Satellite Packet Communication—Multiple Access Protocols and Performance

SIMON S. LAM, MEMBER, IEEE

Abstract—Satellite communication systems have traditionally been designed for voice traffic. Multiple access protocols for conflict resolution have typically been channel-oriented with either fixed or demand assignment. Data communications, however, have much more diverse traffic characteristics and transmission requirements than voice communications. We present in this paper an overview of two major categories of packet-oriented multiple access protocols: contention and reservation protocols. A traffic model suitable for the data communications environment is first introduced. A key element in this model is user-specified message delay constraints. Our primary performance measure of a protocol is the channel throughput versus average message delay tradeoff characteristic. The main attributes of the two categories of packet-oriented protocols are discussed. Four specific contention protocols are described and their performance characteristics are examined. Design considerations of the two important components of reservation protocols, reservation channel and distributed global queue, are discussed. Three reservation protocols with distributed control are described. Finally the performance of channel-oriented protocols and the two classes of packet-oriented protocols are compared using a variety of traffic models.

1. INTRODUCTION

WE ARE witnessing two rapidly growing fields in the world of communications: packet switching networks and satellite communications. Over the past decade, the sharply declining cost of computing has made possible the emergence of packet switching as a cost effective technology for the transmission of digital data. In addition to improving the economics of data communications, packet switching networks also provide enhanced reliability and functional flexibility of the communication path over circuit switching networks [ROBE 78]. Presently the transmission of digital voice in packet networks is also under extensive investigation [GITM 78]. The ever increasing importance of packet networks is evidenced by the large number of packet-oriented public data networks in existence or being planned in many countries [KELL 78]. Within the same time period, communication satellite system costs have come down drastically [ABRA 75]. A union of the two technologies appears to be most promising. In addition to potential cost reductions, satellites offer special capabilities that can be used to great advantage in packet networks. A shared broadcast satellite channel provides a fully connected network topology with direct "logical" connections between all earth station pairs. It also enables the traffic loads of a large population of geographically distributed users to be statistically averaged via some suitable algorithm that provides for dynamic allocation of the satellite transmission capacity.

Manuscript received December 28, 1978; revised April 27, 1979. This work was supported in part by the National Science Foundation under Grant ENG78-01803. Part of this work was performed while the author was with the IBM T. J. Watson Research Center, Yorktown Heights, NY, and was presented at the International Conference on Communications, Chicago, IL, June 1977.

The author is with the Department of Computer Sciences, University of Texas at Austin, Austin, TX 78712.

The problem of multiple access in the design of satellite systems has been solved in the past with voice communications in mind. The design objective has been to maximize the satellite traffic carrying capacity in terms of the number of (voice grade) channels for given constraints of power, bandwidth, error rate, etc. The common multiple access techniques are: frequency division multiple access (FDMA), time division multiple access (TDMA), and code division multiple access (CDMA). In this paper we shall focus only upon the protocols for dynamic allocation of the satellite transmission capacity. The central problem is that of *conflict resolution* among users desiring transmission capacity. The methods for conflict resolution will be referred to as *multiple access protocols*.

Multiple access (MA) protocols have traditionally been channel-oriented. The satellite resource available is subdivided into separate channels (with FDMA, TDMA or CDMA). The basic unit for allocation is thus a channel. Channels can be either (i) fixed assigned, or (ii) demand assigned to users. With demand assignment, a channel needs to be set aside for signaling among users. (Access to the signaling channel is another multiple access problem! A typical solution for this is TDMA with fixed assignment.) Demand assignment can then be accomplished with either a central controller or distributed control algorithm [PRIT 77, PUEN 71].

The channel-oriented MA protocols are suitable for voice traffic and may also be suitable for some data traffic. Data communications in general, however, have very diverse requirements ranging from inquiry-response systems with intermittent traffic to file transfers with large volumes of data. In addition, user-specified delay constraints need to be met. In this environment, an appropriate measure of traffic carrying capacity is no longer the number of (voice grade) channels, but instead the aggregate throughput rate in number of messages (or packets or bits) that can be transported per unit time while satisfying the specified delay constaints.

The problem of interest in this paper begins where the traditional satellite system designers leave off. A satellite channel of C bits/s is assumed to be available which may have been derived from an FDMA, TDMA or CDMA system at a higher level of satellite resource allocation. The satellite channel is to be shared by a population of users, each with his own earth station, for communication among themselves. The users have random traffic demands and delay constraints. The problems of modulation, clock synchronization, coding, random noise

0090-6778/79/1000-1456$00.75

etc. are assumed to have been solved already. Our emphasis is on packet-oriented multiple access protocols for sharing the channel.

The key measure of performance of a multiple access protocol is its channel throughput versus average delay tradeoff characteristic. The throughput of a channel is defined as follows. Let C be the channel transmission rate in bits per second and let there be on the average P bits in a transmitted block of data. The *channel throughput* S is defined to be the ratio of the rate of successfully transmitted data blocks to the rate C/P. Thus channel throughput is a normalized quantity between 0 and 1. It includes as useful throughput overhead bits contained in data blocks for synchronization, addressing, error control and other network control functions; these overheads, however, are not directly attributable to the MA protocol and are roughly the same for all packet-oriented protocols to be considered. Overheads that are directly attributable to the implementation of an MA protocol are accounted for in the calculation of channel throughput. Hence the maximum channel throughput is less than one and is of interest as a gross measure of performance.

We shall next define a model for characterizing data network users as traffic sources. Following that, the two major classes of packet-oriented MA protocols are reviewed: contention protocols and reservation protocols. Specific protocols from each class are described and their performance characteristics examined. Finally we compare the performance of channel-oriented protocols versus the two new classes of packet-oriented protocols under a variety of traffic assumptions; the strengths and weaknesses of these MA protocols are discussed.

Traffic Model

The following model will be used to represent the traffic characteristics and transmission requirements of users of a message- (or packet-) oriented communication network. Examples of users are: human operators of computer terminals, computer programs interacting with such human operators or with other programs/databases in other machines, data concentrators for a multiplicity of such traffic sources, etc. We shall view a user as simply a traffic source that can be modeled as a random point process with instants of message arrivals being the points of interest. A message is defined to be a block of data that has a time delay constraint associated with it for delivery to a destination user. (In a packet-oriented network, a message may be transported in one or more packets.) For simplicity, only *average* message delay constraints are considered in the following. The above definition is quite general. For example, a message may be a computer data file that needs to be delivered within a period of hours. It may be a line of characters in an inquiry-response system that needs to be delivered in a fraction of a second. It may be a digital voice sample (8 bits) that has a delay constraint dictated by the real-time voice sampling rate (e.g., 8000 samples per second).

Computer data traffic sources are often described as "bursty." Traffic burstiness is an important characteristic that influences the design of packet communication systems. This concept is briefly reviewed and a quantitative measure of burstiness, called *bursty factor*, as defined in [LAM 78a] is presented below.

The bursty nature of a data traffic source stems from more than just the randomness in message generation time and size. The user-specified message delay constraints to be met for these traffic sources are actually the single most important factor in determining if data traffic sources behave in a bursty manner. Suppose we are given a traffic source with

T = average interarrival time between messages

and

δ = average message delay constraint

where δ can be estimated in practice from the performance specifications of the intended network users. Generally, a user-specified source-destination delay constraint can be broken up into several parts, each part becoming a constraint for a segment of the communication path. δ is defined here to be the constraint for the message transmission time plus any necessary conflict resolution and queueing delays; it excludes, however, propagation delays through the network as well as message processing delays at the source and destination.

The bursty factor β of the traffic source is defined to be

$$\beta = \delta / T. \tag{1}$$

Note that β depends only upon δ and T which are inherent user characteristics. (The delay constraint δ is indeed an inherent source characteristic in the eyes of the network designer. Failing to satisfy it means that the user will take his business elsewhere!) Next consider a traffic source that is formed by merging together N sources with different statistics and delay constraints. The bursty factor of the aggregate source is defined to be the sum of the bursty factors of the individual sources, namely

$$\beta = \beta_1 + \beta_2 + \cdots + \beta_N. \tag{2}$$

The usefulness of β is due to the following observation [LAM 78a]. Suppose a communication channel is dedicated to a traffic source with bursty factor β and all delay constraints are met. Then the resulting ratio of peak to average channel data rates (PAR) satisfies

$$\mathrm{PAR} \geqslant 1/\beta \tag{3}$$

and the channel throughput S satisfies

$$S \leqslant \beta. \tag{4}$$

In other words, β gives an upper bound on the duty cycle of a traffic source. This information is useful to the network designer and is available independent of the communications system eventually provided. A traffic source with a small β ($\ll 1$) is said to be *bursty*.

The above result also says that for bursty users, channel-oriented MA protocols such as fixed assignment or demand assignment (over a period of time $\gg\delta$) are going to be very inefficient ($S \ll 1$). To improve the throughput of a broadcast channel shared by users with random bursty traffic, it is desirable to dynamically allocate transmission capacity on a per message (or packet) basis. This benefits from both the multiplexing effect of Eq. (2) as well as the statistical averaging effect of the law of large numbers [KLEI 76, section 5.1]. The key to realizing these gains is to design MA protocols for resolving channel access conflicts without excessive overhead.

We note that the definition of bursty factor in Eq. (1) does not involve the average length of messages generated by the traffic source. Thus it is possible for a traffic source to generate very long messages but is still considered to be very bursty by definition. This average message length parameter, we shall see, will also strongly impact the performance of MA protocols.

Network Assumptions

We consider a population of N users sharing use of a single satellite channel. The ith user has a message generation rate of λ_i messages per second. Each message may give rise to one or more fixed length packets with a mean number of L. Each packet carries a destination address so that when the packet is transmitted, with no interference from another user, over the satellite channel, it will be received by the proper addressee(s). The effect of errors due to channel noise is assumed to be negligible and not considered. Notice that the specific connectivity requirements of the population of users are relevant only indirectly through the resulting set of message rates. Given the set of λ_i, it does not matter, for example, whether users want to communicate with each other or they all want to talk to a specific central site. Thus, our only concern is the access problem of the satellite channel.

Each user is assumed to be capable of sending and receiving data at the satellite channel transmission rate of C bits/s. Problems of modulation, clock synchronization, coding and the like are assumed to have been solved already. In a number of MA protocols, the users are synchronized so that the channel can be viewed upon as a sequence of time slots (just as in TDMA). Each time slot can accommodate one data packet. Minislots may also be interleaved with the data slots to accommodate small control packets.

A broadcast network is assumed so that a user when given access to the satellite channel can send data to any other user. An MA protocol is thus simply an algorithm (possibly distributed as well as non-deterministic) for determining the access rights of the users. In some protocols, the access right is not uniquely determined and it is possible for packet transmissions from different users to "collide" in the channel. It is assumed that the collisions are always destructive and none of the packets involved in a collision can be correctly received. Each packet contains parity bits for error detection. After transmitting a packet, a user can find out the outcome of the transmission for himself by monitoring the downlink broadcast.

Multiple Access Protocols

The conflict resolution problem is nontrivial because users are geographically distributed. There are two problems involved: (i) to identify users with data to send (the *ready users*), and (ii) to schedule usage of the shared channel by these users. A consequence of the conservation law in queueing theory [KLEI 76] is that the (average) delay-throughput performance of an MA protocol is determined primarily by the time overhead required to identify a ready user, when one or more are present, and assign channel access to him. The identity of the ready user is unimportant. This can be accomplished by a polling protocol. Polling protocols are used extensively in computer communications for terrestrial multipoint networks [SCHW 77]; note that a multipoint network is also a broadcast network. The presence of a central controller is required. Users are passive in the sense that they may access the channel only when specifically polled by the central controller. Polling protocols, however, are not suitable for a satellite channel due to the long channel propagation time. They are also not suitable for a large population of very bursty users ($N \uparrow \infty$ while $\beta \downarrow 0$ for each user while $N\beta$ remains constant). The reader is referred to [LAM 79a] for a more detailed treatment of this subject.

We next consider two classes of protocols which require ready users to actively seek channel access instead of waiting to be polled. These are called contention and reservation protocols. Under contention protocols, there is no attempt to coordinate the ready users to avoid collisions entirely. Instead, each ready user makes his own decision regarding when to access the channel; he exercises caution, however, to minimize interference with other ready users as much as possible. Under reservation protocols, a reservation channel is provided for ready users to communicate among themselves such that only one ready user is scheduled for channel access at a time. Since users are geographically distributed, the multiple access problem has not really disappeared; it now exists in the access of the reservation channel for the transmission of reservation requests.

2. CONTENTION PROTOCOLS

Unlike polling protocols, the overhead incurred by contention protocols for assigning channel access to ready users is independent of N and channel propagation time, but is dependent upon the level of traffic. Thus, pure contention protocols are suitable for a large population of bursty users. Below, we shall describe two pure contention protocols (ALOHA and slotted ALOHA), a contention protocol which includes an element of reservation (R-ALOHA), and an adaptive protocol (URN). We shall also discuss, in the context of slotted ALOHA, the stability problem of contention protocols and the need for adaptive control in these protocols.

The ALOHA Protocol [ABRA 70, BIND 75a]

Under the ALOHA protocol, users are not synchronized in any way. Each user transmits a data packet whenever one is ready. In the event that two or more packets collide, i.e. overlap in time, each of the users involved realizes this after R seconds (the channel propagation time) and retransmits his packet after a randomized delay. As we shall discuss below, this randomized delay turns out to be crucial to the stability behavior and thus the throughput-delay performance of all contention-based protocols.

In [ABRA 70], Abramson first derived the maximum channel throughput of the ALOHA protocol in the limit of an infinite user population ($N \uparrow \infty$ and for each user $\beta \downarrow 0$). All messages consist of single packets. Hence, the aggregate packet "birth process" is a Poisson process at a rate of S packets per packet transmission time. Abramson also made the assumption that the sum of new transmissions and retransmissions in the channel (called *channel traffic*) can be approximated as a Poisson process at a rate of G packets per packet time. Furthermore, statistical equilibrium is assumed. The probability that a transmitted packet is successful is

$$\frac{S}{G} = e^{-2G} \tag{5}$$

which is obtained from consideration of Fig. 1; each transmitted packet has a vulnerable period of 2 packet times duration. It will be successful only if no other packet begins transmission within the vulnerable period.

From the above equation, the maximum possible ALOHA channel throughput is obtained at $G = 0.5$ and for an infinite user population model (under the above assumptions) is

$$C_A = \frac{1}{2e} \cong 0.184. \tag{6}$$

The Slotted ALOHA Protocol [ROBE 72, ABRA 73, KLEI 73]

The slotted ALOHA protocol is just like ALOHA with the additional requirement that the channel is slotted in time. Users are required to synchronize their packet transmissions into fixed length channel time slots. By requiring synchronization of packet start times, packet collisions due to partial overlaps are avoided and the vulnerable period of a transmitted packet is just the duration of a time slot. (See Fig. 1.) Under the same assumptions given above for ALOHA, we obtain

$$\frac{S}{G} = e^{-G} \tag{7}$$

where S is maximized at $G = 1$. The resulting slotted ALOHA maximum channel throughput for an infinite user population model is twice that of the unslotted case; thus

$$C_{SA} = \frac{1}{e} \cong 0.368. \tag{8}$$

Equations (5) and (7) are plotted in Fig. 2.

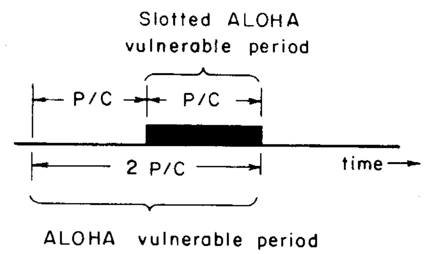


Figure 1 ALOHA and slotted ALOHA vulnerable periods of a transmitted packet.

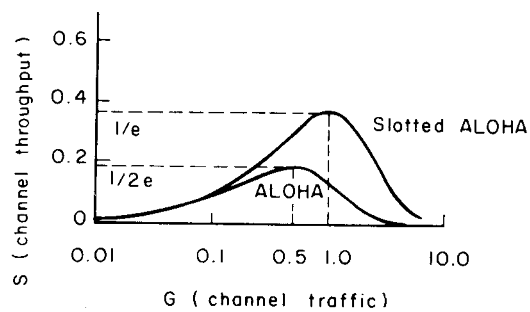


Figure 2 ALOHA and slotted ALOHA throughput curves.

Performance Considerations

The above analysis of ALOHA and slotted ALOHA was based upon three assumptions: (A1) statistical independence of channel traffic, (A2) infinite user population, and (A3) statistical equilibrium. More detailed studies of slotted ALOHA investigated the validity of these assumptions. In particular, it was shown that a necessary condition for (A1) is that the mean randomized delay $\bar{k}$ for retransmissions must be large ($\bar{k} \to \infty$). However, simulations showed that the maximum channel throughput results given by Eqs. (6) and (8) are robust; they are already quite accurate if $N \geqslant 10$ and $\bar{k} \geqslant 5$ [KLEI 73, LAM 74]. In these same references, the delay performance of slotted ALOHA was first studied for finite values of $\bar{k}$. When the traffic distribution is unbalanced with a mixture of high-rate as well as low-rate users, it was shown that the slotted ALOHA maximum throughput of $1/e$ can be considerably improved [ABRA 73].

The validity of assumption (A3) was also examined. It was shown that the slotted ALOHA protocol, without adaptive control, is potentially unstable [METC 73, LAM 74, KLEI 74, KLEI 75]. (We might have deduced this from the curves in Fig. 2 which show two equilibrium values of G for each value of S!) Statistical fluctuations may cause the channel to drift into a saturation state—the channel is filled up with collisions resulting in zero throughput. For an unstable channel, equilibrium conditions assumed earlier exist for a finite period of time before channel saturation occurs. A Markov chain formulation of the infinite population slotted ALOHA model shows that it is always unstable, in the sense that a stationary probability distribution does not exist [LAM 74, KLEI 75, FAYO 77].

Fortunately, N must be finite in a real network. In this case, slotted ALOHA channels may exhibit stable or unstable behavior depending upon the parameters N, $\bar{k}$ and S (channel input rate). In [LAM 74, KLEI 74, KLEI 75] a coherent theory of channel behavior is formulated and a method for characterizing stable and unstable channels is shown. A quantitative measure of instability for unstable channels is also introduced. A theoretical treatment of adaptive control of unstable channels using a Markov decision model is given in [LAM 75b]. Various heuristic control algorithms and their performance are presented in [LAM 75a]. Several effective feedback control algorithms are proposed in [GERL 77a, LAM 79b]. Simulations showed that these techniques are effective means of achieving stability (for initially unstable channels) at the expense of a small amount of delay-throughput performance degradation relative to lower bound values.

It has been observed that the delay-throughput performance of contention protocols is not entirely independent of N as we said earlier. When N is increased (accompanied by a decrease in β so that the input traffic level remains constant), slotted ALOHA channels tend to be less stable [LAM 74, KLEI 75], which impacts the delay-throughput performance. However, this is only a second order effect.

We have discussed the stability problem and adaptive control techniques within the context of slotted ALOHA, which is a pure contention protocol. However, the stability problem is present in all contention-based protocols (many to be described in this section as well as the next section on reservation protocols) and must not be forgotten.*

The R-ALOHA Protocol [CROW 73, LAM 78b]

The traffic environment suitable for ALOHA and slotted ALOHA is that of a large population of low-rate bursty users with short messages (one packet per message). The R-ALOHA protocol was originally proposed by Crowther et al. [CROW 73] and is suitable for users who generate long multipacket messages or users with steady input traffic and queueing capability (see below).

The R-ALOHA protocol requires, in addition to time slotting, that time slots are organized into frames. Time slots are identified by their positions in the frame. The duration of a frame must be greater than the maximum channel propagation time between any two users in the network. Consequently each user is aware of the usage status of time slots in the previous frame. The network operates without any central control but requires each user to execute the same set of rules for transmitting packets into a time slot depending upon the outcome in the same time slot of the previous frame.

A time slot in the previous frame is *unused* if it was empty or contained a collision. Slots unused in the previous frame are available for contention by ready users in exactly the same manner as slotted ALOHA. A slot which had a successful transmission by a user in the previous frame is *used* and is reserved for the same user in the current frame. As a result, such a user now has the equivalent of an assigned TDMA channel for as long as he has traffic to send in it.

* Recently, a new method for resolving contention following a collision was proposed. It uses a tree algorithm for scheduling retransmissions and was shown to give rise to a stable channel [CAPE 77].

We further differentiate between two slightly different protocols depending upon whether an end-of-use flag is included in the header of the last packet before a user gives up his reserved shot:

(P1) end-of-use flag not included, and

(P2) end-of-use flag included.

Under (P1), a time slot is always wasted when a user gives up his reserved slot. The tradeoff for adopting (P2) is additional packet processing overhead by each user.

Under the assumption of equilibrium conditions, the channel throughput S_{RA} of R-ALOHA can be expressed in terms of the slotted ALOHA throughput S_{SA} for the contention portion of the channel [LAM 78b]:

$$S_{RA} = \frac{S_{SA}}{S_{SA} + \frac{1}{\bar{v}}} \qquad \text{under (P1)}$$

or

$$S_{RA} = \frac{S_{SA}}{S_{SA} + \frac{1 - S_{SA}}{\bar{v}}} \qquad \text{under (P2)} \tag{9}$$

where $\bar{v}$ = average number of packets that a user transmits before he gives up a reserved slot.

Note that S_{RA} is a monotonic function of S_{SA} so that the maximum channel throughput of R-ALOHA is

$$C_{RA} = \frac{C_{SA}}{C_{SA} + \frac{1}{\bar{v}}} \qquad \text{under (P1)}$$

or

$$C_{RA} = \frac{C_{SA}}{C_{SA} + \frac{1 - C_{SA}}{\bar{v}}} \qquad \text{under (P2).}$$

Assuming that $C_{SA} = 1/e$, C_{RA} is plotted in Fig. 3 as a function of $\bar{v}$. Note that $\bar{v}$ ranges from 1 to infinity; thus, we have

$$\frac{1}{1+e} \leqslant C_{RA} \leqslant 1 \qquad \text{under (P1)}$$

or

$$\frac{1}{e} \leqslant C_{RA} \leqslant 1 \qquad \text{under (P2).}$$

Notice that the R-ALOHA protocol adapts itself to the nature of the traffic. At one extreme, it behaves like slotted ALOHA when the users are bursty ($\bar{v} = 1$). At the other extreme, the channel capacity of R-ALOHA approaches one in the $\bar{v} \to \infty$ limit; this is the case for high-rate users who can accommodate very long queues. The reader is referred to

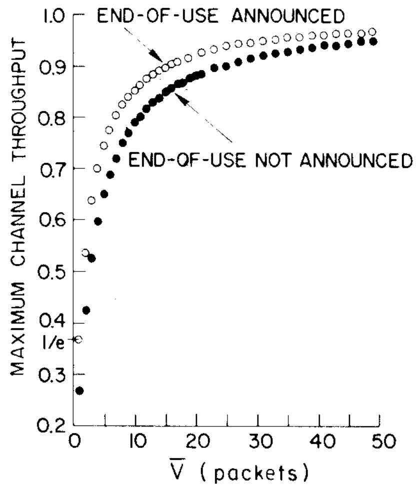


Figure 3 Maximum channel throughput versus $\bar{v}$ for R-ALOHA.

[LAM 78b, LAM 79b] for a queueing and message delay analysis of R-ALOHA as well as simulation results.

The URN Protocol [KLEI 78, YEMI 78]

The slotted ALOHA protocol is an important component of the R-ALOHA protocol. We also learned earlier that the slotted ALOHA protocol needs to be adaptively controlled. Recall that the slotted ALOHA channel throughput rate S in Eq. (7) is maximized when the channel traffic rate G is one. The goal of most adaptive control algorithms is to achieve the $G = 1$ condition in the channel. The main difficulty is that since users are distributed, the number n of ready users at any time is generally not known to individual users.

Let us assume for the moment that n is in fact known to individual users instantaneously. (We shall discuss how to estimate n later.) Then an adaptive strategy for achieving the $G = 1$ condition is to have each ready user transmit into the next time slot with probability $1/n$.

Kleinrock and Yemini [KLEI 78] proposed an alternative "pure" strategy for ready users to determine whether or not to transmit in the next time slot: the probability of transmission is either 1 or 0. In other words, some users have full channel access right while others have none. (Consequently, the URN protocol is said to be *asymmetric*.) A user who has channel access right and is also ready transmits into the next time slot. It is possible to prove that optimal strategies are always pure strategies and therefore asymmetric [YEMI 78].

The URN protocol is described using the following URN model. Consider each user as a colored ball in an urn: black for ready, white for not ready. The access protocol is essentially a rule to sample balls from the urn. Let k be the number of balls drawn from the urn. The probability of a successful transmission (throughput) is that of getting exactly one black ball in the sample. This probability is

$$\text{Prob [throughput]} = \frac{\binom{n}{1}\binom{N-n}{k-1}}{\binom{N}{k}} \tag{10}$$

where N is the number of balls, n is the number of black balls. The above expression is maximized when $k = \lfloor N/n \rfloor$ where $\lfloor x \rfloor$ gives the integer part of x. Not only does this value of k maximize the probability of selecting exactly one black ball, but it also gives that the average number of black balls selected is equal to one (i.e. $G = 1$).

The URN protocol adapts smoothly to network load fluctuations. When the network is lightly loaded, a large number of users get channel access rights. For instance, $n = 1$ gives rise to $k = N$; all users get access rights, but only one (the lone ready user) is going to make use of it. As the network load increases, n increases and the number k of users getting access rights is reduced. When $n > N/2$ then $k = 1$ and the URN protocol becomes effectively a TDMA protocol (which is most suitable for a heavy load). The maximum channel throughput of URN is thus unity.

Two questions arise in the implementation of the URN protocol: How does an individual user obtain the up-to-date value of n? How does the protocol obtain coordination of the distributed decisions of individual users?

A solution for estimating n with high accuracy at the expense of a small overhead is proposed in the references. Briefly, it consists of a binary erasure reservation subchannel. An idle user who becomes ready (n increases by 1) sends a message of a few bits in the subchannel. When a ready user turns idle (n decreases by 1), the condition is detected by other users from examining his last packet or its positive acknowledgment. An erasure (collision) in the subchannel means two or more users become ready in the same time slot. In this case, the increase in n is assumed to be two (an approximation). The resulting error in the estimate of n was found to be negligible since the probability of more than two users becoming ready in the same time slot is very small. Furthermore, the estimate of n is corrected every time the network goes idle ($n = 0$). Other heuristic algorithms for estimating n are studied in [LAM 74, LAM 75a, LAM 75b].

Implementation of the URN protocol also must insure that individual users agree upon k the number of users with access rights as well as their identity. The optimal $k = \lfloor N/n \rfloor$ to be used is determined by each user from n estimated as described above. The selection of which k users should get access rights may be achieved via identical pseudorandom number generators at individual users, or via a window mechanism as well as other methods. The reader should consult the references for details.

Concluding Remark

In this section, we began with pure contention protocols (ALOHA, slotted ALOHA) and then moved on to more sophisticated contention-based protocols (R-ALOHA, URN) with improved delay-throughput performance. These protocols all have distributed control. Each user makes his own decision regarding channel access based solely upon observable outcomes in the broadcast channel. In the URN protocol implementation, however, a reservation subchannel is provided for users to communicate with each other in a limited fashion. In the next section, we describe reservation protocols that require users to cooperate with each other to avoid collisions entirely through use of a reservation subchannel.

3. RESERVATION PROTOCOLS

The objective of reservation protocols is to avoid collisions entirely. Since users are distributed, a reservation subchannel is necessary for users to communicate with each other. There are two key problems to be solved for most reservation protocols: (1) implementation of the reservation subchannel, and (2) implementation of a distributed global queue.

The Reservation Channel

The original satellite channel can be either time or frequency multiplexed into a reservation channel and a data channel. An advantage of time multiplexed channels is that the partition may be variable. With a variable partition, the maximum channel throughput of data messages can be made very close to one under a heavy network load when users have long queues. With a fixed partition, the overhead lost to the reservation channel is a fixed fraction of the original channel capacity.

With a global queue, there is no conflict in the use of the data channel. However, the multiple access problem of the distributed users has not disappeared. It exists now in the access of the reservation channel. Any of the previously described multiple access protocols can be used. However, for simplicity, most proposed reservation protocols adopt either a fixed assigned TDMA protocol or some version of the slotted ALOHA protocol. We are faced with the same tradeoff as before. A TDMA protocol performs poorly for a large population of bursty users. On the other hand, a slotted ALOHA protocol is independent of N but needs to be adaptively controlled for stable operation.

The high achievable channel throughput of reservation protocols (relative to pure contention) comes about from substantially reducing the volume of traffic requiring conflict resolution, and the concomitant overhead, from the totality of data messages to just one short reservation packet per data message (or less). Note that R-ALOHA described earlier derives its efficiency from essentially the same principle.

Part of the price that one pays for the gain in channel throughput of reservation protocols over contention protocols is an increase in message delay. The minimum delay incurred by a message, excluding message transmission time, is at least twice the satellite channel propagation time. The minimum delay can be reduced, however, if one can anticipate future arrivals and make reservations in advance! One possible example is packetized digital speech traffic.

The Distributed Global Queue

There are two approaches to implement a global queue of requests for a population of distributed users. One is to employ a central controller which tells the ready users when to access the channel; an additional subchannel for controller-to-user traffic is typically required. On the other hand, a distributed control implementation is more interesting and probably more desirable. In this approach, each user maintains information on the status of the global queue and makes his own decision on when is his turn to access the channel. An important problem here is the synchronization of queue status information of all users. This means that reservation packets broadcasted in the reservation channel need to be received correctly by *all* users. In the event of an error, an individual user must be able to detect the presence of error in his queue status information. Any such user who is out of synchronization as well as new users who have just joined the network must be able to acquire *queue synchronization* from observing the reservation and data channels within a reasonable duration of time.

Various queue disciplines may be used for the global queue, e.g., random selection, first-in-first-out (FIFO), round-robin, priority based upon type or delay constraint etc. The processing requirement of a sophisticated scheduling algorithm may be quite substantial.

A Protocol with Distributed Control [ROBE 73]

The ideas of a reservation subchannel and a distributed global queue were first proposed by Roberts [ROBE 73]. In his proposal, it is assumed that the channel is time slotted and time slots are organized into frames. Each frame consists of a data subframe and a reservation subframe; each slot in the reservation subframe is further subdivided into V smaller slots. The small slots are for reservation packets (as well as possibly positive acknowledgment packets and small data packets) to be used on a contention basis with the slotted ALOHA protocol.

Roberts' protocol makes use of the broadcast capability of the channel; a reservation packet successfully transmitted with no interference is received by all users. Each reservation request is for a position in the global queue for a group of packets. The queue discipline proposed by Roberts is FIFO according to the order reservation requests are received. (We shall refer to this as the *FIFO protocol*.) Each user maintains his copy of the queue status information. It is sufficient for each user to record only the queue length (in number of packets) as well as the queue positions of his own reservations.

For a currently inactive user who wants to join the queue, it is necessary for him to acquire queue synchronization. In this protocol, the queue length information may be supplied in the header of each data packet transmitted. Alternatively, it may be announced periodically by a "master" controller. Note that such queue length information is one propagation delay old when received. To acquire queue synchronization, a user must update the queue length information so received with reservation requests received within a channel propagation time just prior to receiving the queue length information.

To maintain synchronization among the users, it is necessary and sufficient that each reservation packet that is received correctly by any user is correctly received by all users. This condition may be assured by properly encoding the reservation requests. A simple strategy proposed by Roberts is to send parity-checked copies of requests in triplicate within a reservation packet.

As a result of the distributed nature of queue management, the impact of an error in a user's queue status is to cause some collisions in data slots and to delay some data packets momentarily. However, no catastrophic failure occurs. Users involved in such collisions must declare themselves to be out of synchronization. A user who receives a reservation packet with unrecoverable error must also do the same. Users who are out of synchronization discard their acquired reservations and reacquire queue synchronization in the same fashion as newly activated users described above.

Maximum Channel Throughput

The maximum channel throughput of a reservation protocol is $1 - \gamma$, where γ is the minimum fraction of the original channel capacity needed to accommodate the reservation request traffic. Consider the FIFO protocol above. Let L be the average number of data packets per reservation request. Since there are V small slots in a data slot, the ratio of data bits to reservation request bits is equal to

$$\bar{v} = VL.$$

Assume that the reservation channel is used by reservation packets only (excluding acknowledgment and small data packets mentioned above) and its maximum channel throughput is C_{SA} under the slotted ALOHA protocol. It can be easily shown that in this case

$$\gamma = \frac{1}{1 + C_{SA}\bar{v}}. \tag{11}$$

The maximum channel throughput of the reservation protocol is $1 - \gamma$. We make several observations. First, the maximum channel throughput expression is the same as that for R-ALOHA with just a slightly different interpretation for $\bar{v}$. Second, it is independent of N. Third, $\bar{v}$ is assumed to be a constant parameter; thus, the maximum channel throughput is fixed and strictly less than one.

Now suppose the partition between data and reservation slots in a frame can be dynamically varied. Also, reservation requests can be piggybacked in the header of scheduled data packets. In this case, when the network is heavily loaded with long message queues at individual users, $\bar{v}$ will be very large; the maximum channel throughput becomes one in the $\bar{v} \to \infty$ limit.

Alternatively, if the frame is fixed partitioned and a fixed assigned TDMA protocol is used for the reservation subchannel, we have

$$\gamma = \frac{N}{MV} \tag{12}$$

where M is the duration of a frame (in number of data slots).

A fixed assigned TDMA protocol is sometimes preferable to slotted ALOHA for the reservation channel since it is simple to implement; unlike slotted ALOHA adaptive control is not needed. However, Eq. (12) shows that it is applicable only for a small user population. One can always increase M to increase channel throughput; a large M is undesirable, however, since the delay of reservation packets becomes large and hence the message delay as well.

The maximum channel throughput $1 - \gamma$ is plotted in Fig. 4 versus N for the two cases given by Eqs. (11) and (12).

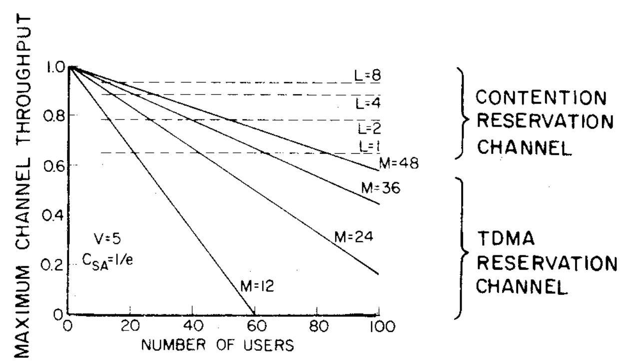


Figure 4 Maximum channel throughput versus number of users for reservation protocols.

Other Protocols with Distributed Control

Another reservation protocol with distributed control was proposed by Binder [BIND 75b]. The channel is divided into time slots which are organized into frames. Let M be the number of slots in a frame. It is assumed that the frame duration is larger than the channel propagation time. Also, the number N of users is required to be less than or equal to M. Each user is fixed assigned a time slot within the frame and sends information concerning his current queue length in the header of each data packet that he transmits into his fixed assigned slot. Note that this is equivalent to a fixed assigned TDMA reservation subchannel.

The global queue is implemented via distributed control. The global queue status consists of the queue lengths of all ready users (those with nonempty queues). Each user exercises the following rules for channel access. A user may send data in his fixed assigned slot at any time. Any unassigned as well as unused slots (assigned to currently idle users) within a frame are used by the ready users in a round-robin fashion. A user who has been idle can transmit in his fixed assigned (owned) slot to deliberately generate a collision. Such a collision is noted by all users. Another rule dictates that following a collision, only the owner of the slot can use it in the next frame. Thus a special feature of this protocol is that even if a user is in the process of acquiring queue synchronization, he still has the use of a fixed assigned TDMA channel. A disadvantage is that the number of users N must be less than M. The problems of maintaining and acquiring queue synchronization are similar to those of the FIFO protocol above.

Recently the CPODA protocol was proposed and implemented in SATNET, a prototype packet broadcast satellite network [JACO 77, JACO 78]. CPODA is essentially an outgrowth of Roberts' FIFO protocol with a more sophisticated scheduling algorithm designed to handle both packetized data and voice traffic. Like the FIFO protocol, each frame consists of a reservation and an information subframe with a slotted ALOHA protocol for the reservation subframe. In addition to using the reservation subframe, a reservation request can also be piggybacked into the header of a scheduled data packet. While the total frame size is fixed, the reservation subframe is allowed to grow or shrink according to network loading. If the global queue is empty, the reservation subframe occupies the entire frame. (This was also permitted in [ROBE 73].) Two types of traffic are distinguished when reservations are made: block message traffic and stream message traffic. An explicit reservation is sent for each block message, while a reservation is sent only once for all messages of a particular stream. In the latter case, after the initial reservation, a reservation is automatically placed into the global queue at predetermined intervals. The scheduling discipline of the global queue is quite elaborate and depends upon both explicit priority classes and delay constraints. Some measurement and simulation performance results of CPODA are reported in [GERL 77b, CHU 78].

Protocols with Centralized Control

The presence of a central controller eliminates the queue synchronization problem discussed earlier for distributed control protocols. Instead, the central controller manages the distributed global queue, accepting reservations and informing users of when to access the channel. An additional subchannel is typically required for controller-to-user traffic. The multiple access problem of the reservation subchannel remains the same as above.

A protocol with centralized control has been proposed and studied by Ng and Mark for a satellite network with on-board processing that serves as the central controller [NG 77].

4. PERFORMANCE COMPARISONS

Our focus in this paper is mainly on the sharing and conflict resolution aspect of the multiple access problem. A broadcast channel of C bits/s is assumed. The performance criteria of interest are channel throughput and average message delay. Even within our limited scope of performance it should be clear to the reader by now that there is no single protocol that is optimum. The performance of a multiple access protocol is strongly dependent upon the traffic model and network loading. In general, some traffic characteristics do favor one class of protocols more than others. We list some of them below.

Traffic model	Multiple access protocols favored
nonbursty users	fixed assigned channels (TDMA, FDMA)
bursty users, short messages	pure contention
bursty users, long messages, large N	reservation protocols with contention reservation channel
bursty users, long messages, small N	reservation protocols with fixed TDMA reservation channel

For traffic models which are a fixed or time-varying combination of the above models, "mixed" protocols (such as R-ALOHA) and adaptive protocols (such as URN) may be suitable. To illustrate the above observations, we show the delay-throughput performance of representatives of the different classes of protocols for given specific traffic assumptions.

In Figures 5-7, four protocols are considered: (1) fixed assigned TDMA channels, (2) slotted ALOHA, (3) FIFO with a fixed TDMA reservation channel, and (4) FIFO with a slotted ALOHA reservation channel; they are labeled as TDMA, ALOHA, FIFO and FIFO* respectively in the figures.

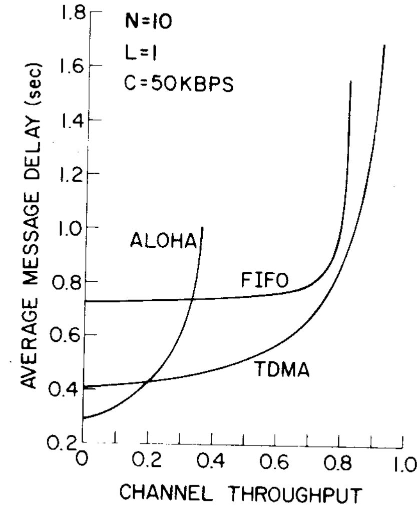


Figure 5 Delay-throughput tradeoff for 10 users and short messages (one packet per message).

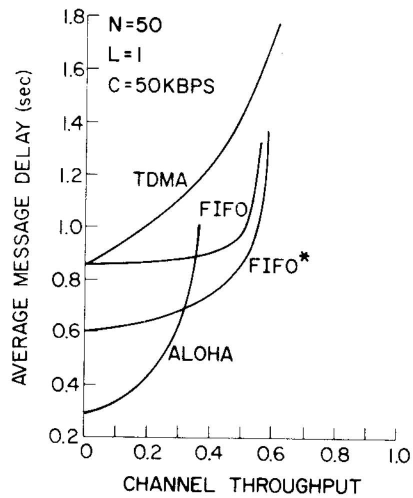


Figure 6 Delay-throughput tradeoff for 50 users and short messages (one packet per message).

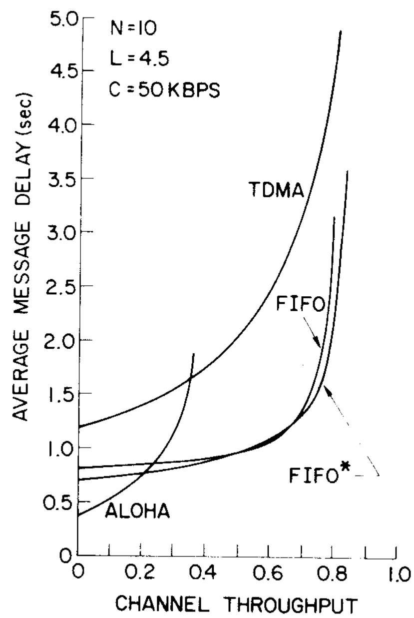


Figure 7 Delay-throughput tradeoff for 10 users and multipacket messages.

Delay formulas for TDMA and slotted ALOHA from [LAM 77b, KLEI 73] are used. The expected delay of a message for the reservation protocols is taken to be the sum of the expected delay D_1 incurred by the reservation request and the expected delay D_2 incurred by the message itself after the reservation has been made. To calculate D_1 and D_2, we shall regard the original broadcast channel at C bits/s to be split up into two separate channels: a data channel at $(1 - \gamma)C$ bits/s and a reservation channel at γC bits/s. D_1 and D_2 are then calculated separately.

In Fig. 5, we show results obtained assuming 10 users sharing a 50 kbit/s satellite channel with a propagation delay of 0.27 s. All messages consist of single packets of 1125 bits each. In the case of FIFO, a frame size $M = 12$ is assumed with $V = 5$ and $\gamma = 1/6$. Observe that the performance of fixed assigned TDMA is the best for this traffic model except when the channel throughput is less than 0.2 where slotted ALOHA gives a smaller delay.

Now suppose we consider a larger population of more bursty users than the above traffic model. Let N be increased from 10 to 50 (so that the bursty factor of each user is now 1/5 of that of the above model). Fig. 6 shows that TDMA has the worst performance for this new traffic model. FIFO, with $M = 24$ and $\gamma = 5/12$, also has poor performance due to the large reservation channel overhead needed for 50 users. FIFO*, with $\gamma = 0.4$ assuming $C_{SA} = 0.3$, has better performance than FIFO, but the reservation channel overhead is still quite large since each message consists of a single packet only. The delay-throughput performance of slotted ALOHA is independent of the change from the first to the second traffic model and appears to be the most suitable protocol for the second traffic model, provided a channel throughput of about 0.3 or less is acceptable. If a channel throughput of more than 0.3 is desired, then FIFO* should be employed.

Now suppose we are back to having 10 users but the data traffic consists of long messages (8 packets each) as well as short messages (one packet each) in equal number. The average message length L is increased from 1 to 4.5 packets. Delay-throughput results for the four protocols are shown in Fig. 7. The delay curve for slotted ALOHA is plotted using the delay formula for multipacket messages in [LAM 74]. For this traffic model, the reservation protocols have the best performance except at a channel throughput of less than 0.2 where slotted ALOHA is better. Slotted ALOHA, as we know, was not designed to take advantage of the presence of multipacket messages. Simulations also showed that when a large number of multipacket messages are present in the input traffic, the slotted ALOHA channel becomes more unstable [LAM 74].

5. CONCLUSION

Satellite systems have traditionally been designed for voice traffic with channel-oriented multiple access protocols. Data communications (including possibly digital voice) have much more diverse traffic characteristics and transmission requirements than voice communications. In this paper, we have examined two major classes of packet-oriented multiple access protocols: contention protocols and reservation protocols. We have been primarily interested in the conflict resolution problem of a single broadcast channel shared by a population of distributed users. Our problem starts where traditional satellite system designers leave off. Problems of modulation, coding and the like are assumed to have been solved. A traffic model was defined that includes user-specified message delay constraints. The key measure of performance of a multiple access protocol is its channel throughput for given delay constraints and message traffic characteristics. The performance of channel-oriented protocols as well as the packet-oriented contention and reservation protocols were compared under different traffic assumptions. Traffic characteristics that tend to favor one class of protocols over others were illustrated.

In interpreting our performance comparison results, the reader is cautioned to bear in mind that we have used a fairly limited measure of performance. Maximizing the channel throughput is equivalent to minimizing the channel cost (only) per unit of data transported. However, many other performance variables have been neglected; for instance, the cost of earth stations has been completely ignored in our discussions above. With the current cost (say, one to three hundred thousand dollars) and size (say, 5 m antennas) of earth stations, it would not be economical to install a separate earth station at a packet network node unless it has a substantial amount of traffic (say, 100 kbits/s). At such a traffic level, traditional channel-oriented multiple access protocols will perform very well; there would be no need for the more complex packet-oriented protocols. However, the packet-oriented protocols considered in this paper will be extremely important in future satellite systems with small (say, 1 m) antennas that can be sited almost anywhere for low-rate users. This scenario will most likely materialize with the advant of satellite systems at the higher frequencies of 14/12, 30/20 GHz, etc. [ABRA 75, PRIT 77, ROBE 78, JACO 78].

ACKNOWLEDGMENT

The author thanks one of the anonymous reviewers for his helpful comments.

REFERENCES

ABRA 70 Abramson, N., "THE ALOHA SYSTEM–Another alternative for computer communications," *AFIPS Conf. Proc.*, Vol. 37, AFIPS Press, Montvale, N.J., 1970, pp. 281-285.

ABRA 73 Abramson, N. "Packet switching with satellites," *AFIPS Conf. Proc.*, Vol. 42, AFIPS Press, Montvale, New Jersey, 1973, pp. 695-702.

ABRA 75 Abramson, N. and E. R. Cacciamani, "Satellites: not just a big cable in the sky," *IEEE Spectrum*, Vol. 12, Sept. 1975.

BIND 75a Binder, R. et al., "ALOHA packet broadcasting–A retrospect," National Computer Conference, *AFIPS Conf. Proc.*, Vol. 44, 1975, pp. 203-215.

BIND 75b Binder, R., "A dynamic packet-switching system for satellite broadcast channels," *ICC 75 Conf. Rec.*, IEEE, New York, 1975, pp. 41-1 to 41-5.

BUTT 74 Butterfield, S. C., R. D. Rettberg, and D. C. Walden, "The satellite IMP for the ARPA network," *Proc. of the Seventh Hawaii International Conf. on System Sciences*, Computer Nets Supplement, Jan. 1974.

CAPE 77 Capetanakis, J. I., "The multiple access broadcast channel: Protocol and capacity considerations," Ph.D. Thesis, Dep. of Electrical Engineering, MIT, August 1977.

CHU 78 Chu, W. W. and W. E. Naylor, "Measurement and simulation results of C-PODA protocol performance," *NTC 78 Conf. Rec.*, National Telecommunications Conference, Birmingham, Alabama, Dec. 1978.

CROW 73 Crowther, W. et al., "A system for broadcast communication: Reservation-ALOHA," *Proc. 6th HICSS,* University of Hawaii, Honolulu, January 1973.

FAYO 77 Fayolle, G. et al., "Stability and optimal control of the packet switching broadcast channel," *J.ACM,* Vol. 24, July 1977.

GERL 77a Gerla, M. and L. Kleinrock, "Closed loop stability controls for S-ALOHA satellite communication," *Proc. Fifth Data Communications Symposium,* Snowbird, Utah, September 1977.

GERL 77b Gerla, M., L. Nelson and L. Kleinrock, "Packet satellite multiple access: models and measurements," *Conf. Rec. NTC,* Los Angeles, December 1977.

GITM 78 Gitman, I. and H. Frank, "Economic analysis of integrated voice and data networks," *Proc. IEEE,* Vol. 66, Nov. 1978.

JACO 77 Jacobs, I. et al., "CPODA–A demand assignment protocol for Satnet," *Proc. Fifth Data Communications Symposium,* Snowbird, Utah, Sept. 27-29, 1977.

JACO 78 Jacobs, I. M., R. Binder and E. V. Hoversten, "General purpose packet satellite networks," *Proc. IEEE,* Vol. 66, Nov. 1978

KELL 78 Kelly, P. T. F., "Public packet switched data networks, international plans and standards," *Proc. IEEE,* Vol. 66, Nov., 1978.

KLEI 73 Kleinrock, L. and S. S. Lam, "Packet-switching in a slotted satellite channel," *AFIPS Conf. Proc.,* Vol. 42, AFIPS Press, Montvale, N. J., 1973, pp. 703-710.

KLEI 74 Kleinrock, L. and S. S. Lam, "On stability of packet switching in a random multi-access broadcast channel," 7th Hawaii Int. Conf. on Systems Sciences, Honolulu, Jan. 1974, *Proc. Special Subconf. on Computer Nets,* 1974.

KLEI 75 Kleinrock, L. and S. S. Lam, "Packet switching in a multiaccess Broadcast channel: performance evaluation," *IEEE Trans. on Commun.,* Vol. COM-23, April 1975, pp. 410-423.

KLEI 76 Kleinrock, L., *Queueing Systems Vol. 2: Computer Applications,* Wiley-Interscience, New York, 1976.

KLEI 78 Kleinrock, L. and Y. Yemini, "An optimal adaptive scheme for multiple access broadcast communication," *Conf. Rec. ICC'78,* Toronto, June 1978.

LAM 74 Lam, S. S., "Packet switching in a multi-access broadcast channel with application to satellite communication in a computer network," Ph.D. Dissertation, Computer Science Department, Univ. of California, Los Angeles, March 1974.

LAM 75a Lam, S. S. and L. Kleinrock, "Packet switching in a multiaccess broadcast channel: dynamic control procedures," *IEEE Trans. on Commun.,* Vol. COM-23, Sept. 1975, pp. 891-904.

LAM 75b Lam, S. S. and L. Kleinrock, "Dynamic control schemes for a packet switched multi-access broadcast channel," *AFIPS Conf. Proc.,* Vol. 44, AFIPS Press, Montvale, N.J. 1975, pp. 143-153.

LAM 77a Lam, S. S., "Satellite multiaccess schemes for data traffic," *Conference Record of the International Conference on Communications,* Chicago, June 1977, pp. 37.1-19 to 37.1-24.

LAM 77b Lam, S. S., "Delay Analysis of a time division multiple access (TDMA) channel," *IEEE Trans. on Commun.,* Vol. COM-25, Dec. 1977, pp. 1489-1494.

LAM 78a Lam, S. S., "A new measure for characterizing data traffic," *IEEE Trans. on Commun.,* Vol. COM-26, Jan. 1978.

LAM 78b Lam, S. S., "An analysis of the R-ALOHA protocol for satellite packet switching," *Conf. Rec. ICC78, International Conf. on Communications,* Toronto, June 1978.

LAM 79a Lam, S. S., "Multiple access protocols," Department of Computer Sciences, University of Texas at Austin, Technical Report TR-88, Jan. 1979; to appear in *Computer Communication: State of the Art and Direction for the Future,* edited by W. Chou, Prentice-Hall.

LAM 79b Lam, S. S., "On protocols for satellite packet switching," *Conf. Rec. Int. Conf. on Communications,* Boston, June 1979.

METC 73 Metcalfe, R. M., "Steady-state analysis of a slotted and controlled ALOHA system with blocking," *Proc. 6th Hawaii Int. Conf. on System Sciences,* Honolulu, Jan. 1973.

NG 77 Ng, S. F. W. and J. W. Mark, "A multiaccess model for packet switching with a satellite having some processing capability," *IEEE Trans. on Commun.,* Vol. COM-25, Jan. 1977, pp. 128-135.

PRIT 77 Pritchard, W. L., "Satellite communication–An overview of the problems and programs," *Proc. IEEE,* Vol. 65, March 1977.

PUEN 71 Puente, J. G., W. G. Schmidt and A. M. Werth, "Multiple-access techniques for commercial satellites," *Proc. IEEE,* Vol. 59, Feb. 1971.

ROBE 72 Roberts, L. G., "ALOHA packet system with and without slots and capture," ARPANET Satellite System Note 8, (NIC 11290), June 1972; reprinted in *Computer Comm. Review,* Vol. 5, April 1975.

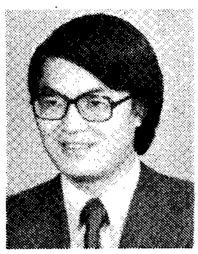
ROBE 73 Roberts, L., "Dynamic allocation of satellite capacity through packet reservation," *AFIPS Conf. Proc.,* Vol. 42, AFIPS Press, Montvale, New Jersey, 1973, pp. 711-716.

ROBE 78 Roberts, L., "The evolution of packet switching," *Proc. IEEE,* Vol. 66, Nov. 1978.

SCHW 77 Schwartz, M., *Computer-Communication Network Design and Analysis,* Prentice-Hall, Englewood Cliffs, N.J., 1977.

YEMI 78 Yemini, Y., "On channel sharing in discrete-time packet switched, multiaccess broadcast communication," Ph.D. Dissertation, Computer Science Department, University of California, Los Angeles, Calif. 1978.

★



Simon S. Lam (S'69–M'69–S'71–M'74) was born in Macao on July 31, 1947. He received the B.S.E.E. degree (with Distinction) in electrical engineering from Washington State University, Pullman, Washington, in 1969, and the M.S. and Ph.D. degrees in engineering from the University of California at Los Angeles, in 1970 and 1974, respectively.

From 1972 to 1974, he was a postgraduate research engineer with the ARPA Network project at UCLA and did research on satellite packet communication. From 1974 to 1977, he was a research staff member with the IBM Thomas J. Watson Research Center, Yorktown Heights, New York where he worked on the performance analysis of various aspects of packet switching networks, SNA and satellite networks, as well as queueing network theory. Since September 1977, he has been with the University of Texas at Austin where he is now Associate Professor of Computer Sciences. His current research interests include computer systems modeling and analysis, computer-communication networks and distributed systems.

At the University of California at Los Angeles, he held a Phi Kappa Phi Fellowship from 1969 to 1970, and a Chancellor's Teaching Fellowship from 1969 to 1973. In 1975, he received the Leonard G. Abraham Award for the best paper of the year in the field of Communication Systems published in the IEEE TRANSACTIONS ON COMMUNICATIONS. Dr. Lam is a member of Tau Beta Pi, Sigma Tau, Phi Kappa Phi, Pi Mu Epsilon and the Association for Computing Machinery.