thus determined by the code rate of the incoming voice source and the transmission rate of Fasnet. To maintain voice continuity, it is necessary to transmit one voice packet of the same source every T seconds. From the LAN's point of view, to maintain an approximate continuous service, it must offer service to voice once every T seconds. The initial portion of the T-second interval will be devoted to serving voice calls. The interval T between consecutive starts of voice service will be referred to as a service cycle. The service cycle period can be expressed in terms of the number of slots, and it is conceivable that the number of voice sources connected to Fasnet is larger than the number of slots in one service cycle. Obviously, the system cannot accommodate more simultaneously "active" voice source than the number of slots available in one service cycle without degrading the signal in some way; some kind of admission control needs to be exercised.

In this paper we assume that a station may be handling voice, data, or both. If we consider voice only, a station that is not in a voice session will be referred to as *idle*. A *nonidle* station producing a speech spurt (talk spurt) will be referred to as *active*; it will be referred to as *silent* if its voice session is in a silent condition.

From an operational point of view, calls already in progress, whether they be in a talk spurt or in a silent interval, must be guaranteed channel time when needed. Since data can be buffered and voice needs continuity, voice will be given a higher priority than data. New calls that are not admitted will be blocked. In the event that a subscriber is blocked, an appropriate tone (short busy) is returned to the subscriber and he/she would try again at a later time. (Of course a satisfactory system design would provide sufficient channel capacity that such an event would occur with no more than a prespecified probability.) On the other hand, unused voice capacity is available to data traffic until there is an increase in voice traffic. In this paper we propose an admission policy for new calls under the constraint that ongoing sessions are guaranteed channel access and that the average number of active voice calls is controlled to within a small neighbourhood of the target operating value, N_t. (Due to statistical variations in the number of active voice calls and the fact that ongoing voice sessions cannot be dropped, the mean number of active voice calls can only be controlled to within a neighbourhood of N_t.) We evaluate the performance of data under this admission policy. An alternative strategy has been explored in Ref. 10, in which data traffic is guaranteed a minimum capacity. Voice stations returning from the silent state to the active state are not guaranteed a voice slot, and clipping of a speech spurt can occur as experienced in an overloaded Time Assignment Speech Interpolation (TASI) system.[11] The maximum voice load is thus determined by the amount of clipping that may be tolerated. The study showed that, in the distributed environment of the LAN, performance approached very closely that of an ideal TASI system.

The priority admission mechanism and the data queueing behaviour in Fasnet are similar to those in Welnet.[8] The difference lies in the media access control, the network architecture, and the method of switching from one traffic type to another. In Fasnet, a "dead time", referred to as a sojourn time, usually accompanies switching from one type of service to another. In Welnet, provided the packet transmission time is greater than the propagation delay, there is no dead time due to switching of traffic type. Otherwise, the service disciplines of Fasnet and Welnet are similar.

Allowing voice sessions to capture the channel when needed will cause the service rate for data to fall below the data arrival rate during some epochs when the system is heavily loaded with voice and data. When such an event takes place, the traffic intensity, defined as the ratio of data arrival rate to service rate, will temporarily exceed unity. In this mode of operation, the aggregate data queue will experience a sustained growth with a nonstationary distribution, and the queueing process will be temporarily unstable. An approximate analysis method, based on a periodic switching single-server model and the fluid approximation method,[12] was developed in Ref. 8 to analyze the data queueing behaviour for integrated voice and data services on Welnet. The approximate analysis method involves a decomposition of the queueing process into stationary and nonstationary queues and a determination of the mean queue contribution from each of the operational modes. With a suitable modification to account for the sojourn time, the approximate analysis method developed in Ref. 8 is used to analyze the data queueing behaviour for integrated voice and data services on Fasnet.

This paper is organized as follows: Section II briefly describes the Fasnet architecture, which includes a description of the frame format of a Fasnet slot and the concept of a voice cycle. Section III discusses the evolution of the state equation for voice, which leads to the proposal of an ad hoc admission policy for new calls in Section IV. Fasnet provides centralized control of classes of traffic, where a class may be a traffic type, a priority, or some combination. However, access by individual stations is a distributed function, being determined by the station itself. It is important to assure that the centralized control is implemented in such a way that service to each traffic type remains fair. (We define a fair system as one in which all active stations have an equal chance to access the common channel.) In Section V we evaluate the performance of data for a "fair" Fasnet, under the new voice admission policy introduced in Section IV and the constraint that ongoing voice calls are guaranteed service.

II. FASNET ARCHITECTURE

Fasnet uses two channels with unidirectional signal propagation in each.[9] The first station (the head station in Fig. 1a) in each channel exercises centralized access control. In the event the first station malfunctions, the second station senses the lack of timing information and assumes the role of the first station. Since there is complete symmetry in the two channels, we will consider the information flow in one of them only.

Each channel has M taps, each of which supports a signal processor that handles both transmit and receive functions. The processor will be referred to as a station. Each station will support a number of user machines or processes. The first station permits access to each class by issuing a slot of a given type. A new class of traffic may be admitted

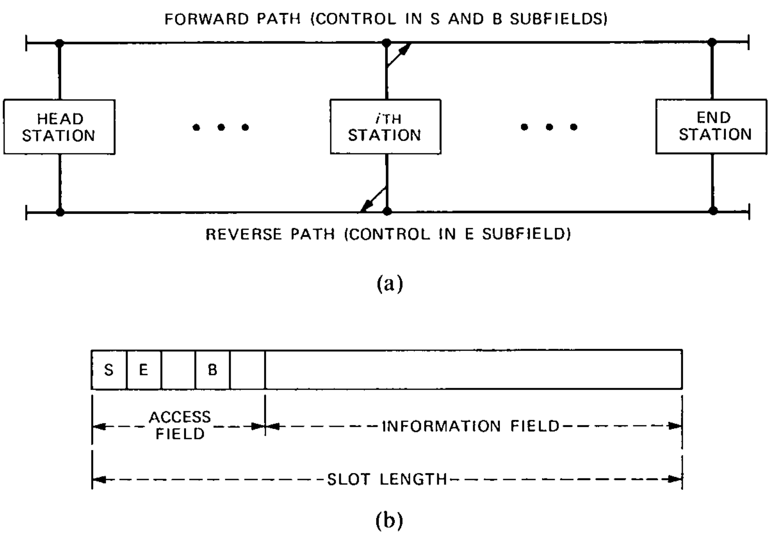

Fig. 1—Fasnet: (a) topology; (b) frame format.

by sending a request to the first station, which, in turn, upgrades its traffic classification to accommodate the new class.

2.1 Fasnet frame format

Channel time is divided into slots, each of a duration sufficiently long to transmit an information packet plus control, addressing, and cyclic redundancy checking information. A Fasnet slot comprises an access field and an information field in a manner depicted in Fig. 1. The access field is endowed with an S (Start), an E (End), and a B (Busy) subfield. The S subfield contains a code that designates the start of a sequence of slots of a particular traffic type. The B subfield is a single-bit subfield that indicates whether the information field is full or empty. Individual stations have the capability to read and write into the B bit. Guard times on either side of the B subfield allow for the read and write deliberations. As shown in Fig. 1, the contents of the information field are under the control of the individual stations. A slot is initiated with an empty information field and the content of the B subfield is a "0". If B = "1", the slot is busy; if B = "0", the slot is free. Stations can only write into free slots of the correct type.

2.2 Channel occupancy

Let r in b/s be the code rate of a voice source, R in b/s be the network transmission rate, and H be the number of overhead bits. H includes the access field, the source/destination addresses, control, and Cyclic Redundancy Check (CRC). Then, the slot length in bits is given by

$$F = H + rT \text{ bits/slot.} \tag{1}$$

The number of bits transmitted by the system in T seconds is RT, and the number of bits in an information packet is rT. The voice cycle length, L, in slots, is then

$$L = \frac{RT}{H + rT} \text{ slots.} \tag{2}$$

Let $C = 1$ be the normalized channel capacity, i.e., the channel can be utilized to transmit at most one packet per slot. Since a slot is longer than the time necessary to transmit one information packet, the maximum channel occupany, $U_{\max}$, is

$$U_{\max} \equiv \frac{rT}{H + rT} \text{ slots.} \tag{3}$$

The capacity required to coordinate packet transmission, C_s, is

$$C_s \equiv C - U_{\max}. \tag{4}$$

2.3 Code designations of priority scheduling

A voice cycle repeats every L slots. All admitted voice sessions are guaranteed one slot per voice cycle. It is expected that the number of slots taken by voice calls during any voice cycle will be less than L slots so that there will be slots available for data transmissions on a per-cycle basis. The concept of a voice cycle is depicted in Fig. 2, with a single slot expanded to emphasize the characteristics of the access field. The access field of each slot has three subfields, namely, S, E, and B. As indicated in Fig. 1a, the first (or head) station issues slots in the forward path, and the last (or end) station echoes acknowledgment in the reverse path. The S subfield contains a code word that designates the traffic type for that slot. The B subfield contains a token bit to indicate whether the slot is full or empty. The code word in the S subfield is set by the head station when the slot is issued. The B bit is to be set by the transmitting station. The end station observes whether a slot, after traversing the entire channel, is full or empty. If the slot is empty, then every station with the designated type of traffic has transmitted the permitted number of packets. The status of an empty slot is indicated (by the end station) in the E subfield. The occurrence of an empty slot signals the end of one type of service and the start of a new service type. The dead time associated with the circulation of an empty slot, as indicated to the head station by the marked E subfield, has been referred to earlier as the sojourn time.

The S subfield, which takes on one of the code words H, G, V, D, and C, determines the mode of the slot. To ensure "fairness", admission

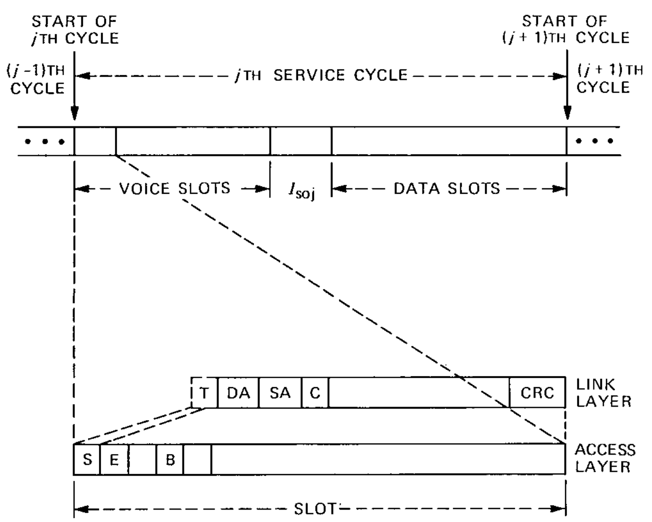

Fig. 2—Service cycle.

of new voice is carried out in two stages. Upon arrival, a new call enters the *wait* state. When the new voice in the wait state sees an H in the S subfield, it transits to the *hold* state and ultimately transmits when a visiting slot is free. The station model for new voice admission is depicted in Fig. 3. The code words H, G, V, D, and C have the following connotation:

- H is a type designation that allows new voice in the wait state to transit to the hold state and to start competing for a slot
- G allows new voice in the hold state to start competing for a slot
- V allows old "active" voice to start competing for a slot

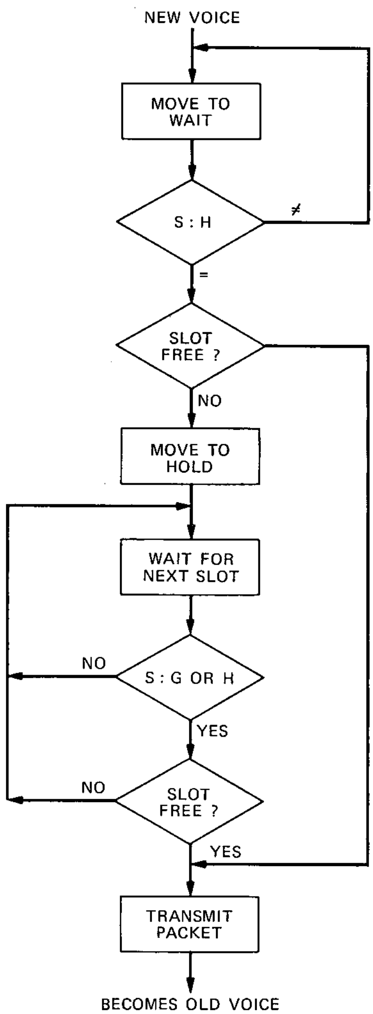

Fig. 3—New voice admission algorithm.

D allows "active" data stations to transit from the wait state to the hold state and to start competing for a slot

C allows active data stations in the hold state to start competing for a slot.

A data subcycle differs from a voice subcycle. Since a data subcycle may be interrupted by a voice subcycle, a data subcycle may span more than one service cycle. The access control code, D, starts a new data subcycle after all data stations in the wait state have been served so as to maintain fairness amongst the data stations. There are basically three types of services, namely, new voice calls, old voice calls, and data. Both H and G admit new voice calls, and both D and C allow data stations to transmit. However, each H (D) slot is issued at most once per voice (data) subcycle. That is, an H slot is followed by a string of G slots and a D slot is followed by a string of C slots. Fig. 4 shows the flowchart for the issuance of slots by the head station using feedback information, denoted by an asterisk, that arrives from the end station, after one channel propagation delay. The quantity Ω_j, determined in Section IV, is the number of new voice calls to be admitted in the jth admission cycle.

III. EVOLUTION OF STATE EQUATION

Consider just the voice subcycle. A station can be either in a voice session or idle. A voice session can be in a talk spurt or silent. Figure 5 shows the state diagram of a voice source.

Figure 5 depicts the state transitions at the start of the jth service cycle [or at the completion of the $(j-1)$th cycle]. The symbols used have the following notations:

- n_{j-1} denotes the number of *active* voice stations in the $(j-1)$th cycle
- d_{j-1} denotes the number of stations moving to the *silent* state after the completion of the $(j-1)$th cycle
- D_{j-1} denotes the number of stations terminating their voice sessions at the completion of the $(j-1)$th cycle
- b_j denotes the number of stations moving from the silent to the active state during the $(j-1)$th cycle
- B_j denotes the number of stations with new calls that are admitted in the jth voice cycle
- G_j denotes the number of new calls arriving in the jth voice cycle.

Then, the number of calls not accepted in the jth voice cycle is $G_j - B_j$. The state equation at the beginning of the jth voice cycle is

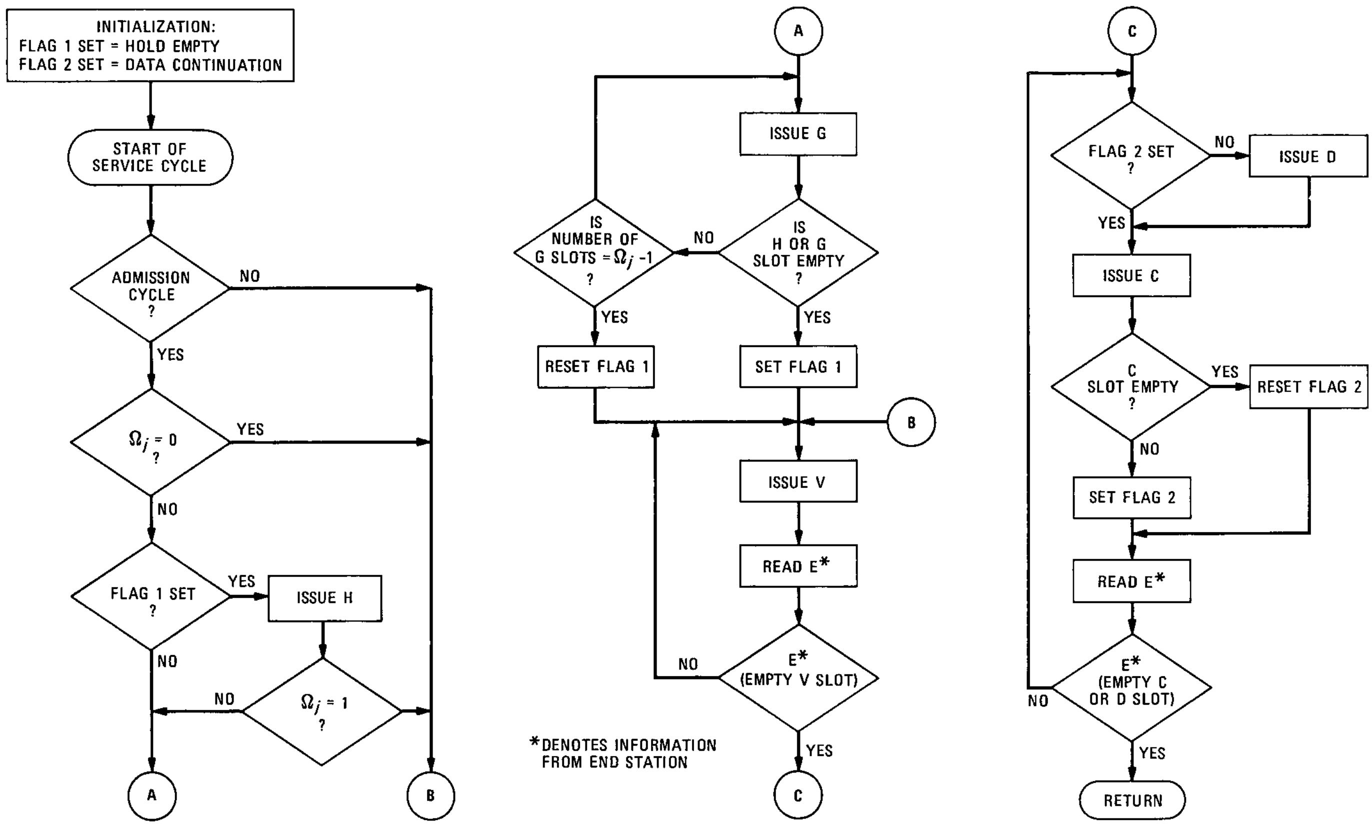

Fig. 4—Flow chart for slot issuance by head station with feedback from end station.

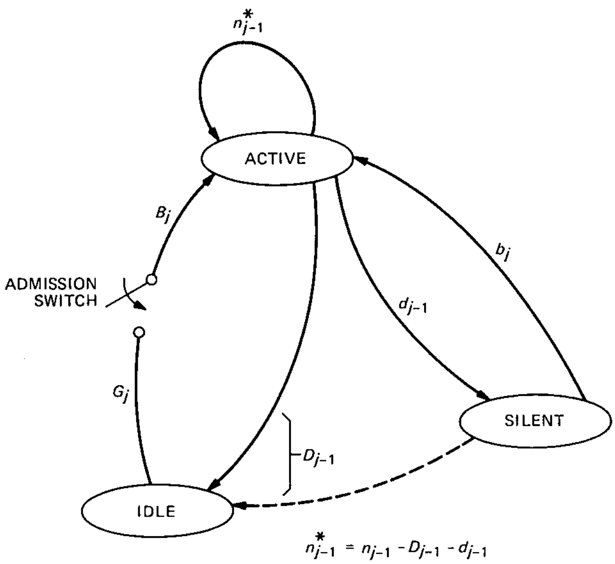

Fig. 5—State transition diagram for voice at start of jth service cycle, assuming deaths occur while in active state.

$$n_j = n_{j-1} - d_{j-1} - D_{j-1} + b_j + B_j. \tag{5}$$

Under the access control mechanism described in Section 2.1, n_{j-1} is the only known quantity in the above equation. (It is possible to obtain information about D_{j-1} and d_{j-1} by the incorporation of a more complex access control mechanism.) We will describe, in the next section, an admission policy for new calls under the Fasnet structure described in Section II and the state development in eq. (5).

IV. NEW VOICE ADMISSION CONTROL

It is desirable to maintain an equilibrium operating state such that the mean number of active voice sessions (in talk spurt) will be $\bar{n} \approx$ N_t, where N_t is the target number of active voice sessions. Let N be the total number of voice sessions to be supported and p be the fraction of voice sessions that are active. Then, $\bar{n} = pN$. In this paper, we let p = 0.4, i.e., 40 percent of the time a (nonidle) voice source is in talk spurt.

Suppose we call each run of identically designated slots, i.e., H, G, V, D, and C, a sequence denoted by S_x, $x \in \{H, G, V, D, C\}$. The termination of a sequence is detected by the last station whenever it sees the first empty slot. Before a new sequence of slots can be initiated, a sojourn time, I_{soj}, elapses. I_{soj} equals one slot time plus one round trip propagation delay. (The round trip propagation delay equals

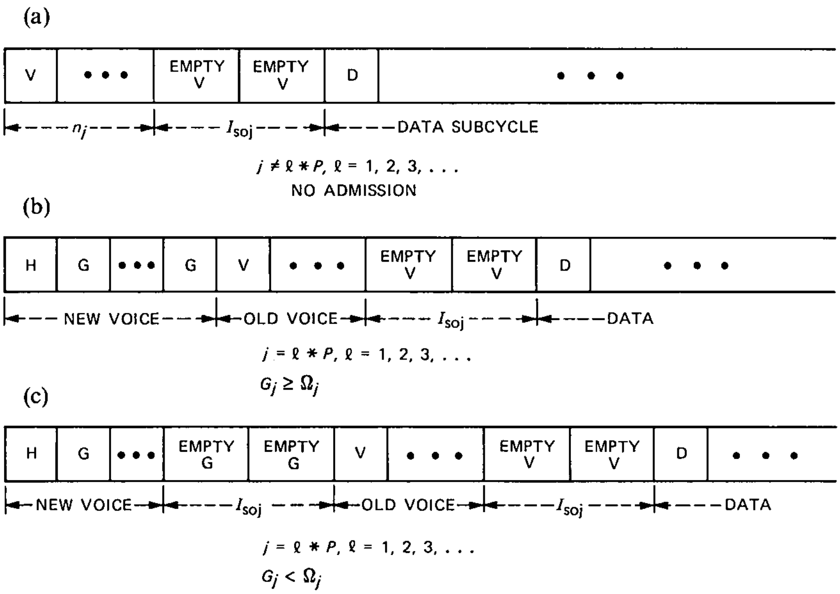

Fig. 6—Concept of sojourn time in jth cycle.

$2 \cdot \lceil \delta \rceil$ where δ, measured in slots, is the propagation delay in one direction, if the starts occur at the same instant in each channel, and the symbol $\lceil \cdot \rceil$ denotes the integer greater than or equal to its argument.)

At the start of a new service cycle, the first station issues a sequence, S_G, of Ω_j H- or G-designated slots (see Fig. 4). The number actually admitted is

$$B_j = \begin{cases} G_j & \text{if} \quad G_j < \Omega_j \\ \Omega_j & \text{if} \quad G_j \geq \Omega_j. \end{cases} \tag{6}$$

The format of a voice cycle is shown in Fig. 6. A sequence S_V will follow S_G. If $G_j \geq \Omega_j$, there will be no sojourn time separating S_G and S_V. Otherwise, there will be a sojourn time $\leq I_{soj}$ slots separating S_G and S_V. Also note from Fig. 6 that there is at least one sojourn time between an S_V and an S_C sequence within a service cycle, where S_C denotes a sequence of D or C starts.

Admission of new voice calls is controlled by the design parameter N_t. Since a cycle time is quite short, it is only necessary to admit new voice calls once every P cycles. Let Δ_j be the accumulated number of calls that left the system since the last admission cycle. Then Δ_j may be estimated by

$$\hat{\Delta}_j = \sum_{i=j-P+1}^{j} \frac{n_i}{p} \frac{T}{T_h}, \tag{7}$$

where T is the cycle length in seconds, T_h is the holding time of a voice session, and p is the probability that a voice session is in talk spurt. The value n_j/p is an estimate of the total number of voice sessions in the system in the jth cycle. On the assumption that, at steady state, the net movement from silent to talk spurt and vice versa is zero, the maximum number of active voice in the jth admission cycle can be approximated as

$$n_j = n_{j-P} + B_j - \hat{\Delta}_j. \tag{8}$$

The objective is to drive the system to operate at a point such that the average number of active voice approximately equals the design parameter N_t. We propose to compute a running average $\bar{n}_j$ by the following recursive formula:

$$\bar{n}_j = \gamma \bar{n}_{j-1} + (1 - \gamma) n_j, \tag{9}$$

where γ, in the range $0 < \gamma < 1$, is a constant factor that determines the relative weighting of the last running average and the current number of active voice. The constant γ also fades the memory of the recursive process and hence affects the size of the moving average window. To have a relatively large window, γ should only be slightly less than unity, e.g., $\gamma = 0.98$. Thus, during start-up, $\bar{n}_j$ will be small. By the same token, the number admitted is expected to be considerably larger than the number that leaves the system. Then, during start-up, the number of new voice already admitted should play an important role in controlling the admission process. When the system attains an equilibrium condition, the running average will approximately equal N_t and should assume primary control of the admission process. That is, at equilibrium, the number admitted should balance the number that left the system.

Let c_j be the parameter that accounts for the difference between the number admitted and the number that left. The number admitted in the past admission cycles should only be remembered until the system attains steady state, i.e., until $\bar{n}_j$ attains a value close to N_t. It is proposed to compute the control variable c_j by a recursive equation of the form

$$c_j = \eta c_{j-1} + B_j - \hat{\Delta}_j, \tag{10}$$

where $0 < \eta < 1$ is chosen to fade the memory at an appropriate rate. η should be related to the time constant with which the system attains steady state from start-up. In fact, the combination of (9) and (10) forms a robust iterative algorithm so that an optimum choice of η is not critically important. The strategy is to admit new voice calls whenever $\bar{n}_j + c_j < N_t$. The number to be admitted in the jth cycle is governed by the equation

$$\Omega_j = \max(0, N_t - \bar{n}_j - c_j). \tag{11}$$

We have run numerous tests of the admission control algorithm described in this section. In all cases, the system quickly locks onto a small neighbourhood of the target operating point, N_t. It is noted that admission control takes effect only when the total number of active voice exceeds N_t. If the total number of voice calls is fewer than N, so that the total number of active voice in the system is fewer than N_t, no admission control would be needed, and all calls would be admitted into the system.

V. QUEUEING ANALYSIS OF DATA PACKETS

Although the topology and the media access protocol of Fasnet[9] differ from those of Welnet,[8] the operational features of the two systems are similar. That is, each of these two systems has a deterministic media access protocol, which allows for priority scheduling and a bounded packet delivery time. With appropriate modifications, the periodic switching model and the mean value approximation method used in Ref. 8 to analyze the data buffer queueing behaviour are applicable here. The queueing analysis described below assumes that the procedure for admitting new voice calls is able to maintain

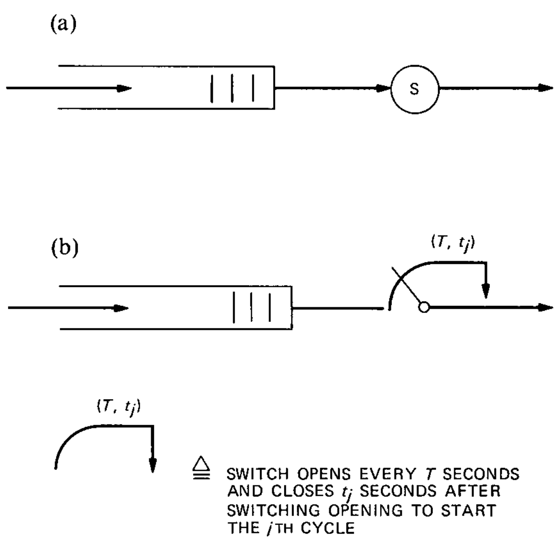

Fig. 7—(a) Single-server model. (b) Periodic switching model.

the voice service operating at the target value N_t, i.e., the mean number of active voice in the system approximately equals N_t.

The periodic switching model shown in Fig. 7 opens every T seconds and closes t_j seconds after the opening of the switch at the start of the jth service cycle. Here, t_j is the time needed to serve n_j active voice plus the total (i.e., including also those sojourns occurring in the data subcycles) sojourn time incurred in the jth service cycle. We have hitherto used n_j to represent the number of active voice calls. Since each active voice occupies one slot per cycle, in what follows, n_j will also denote the number of slots occupied by voice in the jth cycle.

Let m_j denote the number of sojourns in the jth cycle. We then have

$$t_j = \frac{(n_j + m_j I_{\mathrm{soj}})T}{L}, \text{ in seconds,} \tag{12}$$

where T, L, and I_{soj} are as previously defined. The time available for servicing data packets in the jth cycle is then $T - t_j$ seconds or $(L - n_j - m_j I_{\mathrm{soj}})$ slots. Since the transmission of each packet requires one slot time $(L - n_j - m_j I_{\mathrm{soj}})$, data packets can be transmitted in the jth cycle. We define

$$\mu_k = \frac{L - (n_j = k) - m_j I_{\mathrm{soj}}}{T} \text{ packets/s} \tag{13}$$

to be the service rate available to data in the jth cycle. The mean service rate is then

$$\mu_d = \frac{L - \bar{n} - \bar{m} I_{\mathrm{soj}}}{T} \text{ packets/s.} \tag{14}$$

Let λ_{d} be the aggregate data arrival rate in packet/s. The data utilization, ρ_{d}, is defined by

$$\rho_{\mathrm{d}} = \frac{\lambda_{\mathrm{d}}}{\mu_{\mathrm{d}}} = \frac{\lambda_{\mathrm{d}} T}{L - \bar{n} - \bar{m} I_{\mathrm{soj}}}. \tag{15}$$

For stable operation, it is required that $\rho_{\mathrm{d}} < 1$. For a given total number of voice calls in the system, N, there is a small probability (caused by the statistical fluctuation in the number of active voice calls in the jth cycle) that, in a certain jth cycle, $\lambda_j > [(L - (n_j = k) - m_j I_{\mathrm{soj}})]/T$ and the system is said to experience temporary overload, where λ_j is the number of arrivals in the jth cycle. Under the fluid approximation model to be discussed below, we approximate λ_j by λ_d, the mean arrival rate, and define the data utilization in the jth cycle by

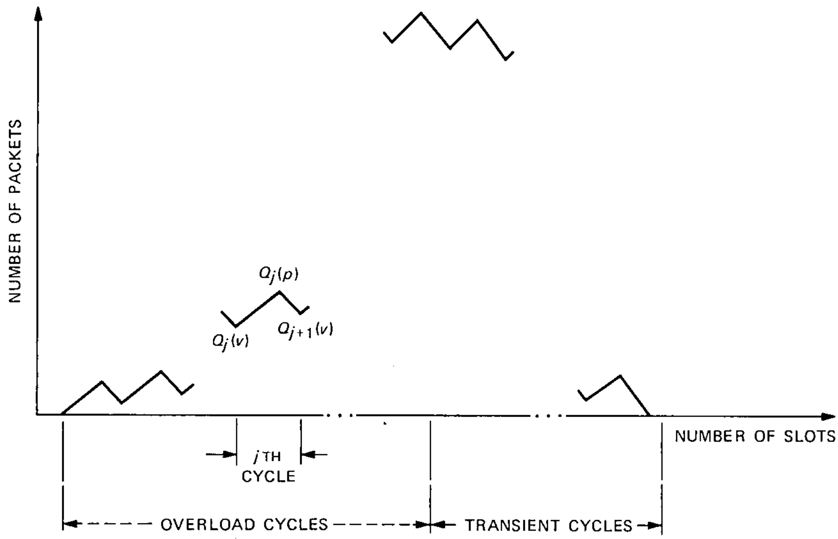

Fig. 8—Fluid approximation model for the jth cycle.

$$\rho_k = \frac{\lambda_d T}{L - (n_j = k) - m_j I_{soj}}. \tag{16}$$

Depending on the value of k, ρ_k can exceed unity.

The mean value, or fluid, approximation method[12] is depicted in Fig. 8, where $N_c(v)$ is the critical number of active voice at which the service available to data in the fluid approximation model equals the number of data packets that have arrived in the same cycle. The jth cycle is on the verge of overload when the combination of voice, sojourn time, and data is such that the queue at the end is the same as at the beginning of the jth cycle, i.e., $Q_{j+1} = Q_j$. At this operating point, the number of active voice is $N_c(v)$. If $n_j > N_c(v)$, there is a net growth in the queue and $Q_{j+1} > Q_j$. The queueing process temporarily becomes unstable and the system experiences overload. Since the number of voice in talk spurt obeys a binomial distribution and since the mean time between transitions from silent to talk spurt, and vice versa, is long compared to a cycle time (e.g., 1.50 seconds compared to 10 ms), once the system enters an overload state, there is every likelihood that it will stay in the overload state for many consecutive service cycles.

To analyze the queueing behaviour, we separate the service cycles into two sets, a stable set in which the queueing process is stationary and an overload set in which the queueing process is nonstationary. In addition, during the interval when the system is servicing voice only, new data arrivals will have to be scheduled for transmission. Let

Ω_s and Ω_o be the sets of stable and overload states, respectively. The average queue in the system is given by the sum of the contributions from the stable and overload states and the temporary queueing due to scheduling:

$$\bar{Q} = \bar{Q}(s) + \bar{Q}(o) + \bar{Q}_{sch}, \tag{17a}$$

where

$$\bar{Q}(s) = E[Q \mid k \in \Omega_s] \cdot \Pr(k \in \Omega_s) \tag{17b}$$

is the contribution from the stable states,

$$\bar{Q}(o) = E[Q \mid k \in \Omega_o] \cdot \Pr(k \in \Omega_o) \tag{17c}$$

is the contribution from the overload state, and $\bar{Q}_{sch}$ is mean queue due to scheduling of the data packets during a service cycle.

An approximate queueing analysis method, based on a decomposition of stable and overload operational modes, was developed in Ref. 8 for the queueing analysis of integrated voice/data services on Welnet. Functionally, the media access protocols of Fasnet and Welnet differ only in that Fasnet may incur a sojourn time whenever a change in the type of traffic to be served takes place. With the incorporation of the sojourn time into the analysis model, the method developed in Ref. 8 can be used to analyze the data queueing behaviour on Fasnet.

When the number of active data stations is small (at light loads), there will be many data subcycles within one service cycle and consecutive data subcycles will be separated by a sojourn time. While the existence of sojourn time represents a loss of channel capacity, not every occurrence of a sojourn affects the data queueing process. The ultimate limit occurs when every data station has packets to be transmitted (as is likely to happen in an overload state). We define the mean number of sojourn times per service cycle, which represents a loss in channel capacity, by

$$\bar{m} = \frac{L - \bar{n}(o) - \bar{m} \cdot I_{soj}}{M \cdot P_{max}} + 1, \tag{18a}$$

where the 1 in the right-hand side of (18a) represents the sojourn between the end of the voice subcycle and the beginning of the data subcycle, $\bar{n}(o)$ is the mean number of active voice in the overload state, M is the number of data stations, and $P_{max} \geq 1$ is the maximum number of packets that an active data station is permitted to send per data subcycle. A value of $P_{max} > 1$ is beneficial for small M, so that the mean number of sojourn times in the overload state is a minimum. For $M > L - N_c(v)$, a value of $P_{max} = 1$ suffices. The expression for $\bar{m}$ can be rearranged to yield

$$\bar{m} = \frac{L - \bar{n}(o) + M \cdot P_{\max}}{M \cdot P_{\max} + I_{\mathrm{soj}}}. \tag{18b}$$

It can be shown that the mean number of voice while in the overload state, $\bar{n}(o)$, is given by Ref. 8:

$$\bar{n}(o) = \frac{\sum_{k=N_c(v)+1}^{N} (n_j = k) \cdot \Pr(n_j = k)}{\sum_{k=N_c(v)+1}^{N} \Pr(n_j = k)}. \tag{19}$$

From Fig. 8, we deduce that

$$\lambda_d T = L - N_c(v) - \bar{m} I_{\mathrm{soj}}. \tag{20}$$

That is, when the number of active voice $n_j = N_c(v)$, the number of packets arrived in the jth cycle equals the number of packets served in the same cycle. The above equation can be rearranged to yield

$$N_c(v) = L - \lambda_d T - \bar{m} I_{\mathrm{soj}}.$$

Substituting $\lambda_d T$ by $\rho_d(L - \bar{n} - \bar{m} I_{\mathrm{soj}})$ from (20) and rearranging, we have

$$N_c(v) = L(1 - \rho_d) + \rho_d \cdot \bar{n} - \bar{m} \cdot I_{\mathrm{soj}}(1 - \rho_d). \tag{21}$$

The parameters $\bar{m}$ and $N_c(v)$ are related by a set of transcendental equations. $\bar{m}$ and $N_c(v)$ can be computed by the following algorithm:

1. Substitute $N_c(v)$ *for* $\bar{n}(o)$ in (18b) and then combine the result with (21) to obtain

$$N_c(v) = \frac{[L(1 - \rho_d) + \bar{n}\rho_d](M \cdot P_{\max} + I_{\mathrm{soj}}) - I_{\mathrm{soj}}(1 - \rho_d)(L + M \cdot P_{\max})}{I_{\mathrm{soj}}(1 - \rho_d)(L + M \cdot P_{\max})}.$$

2. Calculate $N_c(v)$.
3. Using this value of $N_c(v)$, calculate $\bar{n}(o)$ by eq. (19).
4. Calculate $\bar{m}$ using (18b).
5. Recalculate $N_c(v)$ using (21).

The approximate analysis model developed in Ref. 8 is based on the fact that the number of active voice obeys a binomial distribution and that the system (for voice) is at equilibrium when the number of active voice equals the mean, $\bar{n} = pN$, where p is the probability that a voice call is in talk spurt and N is the total number of voice calls in the system. When the number of active voice calls deviates from $\bar{n}$, there is a tendency for the system to move towards the equilibrium point.

Define d_k to be the net drift toward the equilibrium point. Then, at any operating point $n_j = k$, the net drift, d_k, is given by Ref. 8:

$$d_k = \alpha \cdot T \cdot (n_j = k) - \beta \cdot T \cdot (N - (n_j = k)) \,|\, k \in \Omega_o \quad \text{slots/cycle}, \tag{22}$$

where α^{-1} and β^{-1} are, respectively, the mean holding times of a talk spurt and a silent event and T, as previously defined, is the cycle length in seconds. With probability Pr[overload] $=$ Pr$[k \in \Omega_o] =$ Pr$[n_j > N_c(v)]$, the system is in the overload state. Once it enters the overload state, the mean time that the system spends in the overload state is a function of the net drift d_k given by (22). The mean and second moment of the duration that the system spends in the overload state is given by Ref. 8:

$$\tau_o = \frac{\displaystyle\sum_{k=N_c(v)+1}^{N} \frac{(N - (n_j = k))\Pr(n_j = k)}{d_k}}{\displaystyle\sum_{k=N_c(v)+1}^{N} (N - (n_j = k))\Pr(n_j = k)}$$

$$= \frac{\displaystyle\sum_{k=N_c(v)+1}^{N} \frac{(N - (n_j = k))\Pr(n_j = k)}{\alpha \cdot T \cdot (n_j = k) - \beta \cdot T \cdot (N - (n_j = k))}}{\displaystyle\sum_{k=N_c(v)+1}^{N} (N - (n_j = k))\Pr(n_j = k)}, \tag{23}$$

and

$$s_o = \frac{\displaystyle\sum_{k=N_c(v)+1}^{N} \frac{(N - (n_j = k))\Pr(n_j = k)}{[\alpha \cdot T \cdot (n_k = k) - \beta \cdot T \cdot (N - (n_j = k))]^2}}{\displaystyle\sum_{k=N_c(v)+1}^{N} (N - (n_j = k))\Pr(n_j = k)}. \tag{24}$$

The coefficient of variation of the duration in the overload state is then given by

$$C_o^2 \equiv \frac{s_o}{[\tau_o]^2} - 1. \tag{25}$$

It turns out that $C_o \ll 1$. This is a consequence of the fact that the increase in the number of active voice, which drives the data queueing process into an overload condition, is a slowly fluctuating process. Thus, the second moment of the overload duration has negligible effect on the queueing behaviour when the system is in the overload state.

As observed in Ref. 8 and depicted in Fig. 9, a transient state always follows an overload state. The chain of service cycles depicted in Fig. 9 will be referred to as an "overload/transient" chain. The contribution to the average queue in the system from an overload/transient chain is given by the queue in a typical overload/transient chain multiplied

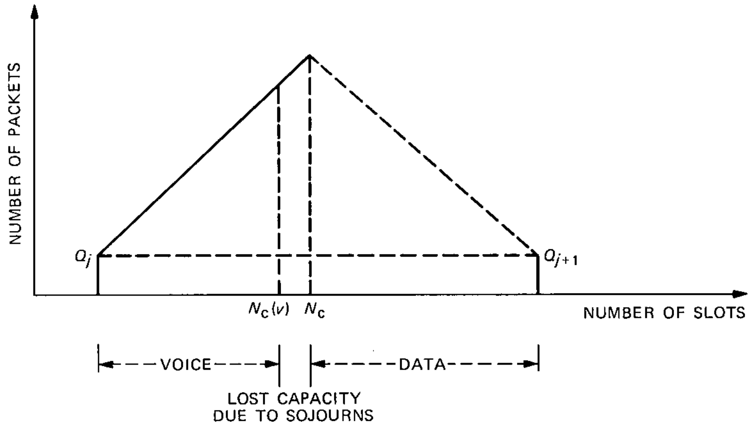

Fig. 9—Chaining of overload/transient cycles.

by the probability of the occurrence of an overload/transient chain. The mean duration of the overload portion of the chain depicted in Fig. 9 is τ_o. The queue growth in the overload state is given by the difference of the queue size at the ends of consecutive service cycles, i.e., $\Delta Q_j = q_{j+1}(v) - Q_j(v)$. The average queue growth per cycle will be denoted by $E[\Delta Q]$. The average queue growth in an overload state is then given by an accumulation of the growth per cycle over the overload duration, i.e.,

$$\sum_{i=1}^{\lceil \tau_o \rceil} iE[\Delta Q],$$

where $\lceil x \rceil$ denotes the integer part of $x + 0.5$, plus $\lambda_d N_c(v)P/2R$, which is the average growth in the cycle when the system enters the transient state. The queue shrinkage will be slightly faster than the queue growth so that the mean duration in the transient state is nominally shorter than τ_o. This is due to the fact that the number of active voice while the system is in the overload state is larger than that when the system is in the transient state. The average number of slots available for data per overload cycle is $L - \bar{n}(o) - \bar{m}\cdot I_{soj}$; that per transient cycle is $L - \bar{n}(t) - \bar{m}\cdot I_{soj}$, where $\bar{n}(o)$ and $\bar{n}(t)$ are the mean number of active voice in the overload and transient states, respectively. $\bar{n}(o)$ is given by (19), and $\bar{n}(t) \leq N_c(v)$. For convenience, we approximate $\bar{n}(t)$ by $N_c(v)$ and define

$$h = \frac{L - \bar{n}(o) - \bar{m}\cdot I_{soj}}{L - N_c(v) - \bar{m}\cdot I_{soj}} \leq 1 \tag{26}$$

to be the fractional contribution from the transient portion to the queue size in an overload/transient chain. Then, the number of packets in an average overload/transient chain is given by Ref. 8:

$$E[Q \mid k \in \Omega_o] = (1+h)\left\{\frac{\lceil \tau_o \rceil \cdot (\lceil \tau_o \rceil + 1)}{2} E[\Delta Q] + \frac{\lambda_d \cdot \bar{n}(o) \cdot P}{2R}\right\}, \tag{27}$$

where

$$E[\Delta Q] = \lambda_d T - \{L - \bar{n}(o) - \bar{m} \cdot I_{soj}\} \tag{28}$$

is the average queue growth per cycle when the system is in an overload state. The average queue contribution from the overload/transient chain is then given by

$$\bar{Q}(o) = E[Q \mid k \in \Omega_o] \cdot \Pr[k \in \Omega_o]. \tag{29}$$

The stationary contribution can be written as

$$\bar{Q}(s) = \sum_{k=0}^{N_c(v)-1} E[Q \mid n_j = k] \Pr(n_j = k), \tag{30}$$

where $E[Q \mid n_j = k]$ is the mean number of packets in the system given the number of voice in the jth cycle is less than $N_c(v)$ and, for Poisson arrivals, is given by the M/G/1 formula:

$$E[Q \mid n_j = k] = \rho_k + \frac{\rho_k^2(1 + C_d^2)}{2(1 - \rho_k)}. \tag{31}$$

ρ_k is the utilization defined by (16) and C_d^2 is the squared coefficient of variation of service time. Since voice changes slowly compared to a cycle time, $C_d^2 \ll 1$.

The Pollaczek-Khinchin M/G/1 formula describes the queueing effect at the imbedded points of the service process which, in our case, correspond to the ends of service cycles. During the initial part of the service cycle when the system is serving voice stations, the data stations experience a temporary queueing. The average temporary queue per cycle corresponds to a delay during which a typical data packet is scheduled for channel access. Depending on the state of the system, the number served in a given cycle may be fewer than the number that has arrived during the same cycle. The queueing of those new arrivals that are not served in the same cycle is accounted for by the M/G/1 formula. To compute the queueing due to scheduling, it is only necessary to consider new arrivals when the system is in the stable mode. Consider the fluid approximation model depicted in Fig. 10. The total number of packets that need to be queued temporarily

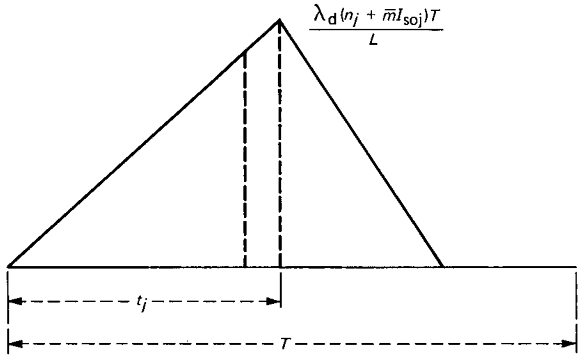

Fig. 10—Fluid approximation model for the evaluation of mean queue due to scheduling.

in the jth cycle is given by the area under the triangle; the average number of packets in the temporary queue is given by the area divided by the base. Thus, the average number of packets in the temporary queue due to scheduling during one cycle is given by

$$\bar{Q}_{\mathrm{sch}} = \frac{\lambda_d T}{2L} \sum_{k=0}^{N_c(v)} [(n_j = k) + \bar{m} \cdot I_{\mathrm{soj}}] \cdot \Pr(n_j = k). \tag{32}$$

The mean number of packets in the system is then given, as indicated in (17a), by the sum of $\bar{Q}(o)$, $\bar{Q}(s)$, and $\bar{Q}_{\mathrm{sch}}$, which are prescribed respectively by (29), (30), and (32).

VI. NUMERICAL RESULTS

Recognizing the difficulty with an exact analysis of the data queueing behaviour in an integrated voice and data service environment, we have introduced an approximate analysis in Section V. Another approach to evaluating the system performance is by means of computer simulation or emulation. Emulation portrays all the steps involved in the actual system. The drawback in an emulation process is that it is extremely time-consuming and that it is not feasible to emulate a high-rate system. We compare the approximate analysis results with those obtained by means of computer emulation. Based on the closeness of the approximate analysis results to those obtained from emulation on a 10-Mb/s Fasnet, we conjecture that the performance results obtained for a high-rate Fasnet (e.g., 100 Mb/s) using the approximate analysis method are reasonable portrayals of system performance.

6.1 Simulation

The simulation consisted of an emulation at the packet level. We assume the system operates at 10 Mb/s. The length of the line was assumed to be 2.5 km, resulting in normalized propagation and signal processing delay for a single line of 0.2 slot. Data and voice stations were randomly positioned along the line. We assumed that 142 voice station pairs were in conversation with an additional 20 active data stations. The other parameters used in the simulation are $r = 64$ kb/s, $T = 10$ ms, and $P = 700$ b/packet.

An ongoing voice conversation is modeled as a two-state Markov process. It spends 40 percent of the time in the active state and 60 percent in the silent state (activity factor = 0.4). The average talk spurt duration is 1.5 seconds and the average silent interval is 2.25 seconds, both with an exponential length distribution. All data stations generated traffic at the same mean rate and each station transmits a maximum of three packets from its queue during one data subcycle ($P_{max} = 3$). Data packets arrive at each station according to a Poisson distribution.

Each simulation run was for 500,000 packets, corresponding to a duration of 35 seconds. The simulation started with voice conversations assigned to talk and silent stations in the same proportion as the steady-state distribution. To ensure that the steady state was reached before data were recorded, statistics were not gathered until after the first 20,000 packets.

Performance is measured in terms of the average number of packets in the system. In a given service cycle, there is temporary queueing when the system is serving the voice stations. The queue at the end of each cycle is smaller than it is during the cycle. The simulation results shown in Fig. 11 reflect the effects of temporary queueing due to scheduling during one service cycle.

For each data point shown in Fig. 11, the results for between five and twenty independent runs were combined. At lower data rates, the variance is small and few runs were required to obtain accurate measures of the mean data queue. At higher data rates, the variance was significantly larger. The simulation did not guarantee that the average amount of voice traffic would be equal to the value used to calculate the theoretical results. This affects the measurement of the average queue size for data since the channel capacity available for data depends directly on the capacity taken by the voice stations. To improve the estimate of the mean queue for data, the average number of active voice conversations (which was measured for each run) was regressed against mean queue and, from this regression, the value at 56.8 (0.4×142) active calls (the value used in the calculations) was estimated. An example of this is shown in Fig. 12, where the experi-

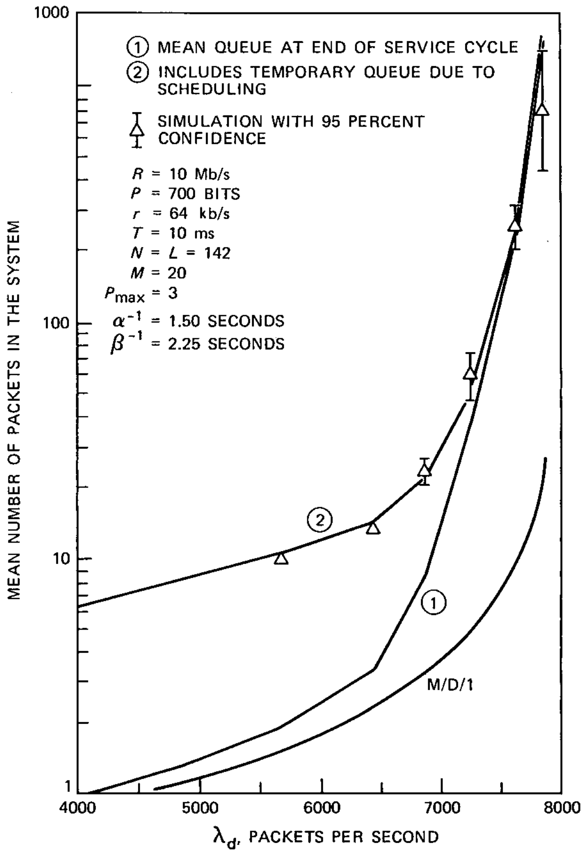

Fig. 11—Data queueing characteristics.

mental points, regression line, and 95-percent confidence interval are shown.

6.2 Calculation

The same parameters used in the simulation were employed to calculate the average queue contributions from the stable and overload modes and channel access scheduling, using the equations developed in Section V. The calculated results are tabulated in Table I. For comparison purposes, the mean number of packets in the system is plotted in Fig. 11 for the cases with and without the mean scheduling queue per cycle. Inspection of Fig. 11 shows that the calculated results closely approximate those obtained from simulation.

The data arrival rate in packets/slot is expressible as $\lambda_d T/L$. Then, by Little's formula,[13] the mean packet delay is given by

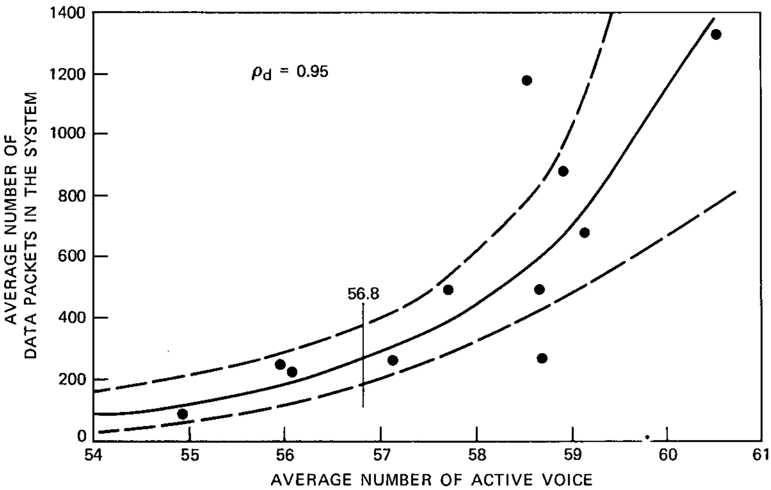

Fig. 12—Regression analysis of mean data queue vs. mean number of active voice with 95-percent confidence.

Table I—Mean number of packets in the system as a function of the arrival rate λ_d in packets/s*

ρ_d	λ_d	$N_c(v)$	$\bar{m}$	Pr[over-load]	$\bar{Q}_{sch}$	$\bar{Q}(s)$	$\bar{Q}(o)$	$\bar{Q}$
0.50	4091.80	97.72	1.68	2.6449e-12	5.19	1.01	—	6.21
0.60	4891.12	89.43	1.82	1.6032e-08	6.86	1.38	—	8.24
0.70	5691.21	81.19	1.95	1.4707e-05	8.78	1.96	—	10.73
0.80	6483.12	73.01	2.08	2.3096e-03	11.02	3.25	0.23	14.51
0.85	6877.15	68.94	2.15	2.3249e-02	12.01	4.71	4.31	21.03
0.90	7269.90	64.88	2.21	9.4180e-02	12.16	6.86	30.56	49.58
0.95	7895.67	60.83	2.28	2.6206e-01	10.54	8.64	228.68	247.86
0.98	7895.67	58.41	2.32	3.8351e-01	8.02	6.30	1512.02	1526.34

* $R = 10$ Mb/s, $r = 64$ kb/s, $H = 60$ bits, $P = 700$ b/packet, $\alpha^{-1} = 1.50$ seconds, $\beta^{-1} = 2.25$ seconds, $L = 142$ slots, $N = L$, $M = 20$, $P_{max} = 3$, $I_{soj} = 2$.

$$\bar{d} = \bar{Q}L/\lambda_d T.$$

The analysis and simulation results of the delay-throughput characteristics are shown in Fig. 13. These curves are an alternative presentation of the results shown in Fig. 11.

To demonstrate the effects on data queueing when the number of voice calls is reduced, we compare in Fig. 14 the cases for $N = 128$, 136, and 142. It is observed that at a relatively high data rate, the average queue decreases significantly when the number of voice calls is reduced by approximately 5 percent, i.e., from 142 to 136. The improvement in queueing comes from the fact that a reduction in the number of voice calls is accompanied by a relatively large reduction in the probability that the system is in the overload state.

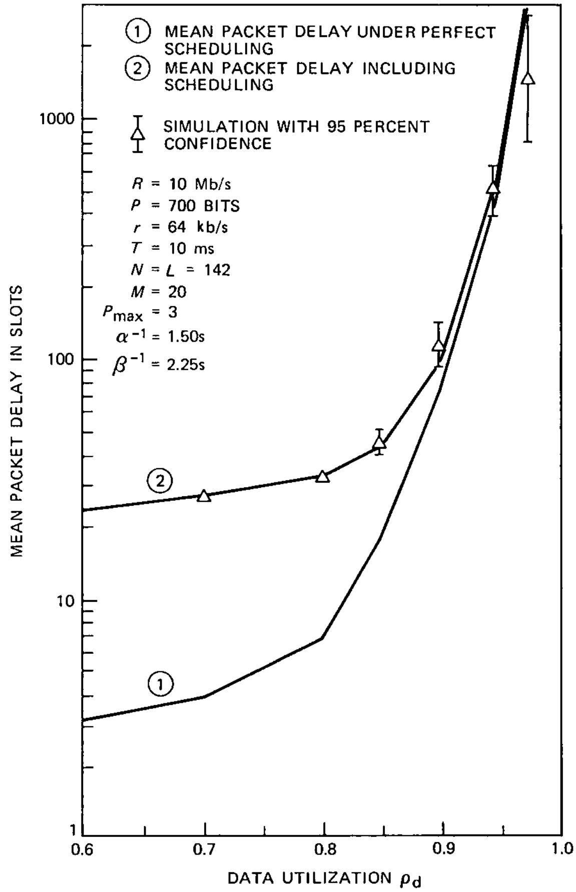

Fig. 13—Mean data packet delay characteristics.

As the voice arrival rate r decreases or the network transmission rate increases, the number of slots in a cycle increases. For example, with $R = 100$ Mb/s, $r = 64$ kb/s, $P = 700$ b/packet, and $T = 0.01$ second, the number of slots per cycle is approximately 1408. For $r =$ 32 kb/s and $P = 380$ b/packet ($H = 60$ bits in each case), the number of slots per cycle is approximately 2624. Letting the number of voice calls $N = L$, the average number of data packets in the system as a function of the traffic intensity ρ_d is plotted in Fig. 15. It is noted that the mean contribution to the queue due to scheduling is the same for the two cases, since the packet length for the $r = 64$ kb/s case is 1.8421 times longer than that of the $r = 32$ kb/s case. The longer cycle length (due to a shorter packet length) case exhibits better long-term queueing performance (excluding scheduling queueing). Consider for

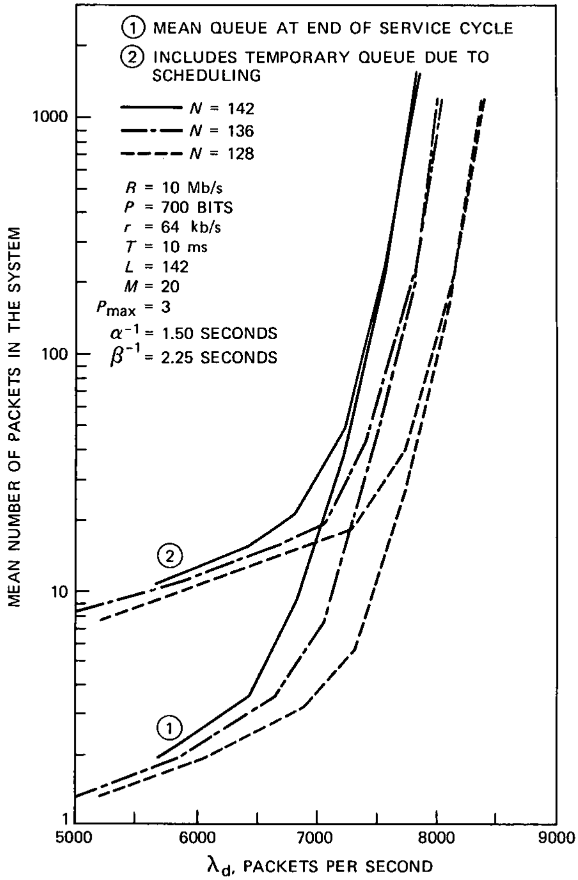

Fig. 14—Effects on queueing with a decrease in the total number of voice stations.

example, at $\rho_d = 0.95$, the average number in the system (excluding scheduling queueing) is 18.21 packets for the $r = 64$ kb/s case and 12.72 packets for $r = 32$ kb/s. Since the relative packet length ratio is 700/380, the effective mean queue for the 64 kb/s case is 2.64 times that of the shorter packet case. Thus, the mean queue for the case where $P = 700$ bits is equivalent to 33.58 380-bit packets at $\rho_d = 0.95$ compared to 12.72 380-bit packets for the 32 kb/s case at the same utilization.

For comparison purposes, the M/D/1 curve is also plotted in Figs. 11 and 15. The M/D/1 result is based strictly on a stationary distribution, i.e., ignoring the statistical variations associated with the speech spurts. At light loads where the system always operates in a stable mode, the M/D/1 represents a reasonable lower bound. At heavy loads where the system experiences a reasonable percentage of

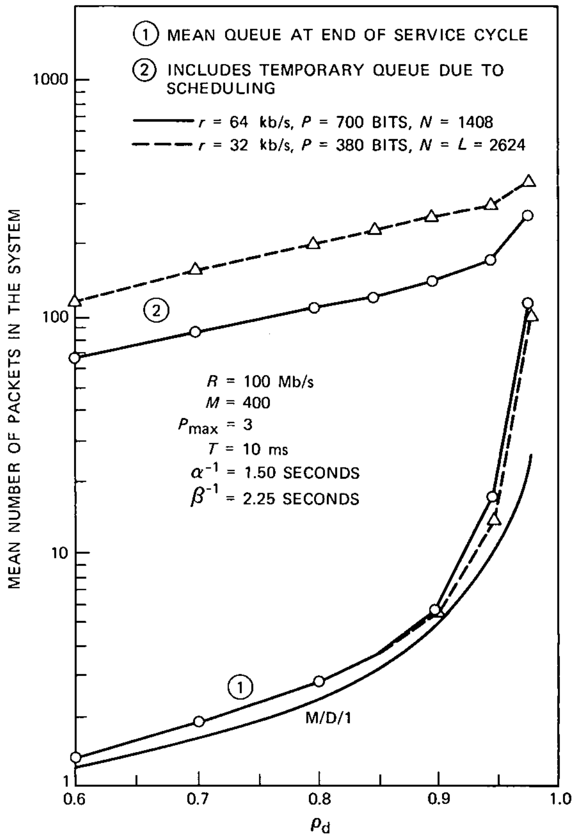

Fig. 15—Queueing characteristics for different packet lengths.

overload, the M/D/1 curve serves no useful comparison (see Fig. 11). Of significance are the numerical results shown in Fig. 15, in which the perfect scheduling Fasnet results deviate from the M/D/1 curve by almost a constant amount up to 95-percent data utilization. This is a consequence of the fact that, at the higher speed ($R = 100$ Mb/s), overload does not become significant until the data utilization approaches unity. The discrepancy between the M/D/1 and the perfect scheduling Fasnet curves can be attributed to the presence of sojourn time in the Fasnet access discipline.

VII. DISCUSSIONS AND CONCLUSIONS

In previous work, Limb and Flamm[10] described a scheme for accommodating voice and data that differs from the present method in that calls returning from idle may get clipped. New calls are accepted until an unacceptable level of clipping is approached, whereupon they are blocked. Unused voice capacity may be utilized by data, although a sojourn time is required to switch between the voice subcycle and the data subcycle unless the voice subcycle is full.

A direct comparison between the two schemes is difficult because of the different way in which the voice traffic is treated—the new calls admission policy and the clipping of talk spurts in one case and the absence of clipping in the other. However, a couple of observations can be made: Both systems have comparable performance since they both utilize the available slots efficiently when traffic is heavy; clipping may be undesirable in certain instances, for example, where a number of Fasnet links are connected in tandem. A speech spurt could be clipped more than once, leading to a noticeable degradation. In this instance, the present method would be preferable.

The global access control that is enabled by the endowment of each slot with a type designation renders Fasnet particularly suitable for integrated services. We have described the ramifications for integrated voice and data services on Fasnet and a policy for admitting new voice into the system. Once admitted, a voice station is guaranteed service when needed. Voice stations are served on a periodic basis, so that each voice source sees almost synchronous service, i.e., the service is synchronous within the statistical variations dictated by the binomial distribution of the silent and talk spurt intervals in a voice process. The mean interpacket delay can be expected to be negligibly small. Note that the service discipline described will not clip any voice packets from a talk spurt.

Since data traffic can be buffered, it is not necessary to maintain continuous service for data. During intervals when the system is servicing voice stations, data stations are temporarily locked out and have to be buffered. Depending on the data input load relative to the service time available for data during one service cycle, the temporary queues may be cleared entirely at the end of the service cycle, or some residual packets will carry over from one cycle to the next. If the latter phenomenon occurs over many consecutive cycles, the system will sustain a net growth in its queues. Under this condition the system experiences overloading, and the data utilization during these cycles will exceed unity for a short period and the data queues will experience significant growth.

The conventional approach to analyzing the queueing behaviour is the generating function analysis method. However, the size of the state space in a practical network renders the generating function method impractical. An approximate analysis method, based on a merge of a periodic service single-server model and a fluid approximation method, has been developed in Ref. 8 to analyze the queueing behaviour for integrated voice and data services on Welnet, a local area network in which the service discipline is functionally similar to that of Fasnet. With a suitable modification to account for the sojourn time that accompanies each switch in the type of service by the Fasnet access control, the approximate analysis method introduced in Ref. 8 was used to analyze the data buffer queueing behaviour for integrated voice and data services on Fasnet.

System performance may also be evaluated by means of computer simulation. Simulation is time-consuming and it is impractical to simulate high transmission rate systems. In this study, we simulated a 10-Mb/s Fasnet and measured performance in terms of the mean queue in the system as a function of the data input load. The analytic results closely approximate those obtained from simulation. It is conjectured that the analysis method yields reasonably accurate results. Analytical results for a 100-Mb/s Fasnet have been presented. An interesting observation is that the larger bandwidth has the effect of reducing the system overload probability.

In this paper we used an approximate model to analyze the network performance. We verified the reasonableness of the approximate model. An alternative approach is to approximate the correlated voice source by a memoryless process, e.g., a birth-and-death process. The validity of such an approximation needs to be shown. An analysis of the backlog in the buffers for a birth-and-death process is discussed in Ref. 14.

REFERENCES

1. K. Kummerle, "Multiplex Performance of Integrated Line and Packet-Switched Traffic," Proc. 2nd Int. Conf. Comp. Commun., Stockholm, Sweden, August 1974, pp. 517–23.
2. 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.
3. C. J. Weinstein, M. K. Malpass, and M. J. Fischer, "Data Traffic Performance of an Integrated Circuit-and Packet-Switched Multiplex Structure," IEEE Trans. Commun., *COM-28* (June 1980), pp. 873–7.
4. B. Maglaris and M. Schwartz, "Performance Evaluation of a Variable Frame Multiplexer for Integrated Switched Networks," IEEE Trans. Commun., *COM-29* (June 1981), pp. 800–7.
5. I. Gitman, W-N. Hsieh, and B. J. Occhiogross, "Analysis and Design of Hybrid Switching Networks," IEEE Trans. Commun., *COM-29* (September 1981), pp. 1290–1300.
6. B. R. Iyer and J. B. Sinclair, "Analysis of Data Delays in an Integrated Data-Voice Link," Technical Report 8204, Electrical Engineering Department, Rice University, Houston, Texas, April 1982.
7. A. Leon-Garcia, R. H. Kwong, and G. F. Williams, "Performance Evaluation Methods for an Integrated Voice/Data Link," IEEE Trans. Commun., *COM-30* (August 1982), pp. 1848–58.
8. J. W. Mark, "Priority Scheduling for Integrated Voice/Data Services in Local Area Networks," CCNG Report E-111, Computer Communication Networks Group, University of Waterloo, Waterloo, Ont., Canada, March 1983.
9. J. O. Limb and C. Flores, "Description of Fasnet—A Unidirectional Local Area Communications Network," B.S.T.J., *61,* No. 7, Part 1 (September 1982), pp. 1413–40.

10. J. O. Limb and L. E. Flamm, "A Distributed Local Area Network Packet Protocol for Combined Voice and Data Transmission," GLOBECOM '82 Conf. Rec., November 29–December 2, 1982, Miami, Florida, pp. B2.5.1–6.
11. C. J. Weinstein, "Fractional Speech Loss and Talker Activity Model for TASI Packet-Switched Speech," IEEE Trans. Commun., *COM-26* No. 8 (August 1978), pp. 1253–7.
12. G. F. Newell, *Applications of Queueing Theory*, London: Chapman and Hall Ltd., 1972, Chapter 2.
13. J. D. C. Little, "A Proof of the Queueing Formula $L = \lambda W$," Oper. Res., *9* (May 1961), pp. 383–7.
14. S. Halfin, "The Backlog of Data in Buffers with Variable Input and Output Rates," Proc. of the Second International Symposium on Performance of Computer-Communications Systems, Zurich, March 21–23, 1984.

AUTHORS

John O. Limb, B.E.E., 1963, Ph.D. (Electrical Engineering), 1967, University of Western Australia, Nedlands; Research Laboratory, Australian Post Office, 1966–1967; AT&T Bell Laboratories, 1967–1983. Present affiliation Bell Communications Research, Inc. At AT&T Bell Laboratories, Mr. Limb worked for a number of years on the coding of color and monochrome picture signals to reduce channel capacity requirements, and has published widely in this area. He is now manager of the Communication Research Department in Bell Communications Research, Inc. His current research interest is wideband, local area networks. Fellow, Institute of Electrical and Electronics Engineers; member, Association for Computing Machinery, Association for Research in Vision and Opthalmology, Optical Society of America.

Jon W. Mark, B.A. Sc., 1962, University of Toronto; M.Eng., 1968, Ph.D., 1970, McMaster University; Canadian Westinghouse, 1962–1968; IBM, Thomas Watson Research Center, 1976–1977; AT&T Bell Laboratories, 1982–1983; University of Waterloo, 1970—. At the IBM Research Center Mr. Mark was a visiting scientist. At AT&T Bell Laboratories, he was a consultant to the Computer Technology Research Laboratory. He worked for a number of years in the areas of radar and sonar signal processing, equalization, data compression, and spread spectrum communication. At present, Mr. Mark is a Professor of Electrical Engineering at the University of Waterloo. His current research interests are network modeling and performance evaluation, satellite networks, and local area networks. Senior Member, IEEE.