Judicious Use of Redundant Transmissions in Multichannel ALOHA Networks with Deadlines

Yitzhak Birk, *Member, IEEE,* and Yaron Keren

***Abstract*—This paper shows how to improve the classic multichannel slotted ALOHA protocols by judiciously using redundant transmissions. The focus is on user-oriented requirements: a deadline along with a permissible probability of failing to meet it. Subject to satisfying those, maximization of capacity is the optimization goal. When there is no success/failure feedback prior to the deadline, the use of information dispersal with some redundancy provided by error-correcting codes for the data in conjunction with a replicated, separately transmitted synchronization preamble, is proposed. It is shown to sharply reduce the overhead resulting from the use of shorter packets and to significantly increase capacity. When the deadline permits several transmission–feedback rounds, we propose a novel replication-based retransmission policy: all attempts except the final one entail the transmission of a single or very few copies, and a larger number of copies are transmitted in the final attempt. This sharply increases channel capacity, even with a single transmitter per station. The proposed approaches are particularly suitable for high-bandwidth satellites with on-board processing.**

***Index Terms*— Deadline scheduling, dispersity routing, information dispersal, multichannel ALOHA, redundancy, VSAT.**

I. Introduction

The slotted ALOHA access scheme [1] gained popularity due to its simple implementation and random-access nature. Each station transmits a packet of data when it becomes available (aligned to a common time clock) on a multiple-access channel. Should another station transmit in the same time slot, both transmissions would not be received correctly. After such failure, some randomization takes place according to a retransmission policy [2] in order to avoid a definite repeated collision, and another transmission is attempted. Attempts repeat until the transmission is received successfully at its destination. With a single channel, temporal randomization is the only choice. With multiple channels, in contrast, the choice of channel is the primary avenue for randomization, as this permits immediate retransmission following a collision.

Most of the work on ALOHA has been carried out for single-channel systems, with a focus on capacity and sometimes on the interplay between throughput and mean delay. In practice, however, a communication system is often the provider of a service, whose quality is specified by the users. This may, for example, include a maximum permissible delay (deadline) along with a maximum permissible probability of exceeding it. The optimization goal of the system designer may be to maximize communication capacity while meeting the quality-of-service requirements. Our focus is on this situation in multichannel systems.

The main contribution of this paper is new and optimized schemes for the judicious use of redundancy in order to improve performance as just defined. This entails the choice of the proper degree of redundancy, the timing of redundant transmissions and, optionally, the use of several power levels as a priority mechanism. (The latter is not studied in this paper.) One form of redundancy entails dividing a packet into several subpackets, computing from those a larger set of subpackets, and transmitting them; any subset of sufficiently large cardinality, typically equal to the original packet, suffices for reconstruction of the original packet. One can, for example, use MDS error-correcting codes for this purpose [3]. (We are assuming an erasure channel, so error detection is provided by other means and the codes are utilized solely for correction.) In parts of the paper, we will use a simpler form, namely, packet replication, whereby several copies of the packet are transmitted. (Replication does not permit fine control over the degree of redundancy.)

Consider two cases: 1) all decisions must be made up front, without waiting for success/failure notification, and 2) there are several transmission–feedback rounds prior to the deadline. We refer to those as *single-round* and *multiround*, respectively. A "round" is composed of a (possibly multicopy) transmission attempt and the delay until feedback arrives. (Without redundancy, retransmissions take place only post-feedback, and are never redundant since they are known to be necessary.) The terms "attempt" and "round" will be used interchangeably. The single-round case is of primary interest for systems that do not provide feedback, as well as for cases wherein waiting for feedback would cause the deadline to be missed. The multiround case is of primary interest when the permissible delay is several-fold larger than the time until feedback is received. The techniques presented in this paper are particularly suitable for situations that combine high bandwidth with a long propagation delay (relative to packet-transmission time). For the multiround schemes, a deadline that permits multiple rounds is necessary. These situations are likely to be common in new high-bandwidth satellites with on-board processing.

The judicious use of redundancy in slotted ALOHA networks with different goals than those considered here has recently been addressed. In [4], the objective is maximizing capacity. In [5] and [6], multicopy transmissions are considered on a multichannel with no deadlines, and the

Manuscript received February 27, 1998; revised September 14, 1998. This research was supported in part by the Consortium for Satellite Ground Stations, administered by the Chief Scientist of the Israeli Ministry of Industry and Commerce.

The authors are with the Department of Electrical Engineering, Technion—Israel Institute of Technology, Haifa 32000, Israel.

Publisher Item Identifier S 0733-8716(99)01714-X.

throughput–mean-delay characteristics are improved substantially by a proper choice of the (fixed) number of copies transmitted in each attempt. In [7], a single two-copy transmission attempt is considered on a single channel: given a deadline and the permissible probability of missing it, the goal is to minimize the expected delay of successful packets. The optimal probability function of the intercopy delay is derived, and is shown to be a linear monotonically decreasing function.

The remainder of this paper is organized as follows. Section II describes the traffic model and performance measures. Section III presents and analyzes the use of error-correcting codes for the single-round case on a multichannel, exploiting unique characteristics of geostationary satellite systems and their ground stations. In Section IV, we consider the multiround case for a multichannel system with a deadline, and present retransmission policies that dramatically increase capacity for any given probability of meeting the deadline. In Section V, we revisit some of our assumptions and discuss operational issues, and Section VI offers concluding remarks.

II. Traffic Model and Performance Measures

A. Traffic Model

Our model is similar to [8], as follows. There are M multiple-access channels, over which an infinite number of user stations transmit data packets at discrete (slotted) starting times. The transmission of a data packet takes a single time slot, unless a packet is partitioned into subpackets, in which case the transmission of a subpacket takes a single time slot. We assume an erasure channel, and that collisions are the sole source of erasure. Feedback, if available, arrives several time slots after transmission, and the absence of an expected ACK serves as an implicit collision-notification mechanism. Whenever there is a deadline, which arrives after D_s time slots (or D_r rounds), a station ceases to retransmit a packet if it will not meet the deadline. Such a packet is considered lost at the media-access level, although a higher level protocol may eventually resubmit it. Accordingly, we distinguish between the *generated throughput* S_g and the actual *throughput* S, although the difference between them is very small in most practical situations. The number of new data packets per channel in each slot is generated according to a Poisson distribution with mean S_g. This, together with the retransmitted data, is a random variable distributed according to a Poisson distribution with mean G; the Poisson assumption for the offered load G is justified by randomizing retransmissions [2]. (The packet-generation *process* is not assumed to be a Poisson process.) In the case of multichannel systems, even when multicopy transmissions occur, we will continue to use the per-channel measures while taking care not to count duplicate successful receptions of the same packet. This is correct since the transmission of multiple copies takes place on randomly selected channels. Stations are assumed to continue generating new data even while attempting to retransmit a previous packet, in contrast to the commonly used station state model (idle/backlogged). This reflects a situation whereby the generation of messages by applications is unaffected by the details of the state in lower level protocols. (The effect of this subtlety on the results is negligible.)

With temporal randomization on a single channel, the limited permissible delay often severely restricts the number of time slots from which the retransmission time can be chosen, and this brings about a dependence among the fates of different copies of a packet, which reduces performance. In contrast, with multichannel systems featuring more than 100 channels and a restricted number of copies transmitted in each attempt, there is effectively no such dependence among the fates of transmissions in different time slots and among those of multiple copies transmitted in the same time slot. The analysis in this paper is carried out under such an independence assumption, which is confirmed by simulations.

A station is likely to have a limited number of transmitters, which limits the number of concurrent transmissions by a given station. In parts of the paper, this constraint will initially be ignored, but will subsequently be addressed.

The techniques presented in this paper may somewhat reduce network stability. We will assume that a higher level protocol level is used to stabilize the network. The analysis in this paper applies only to the stable periods.

B. Performance Measures

The *success probability* P_s is the probability of decoding a data packet correctly prior to the specified deadline. In the case of a packet that is broken into subpackets, this refers to the decoding of the entire packet from received subpackets. The *error probability* P_e is $1 - P_s$. *Throughput* (S) is the mean number of distinct data packets that are decoded correctly in each time slot, divided by the number of channels. It is related to the generation rate S_g through $S = S_g \cdot P_s$.

For convenience, we define the *delay* incurred by a successfully received packet as the time from its generation (and thus first transmission) until the transmission of the copy (or subpacket) whose successful reception renders the packet "received." The deadline is similarly defined as the time from the generation of a packet until the latest time slot in which a copy or subpacket may be transmitted and still received in time to be considered a success. Note that the performance measures relate to entire data packets, as opposed to subpackets.

III. A Single-Round Transmission Technique

In this section, we explore the case of a multichannel system, in which the deadline is such that all transmissions related to a given packet must take place prior to the receipt of any success/failure feedback. Given the permissible number of time slots (limited by the deadline) and a large number of channels, we may spread our transmissions in time and/or frequency. Our goal in this case is to maximize the attainable throughput subject to a permissible probability of failure in the first and only transmission round.

One proposal for using redundancy in order to enhance performance in similar situations is *redundant dispersity routing* [9]. This entails breaking a packet down into several subpackets, constructing several redundant subpackets, and transmitting all subpackets. If a number of subpackets which equals or exceeds the number of nonredundant subpackets is received prior to the deadline, this is considered a success. A similar idea was presented in [10]. In [11], the idea was combined with that of selective exploitation of redundancy [12] to produce prioritized dispersal, whereby the redundant subpackets receive a lower priority than the "original" ones.

The discussion here will be carried out in the context of very small-aperture terminals (VSAT's) and geostationary satellites. Such systems are characterized by a very small variability in propagation delay and a single point of synchronization (satellite or hub). Accordingly, the temporal guard bands required between time slots may be very small, thereby making it practical to break a packet down into several subpackets without incurring a large overhead due to guard bands. In this case, two important issues must be addressed.

- Helping the transmitter and receiver coordinate a "code," i.e., a sequence of (time, frequency) slots in which the subpackets of a given packet are transmitted. (Note that this "code" is not an error-correcting code; the term is borrowed from code-division multiple access.) Previously analyzed options for code selection are transmitter-based codes, receiver-based codes, and hybrid methods [13] [14]. With finite networks, one can indeed assume that a specific code and hopping pattern are assigned to each transmitter [15]. With a very large population, however, as in our case, this is problematic. Our solution is to include the code in the header of a message.
- Overcoming the large header overhead that results from partitioning a data packet into many small subpackets. Despite the small guard bands, overhead increases as one reduces packet size due to the required header. (This header contains the usual source and destination information; in our case, it must also contain code-related information.)

Our solution to both problems is as follows. In order to establish synchronization, a transmitter selects a seed for a previously agreed upon random number generator. The seed, together with synchronization, code, address, and other control information, is transmitted as a "preamble" subpacket several times, so the probability of not receiving any copies of this initial subpacket is very low. After this phase, the transmitter proceeds to transmit the data subpackets on channels selected according to the random number generator. Since the receiver knows exactly when and on which channel the next transmission will occur, no overhead is needed in each subpacket, except for the small guard bands and a bit-synchronization pattern. Before transmission, the subpackets are coded redundantly so as to ensure correct decoding for the expected number of collisions. In the analysis, we consider subpacket collisions to be independent of one another since channels are selected at random. With this technique, capacity is maximized for a given probability of success. This method can be considered as a hybrid technique combining ALOHA and frequency-hopping spread spectrum.

The permissible delay and the number of channels define a boundary within which the subpackets may be placed. The only additional constraint is that the preamble subpackets must be transmitted prior to the data subpackets. In order to minimize delay (a possible secondary optimization goal), one would transmit the preamble subpackets in one slot, followed immediately by all copies of the data subpackets. However, hardware constraints such as a limited number of transmitters may mandate the spreading of transmissions over several time slots. Since we only care about the attainable capacity subject to meeting a given deadline with the required probability, spreading transmission within the permissible time interval is "free." The combination of the deadline (in slots) and the number of transmitters may, however, restrict the choice of the number of replicas of the header and the error-correcting code for the payload. The analysis below is unconstrained, but the incorporation of a constraint is straightforward.

Analysis: Let each original packet comprise d bits of data and an h-bit header. Each data subpacket comprises d/k bits of data and no header. The "preamble" subpackets contain only an h-bit header. For facility of analysis, we set k such that $d/k = h$. Also, we ignore the fact that the preamble packet must be longer than the header of the original packet due, for example, to the need to also include the seed for the code generator. This effect is secondary, and in any case, our results in this section should be taken as an indication rather than as precise numbers.

Subpacket size will be derived from the number of overhead bits h (including the seed), and the d bits of data will be split among k subpackets, each consisting of h bits. The k data subpackets will be redundantly coded (this is the error-correcting code) to $n > k$ subpackets and transmitted on "randomly selected" (based on the sequence generated by the random number generator) channels. Transmission of the preamble subpacket will be repeated R times.

The receiver can decode the original packet from any k subset of the n transmitted subpackets (erasure channel), provided that it has successfully received at least one copy of the preamble subpacket. The probability of receiving at least one of the copies of the preamble is

$$1 - \left(1 - e^{-G}\right)^{R}. \tag{1}$$

The probability of losing fewer than $n - k$ data subpackets is

$$\sum_{i=0}^{n-k} \binom{n}{i} \cdot \left(1 - e^{-G}\right)^{i} \cdot e^{-G(n-i)}. \tag{2}$$

Since those are independent events, the probability of success P_s is

$$P_s = \left(1 - \left(1 - e^{-G}\right)^{R}\right) \cdot \sum_{i=0}^{n-k} \binom{n}{i} \cdot \left(1 - e^{-G}\right)^{i} \cdot e^{-G(n-i)}. \tag{3}$$

The overhead for R subpackets is $R \cdot h$ bits, compared with h bits of ordinary ALOHA. The useful data enclosed are $d = k \cdot h$ bits. The throughput is therefore

$$S = G \cdot P_s \cdot \frac{h + k \cdot h}{R \cdot h + n \cdot h} = G \cdot P_s \cdot \frac{k+1}{n+R}. \tag{4}$$

(If the difference between the size of the preamble packet and the size of the original packet header were taken into

TABLE I
ATTAINABLE THROUGHPUT (CAPACITY) S FOR $P_s = 0.9, 0.99, 0.999$

P_s	R_{opt}	S	$S(ALOHA)$
0.9	6	0.2219	0.0948
0.99	8	0.1820	0.0099
0.999	5	0.1458	0.0001

account, "$k+1$" in the above equation would be replaced with "$k+h'/h$," where h' is the original header size and h is the size of the preamble subpacket.)

Note that $k, n, G, R,$ and P_s are related through (3), so G is determined once the others are assigned values.

Numerical Results: In obtaining the results, we used $d = 1000, h = 100$, $k = 1000/100 = 10$, and a (32, 10) error-correcting code ($n = 32$). Several values of P_s were used, and R was selected to maximize the throughput in each case.

Table I compares our scheme with single-channel ALOHA (our throughput is per channel). S(ALOHA) is the throughput achievable by classic ALOHA for a probability of success P_s, namely

$$S = G \cdot e^{-G} = -\ln(P_s) \cdot P_s. \tag{5}$$

The results show that our method provides a dramatic increase in the throughput that can be attained while still providing a very high probability of success in the first attempt. Moreover, it permits the use of simple narrow-bandwidth transmitters together with inexpensive processing power. References [14] and [16] provide further analysis of ECC usage.

The nonmonotonic behavior of the optimal value of R can be explained by the tradeoff between the negative effects of increasing it on the probability of success of data subpackets through increasing the load on one hand, and increasing the probability of successful synchronization on the other hand. Finally, we note that even better results can be obtained by jointly optimizing $k, n,$ and $R,$ so the above serves as a lower bound on the achievable improvement.

IV. MULTIROUND RETRANSMISSION POLICIES

In this section, we consider the situation wherein the deadline permits up to D_r transmission attempts (rounds), with a new attempt being made only after success/failure feedback has been received for the previous one. Immediately following the receipt of feedback which indicates failure of a transmission attempt, a station transmits one or more copies of the lost packet over randomly chosen channels. (A negative acknowledgment may be implicit, i.e., deduced from the lack of a positive one.) Since there is no benefit from delays in a multichannel system wherein collisions are independent of one another, at least one packet is transmitted in each attempt. Following success, retransmission ceases; after D_r attempts, a packet is declared lost and is discarded. (In practice, a higher level protocol may resubmit the packet, but it would be considered a new one. Moreover, assuming a low permissible probability of failure, the high-level retransmission traffic is negligible.)

Given D_r and the permissible probability of failure, our goal is again to maximize the attainable throughput. This is done through a judicious choice of the number of copies that should be transmitted in each attempt (not necessarily the same number in all attempts).

Conventional back-off policies aimed at preventing instability, when applied to a multichannel, would call for a monotonically nonincreasing number of copies in successive retransmission attempts. However, while the stability argument underlying such an approach is valid asymptotically, we claim that this monotonicity may be violated for any bounded number of retransmissions without hurting stability, and we will show that so doing can dramatically increase performance.

Given the maximum number of copies that may be transmitted (jointly in all attempts), we strive to minimize the expected aggregate number of copies transmitted within the D_r permissible attempts for any given probability of success P_s. This, in turn, minimizes the load generated per successful message, thereby maximizing the system's communication capacity. Our approach typically entails the transmission of a single or very few copies in all but the final attempt, in which the remaining copies are transmitted if necessary.

There is a tradeoff in selecting how to use an allotted "budget" of copies: on one hand, we would like to postpone the transmission of all but one copy per attempt in the hope that an early attempt will be successful and later ones thus avoided; on the other hand, it might be beneficial to transmit more than one copy per attempt prior to the last attempt since this increases the probability of avoiding the last, "costly" attempt.

Analysis: We begin by presenting the relations among the various variables. Next, we analyze two schemes: 1) multicopy ALOHA [5] (a constant number of copies in each attempt), and 2) a new scheme, whereby a single copy is transmitted in all but the last attempt. Finally, we employ dynamic programming [17] to derive the optimal retransmission strategy.

The number of copies transmitted by a single station, even in its last attempt, is assumed to be much smaller than the number of channels. As explained in the channel model, this, combined with the large population, makes the probability of success of any given copy effectively independent of the number of copies of the same packet that are transmitted in the same time slot.

Let N denote the total number of copies of each packet (until success or deadline); $N_{\max}$ and $\overline{N}$ will be used to denote its maximum and expected value, respectively. Then, the throughput is

$$S = \frac{G \cdot (1 - P_e)}{\overline{N}}. \tag{6}$$

Therefore, if G and P_e are held constant, minimizing $\overline{N}$ will maximize S. The channel capacity will be the maximized throughput value.

Since failures are independent and $N_{\max}$ copies are transmitted unless success is detected prior to the last round, the probability of all transmissions of a packet failing P_e is

$$P_e = \left(1 - e^{-G}\right)^{N_{\max}}. \tag{7}$$

Alternatively, we can express G as

$$G = -\ln\left(1 - P_e^{1/N_{\max}}\right). \tag{8}$$

Let n_i denote the number of copies transmitted in the ith attempt. Then

$$N = \sum_i n_i. \tag{9}$$

Also

$$N_{\max} = \sum_{i=1}^{D_r} n_i. \tag{10}$$

The ith transmission attempt will occur if and only if all preceding attempts fail. The probability of this event is

$$\Pr(i\text{th attempt}) = \left(1 - e^{-G}\right)^{\Sigma_{j=1}^{i-1} n_j}, \qquad i = 2, \cdots, D_r \tag{11}$$

and

$$\Pr(1\text{st attempt}) = 1. \tag{12}$$

Transmission proceeds until one of the attempts is successful or until the deadline. Therefore, the expected total number of copies transmitted per packet is

$$\overline{N} = \sum_{i=1}^{D_r} n_i \cdot \left(1 - e^{-G}\right)^{\Sigma_{j=1}^{i-1} n_i}. \tag{13}$$

Multicopy ALOHA (c Copies per Attempt): Here, $N_{\max} = c \cdot D_r$ and $n_i = c$ for $i = 1, \cdots, D_r$. Substituting in (13) and (8) yields

$$\begin{aligned}\overline{N} &= c \cdot \left(1 + \left(1 - e^{-G}\right)^c + \cdots + \left(1 - e^{-G}\right)^{c(D_r - 1)}\right)\\ &= c \cdot \frac{1 - \left(1 - e^{-G}\right)^{cD_r}}{1 - (1 - e^{-G})^c}\end{aligned} \tag{14}$$

where

$$G = -\ln\left(1 - P_e^{1/(c \cdot D_r)}\right). \tag{15}$$

Minimizing $\overline{N}$ over c produces the final results.

A Single Copy in All But the Last Attempt: (1, 1, $\cdots$, 1, c): Here, $N_{\max} = D_r + c - 1$. Substituting in (13) and (8) yields

$$\begin{aligned}\overline{N} &= 1 + \left(1 - e^{-G}\right) + \cdots + \left(1 - e^{-G}\right)^{D_r - 2}\\ &\quad + c\left(1 - e^{-G}\right)^{D_r - 1}\\ &= e^G \cdot \left(1 - \left(1 - e^{-G}\right)^{D_r}\right) + (c-1)\left(1 - e^{-G}\right)^{D_r - 1}\end{aligned} \tag{16}$$

where

$$G = -\ln\left(1 - P_e^{1/D_r + c - 1}\right). \tag{17}$$

Minimizing $\overline{N}$ over c again produces the final results.

The Optimal Replication-Based Policy: Given D_r and P_e, we now proceed to minimize $\overline{N}$ using a dynamic programming approach. Initially, we assume that $N_{\max}$ is given and the entire budget must be spent, so our goal is to optimize the allocation of $N_{\max}$ copies among the D_r rounds.

We use (8) to express G in terms of P_e and $N_{\max}$; next, this is substituted for G in (13) to yield $\overline{N}$ as a function of the requirements and the budget distribution (n_i). For clarity of presentation, we will nonetheless continue to show G in the expressions.

In dynamic programming terms, the number of attempts used so far will be the system *stage*. The vector of number of copies per attempt n_i is the *state* of the system. The *decision* made in each attempt is how many copies should be transmitted out of the remaining budget. After the decision, the state variables undergo a *transformation* whereby the chosen number of copies is appended to the vector. Our *return function* is the expected number of transmissions so far which will be denoted as $\overline{N}_t(n)$. t and n will denote the iteration variables, $1 \le t \le D_r$ and $1 \le n \le N_{\max}$.

To use the optimality principle, which states that the dynamic programming technique will find a global optimum [17], two conditions must be met: 1) the objective function must be separable in the sense that the effect of the final stage on the objective function depends only on the previous state and the last decision; and 2) state separation property: after a decision is made, the next state depends upon the previous state and the decision.

Condition 1) is met since the expected contribution (to N) of a possible transmission of x copies after $\Sigma\, n_i$ copies have been transmitted is $x \cdot (1 - e^{-G})^{\Sigma\, n_i}$, which depends only on the state variables n_i and the decision variable x. From the definition of the transformation, our next state vector depends only on the previous state vector and on the decision made, so the second condition holds true.

The recurrence equation is

$$\overline{N}_t(n) = \min_{i=1 \cdots n-t+1} \left(i \cdot \left(1 - e^{-G}\right)^{n-i} + \overline{N}_{t-1}(n-i)\right) \tag{18}$$

with the boundary condition

$$\overline{N}_1(n) = n. \tag{19}$$

The optimal policy for transmitting n packets in t attempts is composed by abutting an optimal subpolicy for transmitting fewer than n packets in $t-1$ attempts, with the transmission of the remaining copies in the last attempt. (The determination of how many of the n packets should be transmitted in the last attempt is part of the optimization.) Thus, we have constructed a recursive formula for $\overline{N}_{D_r}(N)$, the minimum expected number of copies of a packet transmitted in up to D_r attempts, given a maximum permissible number $N_{\max}$ (and recalling that transmissions cease upon successful reception).

Having solved the optimization problem for any given total number of copies $N_{\max}$, the remaining step is to optimize over $N_{\max}$. Although intuition suggests that there should be a single local minimum of $\overline{N}$, this is not always the case, apparently due to quantization problems. Nonetheless, since $N_{\max}$ is small in all practical situations and the optimization

TABLE II
ATTAINABLE-THROUGHPUT COMPARISON AMONG VARIOUS RETRANSMISSION SEQUENCES ON A MULTICHANNEL WITH UP TO D_r TRANSMISSION ATTEMPTS (ROUNDS) AND AN ERROR PROBABILITY P_e

Pe	D_r	$S(1,1,\cdots,1)$	$S(c,c,\cdots,c)$	c	$S(1,1,\cdots,c)$	c	S(opt)	Sequence(opt)
0.01	3	0.1904	0.2447	2	0.2683	3	0.2787	[1 2 4]
0.01	5	0.3056	0.3056	1	0.3374	3	0.3404	[1 1 1 2 3]
0.001	3	0.0948	0.1872	3	0.2254	5	0.2470	[2 3 7]
0.001	5	0.2166	0.2604	2	0.3098	5	0.3213	[1 1 1 2 5]
0.0001	3	0.0453	0.1488	4	0.1984	6	0.2330	[2 3 10]
0.0001	5	0.1452	0.2186	3	0.2913	7	0.3125	[1 1 2 3 8]

over $N_{\max}$ is carried out once, an exhaustive search over $N_{\max}$ is reasonable.

Numerical Results: The probability of failure was held equal for all schemes, and in each of the two parameterized schemes c was chosen to maximize capacity. Numerical results have been obtained for several values of P_e, for $D_r = 3$ and for $D_r = 5$. Sample results are provided in Table II, and more are plotted in Fig. 1.

For $P_e = 0.001$, the optimal sequence (1, 1, 1, 2, 5) achieves a capacity of 0.3213. This is 50% higher than the capacity with a (1, 1, $\cdots$, 1) sequence, 23% higher than with a (2, 2, $\cdots$, 2) sequence, and 3% higher than with the $(1, 1, \cdots, 5)$ sequence. The advantage of the optimal solution is even more pronounced for lower values of P_e or D_r.

Remarks:

1) The maximum capacities of the different schemes (even for the same probability of failure and deadline) occur with different maximum numbers of copies, and therefore at different values of G (different probability of collision for any given copy).
2) We see that our optimal method achieves the highest capacity for any given P_e and D_r. The advantage becomes more pronounced as the permitted error probability is reduced.

A Limited Number of Transmitters: In practice, the number of transmitters (and hence concurrent transmissions by a single station) is severely limited. We denote this limit by K. The foregoing dynamic programming analysis is next modified to accommodate this limitation

$$\overline{N}_t(n) = \min_i \left(i \cdot \left(1 - e^{-G}\right)^{n-i} + \overline{N}_{t-1}(n-i) \right) \quad (20)$$

with i constrained to the range

$$i = \max(1, n - K \cdot (t-1)), \cdots, \min(K, n-t+1) \quad (21)$$

and the boundary condition

$$\overline{N}_1(n) = n. \quad (22)$$

A comparison with the unconstrained case is presented in Table III for two and three transmitters $(K = 2, 3)$ and for three and five rounds $(D_r = 3, 5)$ for several values of P_e. The disadvantage of a restricted number of transmitters (and thus transmissions per round) is more pronounced for smaller permissible probabilities of not meeting the deadline. For $P_e > 0.01$, the difference becomes negligible.

Round Stretching: So far, a station was permitted to transmit only in the first slot of a round, and subsequently had to wait for feedback. With this constraint, the optimal way of accommodating a constrained number of transmitters per station was to solve the constrained version of the dynamic programming problem.

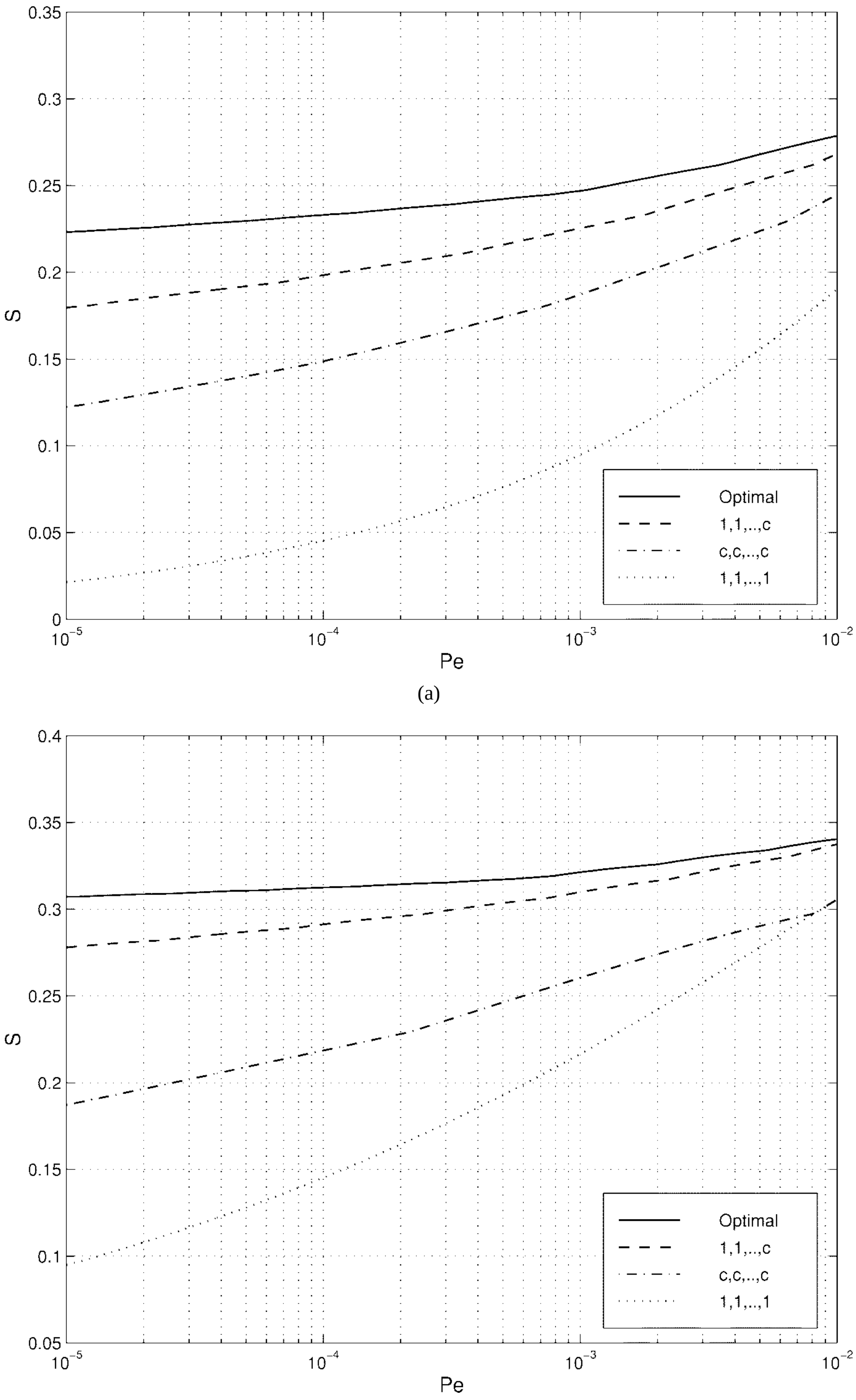
In this section, we present an alternative approach, whereby the transmissions of a given round occur in a contiguous sequence of slots, beginning with the first slot of the round. Following the transmissions, a station ceases transmission awaiting feedback. Compared with the original scheme, this approach entails "stretching" the round by one less than the number of slots during which transmission is permitted. Taken to the extreme, a single transmitter can be used by each station. In the remainder of this section, we focus on the single-transmitter case in view of its practical importance; nonetheless, hybrids are possible, and the analysis can be adapted.

Unlike slot synchronization which must be global, a round is "private" to each transmitting station. Accordingly, the duration of a round is determined by the number of copies transmitted in it (and thus the number of slots used for transmission). The optimal transmission scheme entails transmitting very few copies, usually one, during all but the last round; therefore, the total number of time slots required for emulating the multitransmitter policy using a single transmitter is not much greater than that required with multiple transmitters. In the remainder of this section, we derive the optimal single-transmitter policy, and compare it with the unconstrained optimal solution.

Derivation of Capacity with Round Stretching and a Single Transmitter: A deadline (along with the propagation delay in slots) defines the maximum number of transmission rounds. Any permissible number of rounds moreover determines the maximum aggregate number of copies per message $N_{\max}$. With a single transmitter per station, there is complete flexibility in the allocation of this "budget" to the rounds as long as at least one copy is transmitted per round.

For each number of permissible rounds, we begin by employing dynamic programming to find the optimal allocation of the total transmission budget among the rounds. The relationship between the total budget and the deadline enables us to plot the attainable throughput versus the deadline, as depicted by the dashed lines in Fig. 2.

Extending the deadline while holding the number of rounds fixed increases the budget. However, spending the entire



(a)

(b)

Fig. 1. Attainable throughput (capacity) versus P_e for multiround transmissions on a multichannel. (a) The deadline permits three rounds and (b) five rounds.

budget may be suboptimal: it may entail transmitting more copies than are required for achieving the specified probability of failure to meet the deadline, thereby creating excessive offered load and reducing the attainable useful throughput. Therefore, forcing the use of the entire budget causes the attainable throughput with any given number of rounds to peak at a certain value of the deadline and to then fall off. The optimal policy for any chosen number of rounds (as a function of deadline) is to use the entire budget up to the peak of the curve, and to not transmit any more even if the deadline is extended. As depicted by the dot–dash curves (partly masked by the solid curves) in Fig. 2, the attainable throughput then stays at its peak as the deadline is extended. (It should be noted that the avoided transmissions would have contained additional copies in the last permissible round, i.e., would not have been based on knowledge that all prior transmissions had failed. Avoiding them is thus not inconsistent with the intuition whereby one should not give up prior to the deadline.)

TABLE III
ATTAINABLE-THROUGHPUT (CAPACITY) COMPARISON AMONG THE OPTIMAL RETRANSMISSION SEQUENCES ON A MULTICHANNEL WITH ERROR PROBABILITY P_e AND UP TO K CONCURRENT TRANSMISSIONS (K TRANSMITTERS PER STATION): (a) UP TO THREE TRANSMISSION ATTEMPTS (ROUNDS); (b) UP TO 5 ROUNDS

D_r	Pe	S_{opt}	Seq.	$S_{opt}(K=3)$	Seq.	$S_{opt}(K=\infty)$	Seq.
3	0.01	0.2614	[1 2 2]	0.2772	[1 2 3]	0.2787	[1 2 4]
3	0.001	0.1884	[1 2 2]	0.2198	[1 2 3]	0.2470	[2 3 7]
3	0.0001	0.1302	[1 2 2]	0.1716	[1 3 3]	0.2330	[2 3 10]
3	0.00001	0.0877	[1 2 2]	0.1355	[1 3 3]	0.2232	[2 4 13]
5	0.01	0.3378	[1 1 1 2 2]	0.3404	[1 1 1 2 3]	0.3404	[1 1 1 2 3]
5	0.001	0.2954	[1 1 2 2 2]	0.3135	[1 1 2 3 3]	0.3213	[1 1 1 2 5]
5	0.0001	0.2471	[1 1 2 2 2]	0.2828	[1 1 2 3 3]	0.3125	[1 1 2 3 8]
5	0.00001	0.2034	[1 2 2 2 2]	0.2489	[1 1 3 3 3]	0.3071	[1 1 2 3 11]

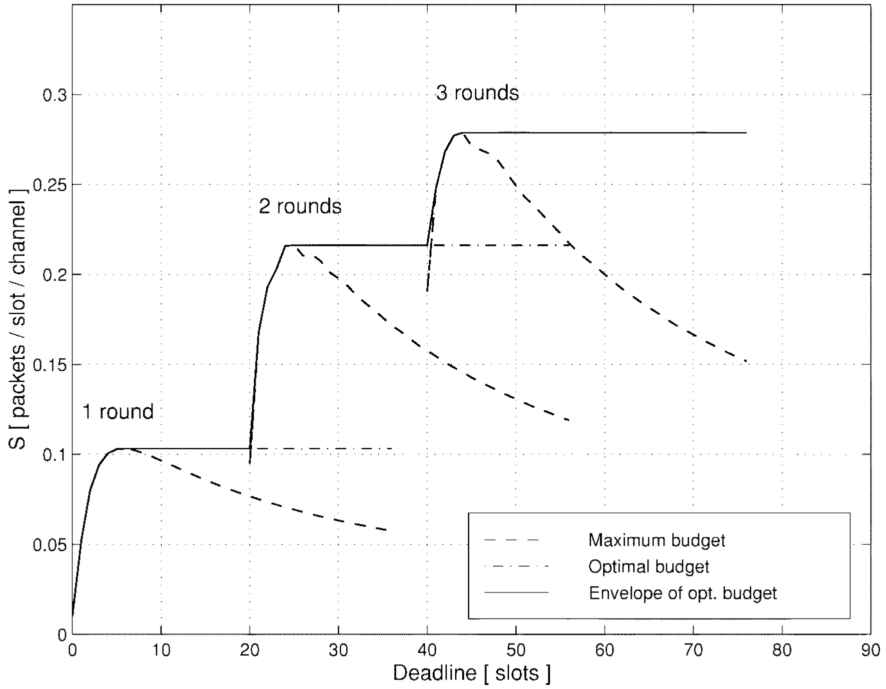


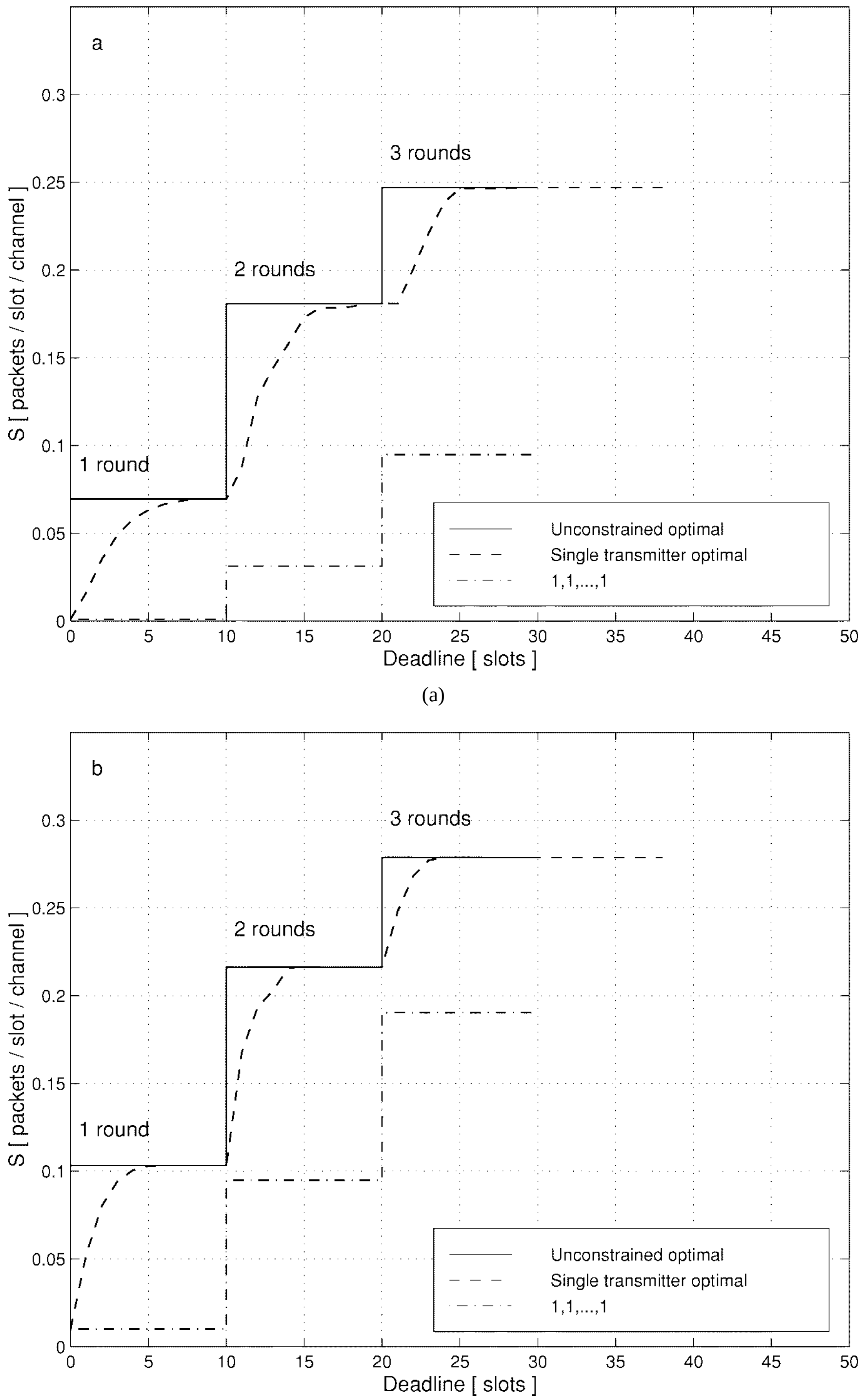
Fig. 2. Attainable throughput (capacity) versus deadline with round stretching using a single transmitter. The graph envelope is constructed as the maximum (at any given deadline) among the “nondecreasing” results for the different numbers of rounds, and represents the optimum among all replication-based single-transmitter policies. The plots are for $P_e = 0.01$ and a round-trip delay of 20 slots.

Finally, for any value of deadline, we pick the optimal number of rounds by selecting the largest value from among the individual curves, as depicted by the solid curve in Fig. 2. This choice represents the optimal replication-based single-transmitter policy for the given set of parameters and requirements, and the solid curve depicts capacity as a function of D_r. Throughout the remainder of this section, we use these envelope curves in comparing the optimal single-transmitter scheme with optimized versions of other schemes.

Fig. 3 presents a comparison among the optimal unconstrained and single-transmitter replication-based policies, as well as the baseline (single-transmitter, single-transmission per round). Plots are shown for $P_e = 0.001$ and 0.01 and round-trip delays of 10 and 20 rounds; these are realistic engineering values. It is readily noticed that the difference in attainable throughput between the unconstrained scheme and the single-transmitter one depends on the deadline in a periodic manner, corresponding to the remainder of the division of the deadline by the round-trip propagation delay. As the number of permissible rounds increases, the approximation becomes closer and closer.

In practice, the round-trip propagation delay is on the order of tenths of a second or even less, and a slot is perhaps ten times (or more) shorter than that. Since deadlines are often dictated by human-response measures, it is clear that extending a deadline by a few slots is insignificant. Therefore, another interesting way of interpreting the comparison is by measuring the horizontal distance between curves, i.e., the number of slots by which the deadline would have to be extended in order for the single-transmitter scheme to attain the throughput that is possible (with the original deadline) with the unconstrained scheme. This comparison reveals that the required extension is quite small: approximately five time slots for $P_e = 0.001$ and three slots for $P_e = 0.01$. As the round-trip delay increases (in terms of slots), the relative cost of stretching decreases.

Having demonstrated that the attainable performance with the single-transmitter scheme exhibits most of the benefits of the unconstrained scheme, we conclude this section with a



(a)

(b)

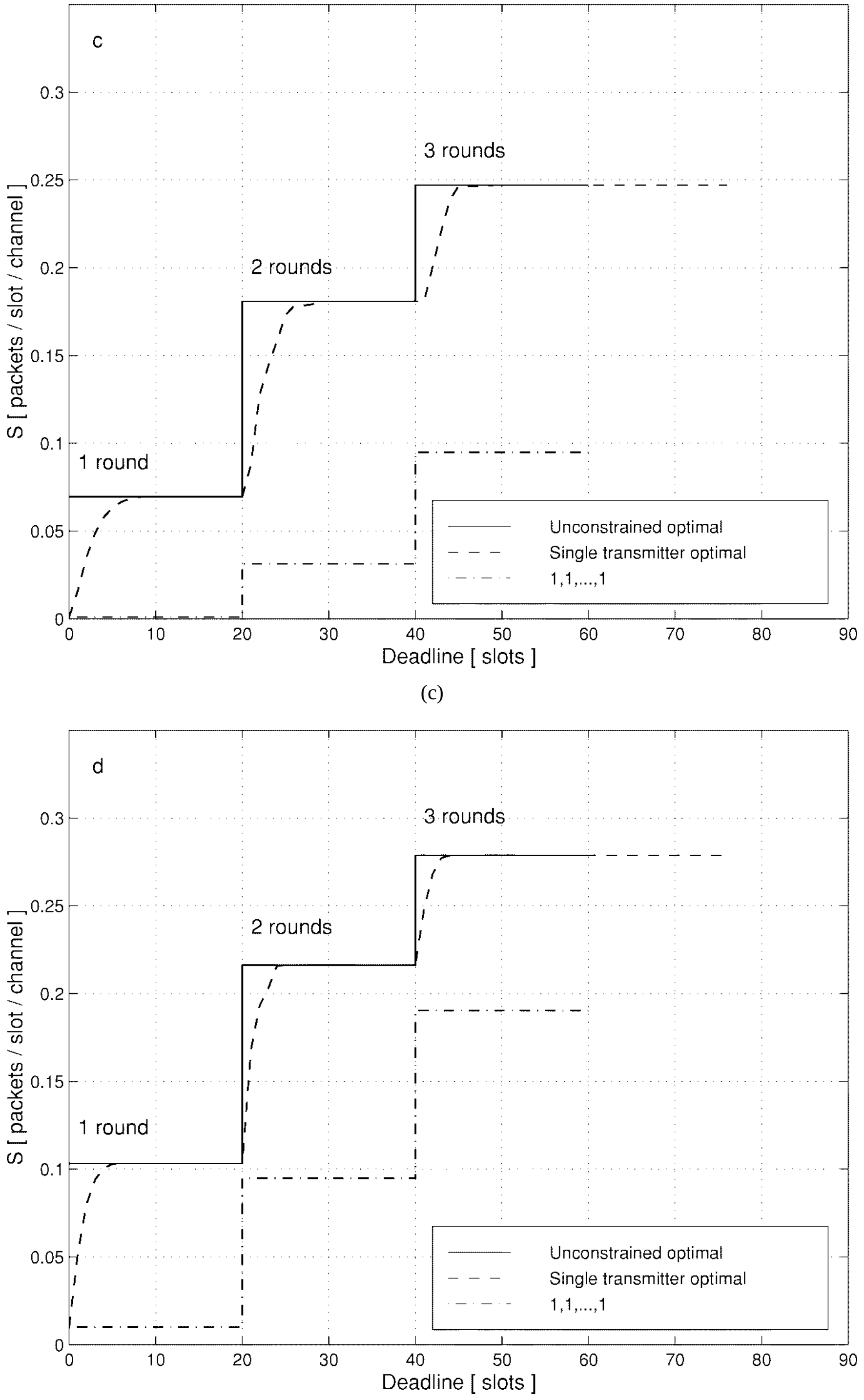
Fig. 3. Capacity versus deadline with an optimal unconstrained replication-based scheme, an optimal such single-transmitter scheme (round stretching), and the baseline scheme of a single transmitter with no redundancy. (a) $P_e = 0.001$, round trip = 10; (b) $P_e = 0.01$, round trip = 10.

comparison among single-transmitter versions of our scheme, multicopy ALOHA, and the baseline (nonreplicated) scheme. Fig. 4 depicts the results for $P_e = 0.001$ and 0.01, and round-trip delays of 10 and 20 slots. Clearly, our optimal scheme achieves a much higher capacity than (optimized) multicopy ALOHA, not to mention single-copy ALOHA. With three rounds and $P_e = 0.001$, for example, capacity increases from 0.19 to 0.25, an improvement of 30%. When $P_e = 0.01$, it increases by 15%.

V. Discussion

In the last two sections, we have presented promising schemes that can very substantially increase the capacity of multichannel ALOHA networks in deadline-constrained operation. In this section, we review the main simplifying assumptions that were made, and discuss operational issues.

Stability and Control Policy: In practice, there may be situations in which the attempted throughput exceeds capacity and the ALOHA protocol becomes unstable [18], [19]. For



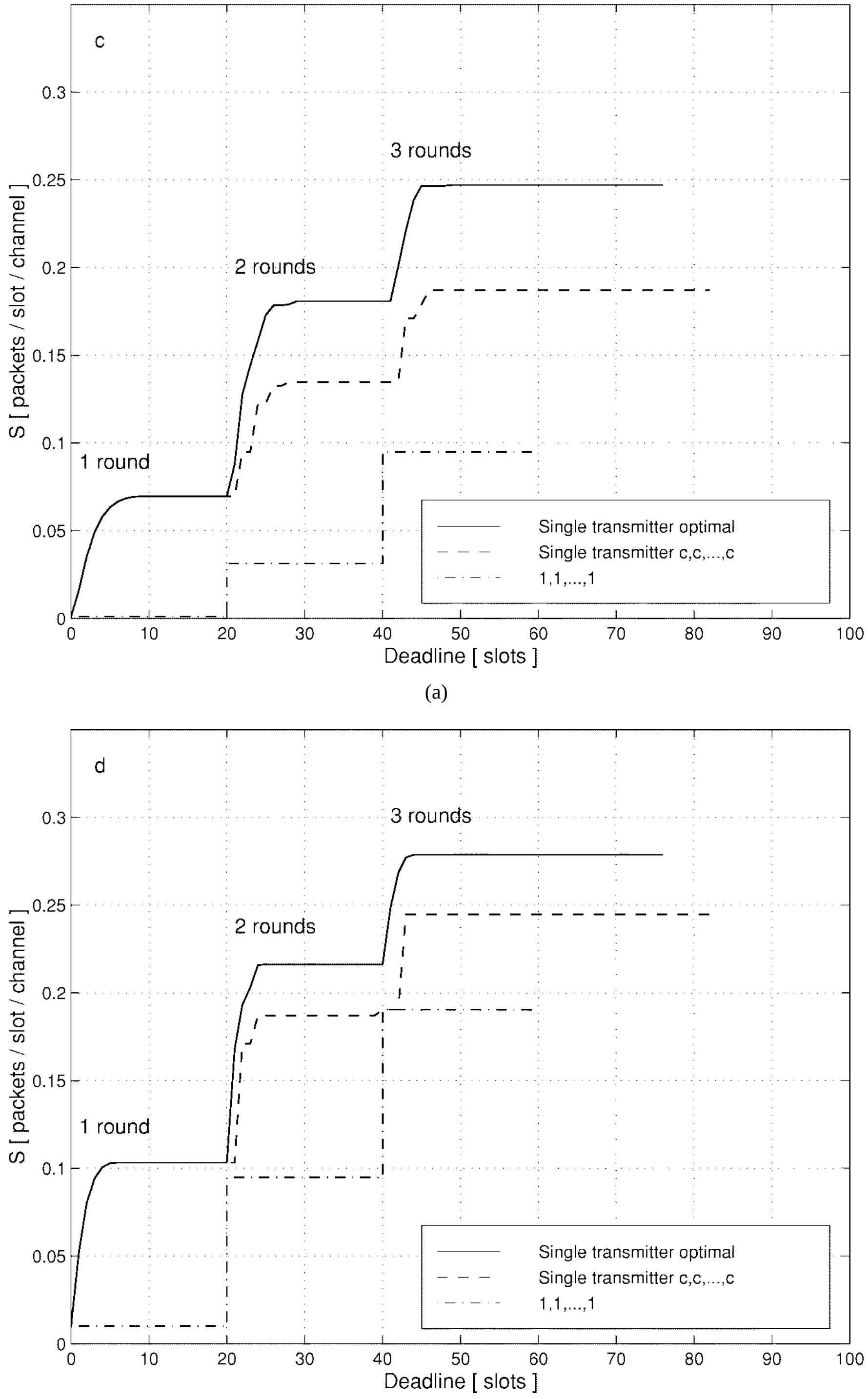
(c)

(d)

Fig. 3. (*Continued.*) Capacity versus deadline with an optimal unconstrained replication-based scheme, an optimal such single-transmitter scheme (round-stretching), and the baseline scheme of a single transmitter with no redundancy. (c) $P_e = 0.001$, round trip = 20; (d) $P_e = 0.01$, round trip = 20.

those cases, which would typically be infrequent, a background process could be used to estimate the offered load and cause stations to throttle down their traffic in order to bring the system back into a stable region. Such activities are not very demanding, and are not part of the core random-access scheme. Moreover, their incorporation is not expected to substantially alter the results of this paper.

Independent Collisions: We have assumed a sufficiently large number of channels so that collisions may be considered independent of one another. This assumption is not accurate for the case wherein several copies are transmitted over tens of channels, as in practical systems today. The dependency among multiple collisions in such cases will lower the performance gains seen in this work, and an exact analysis is warranted. Nonetheless, the approximation is sufficiently close and the improvement very substantial, so the merit of the proposed schemes is established.

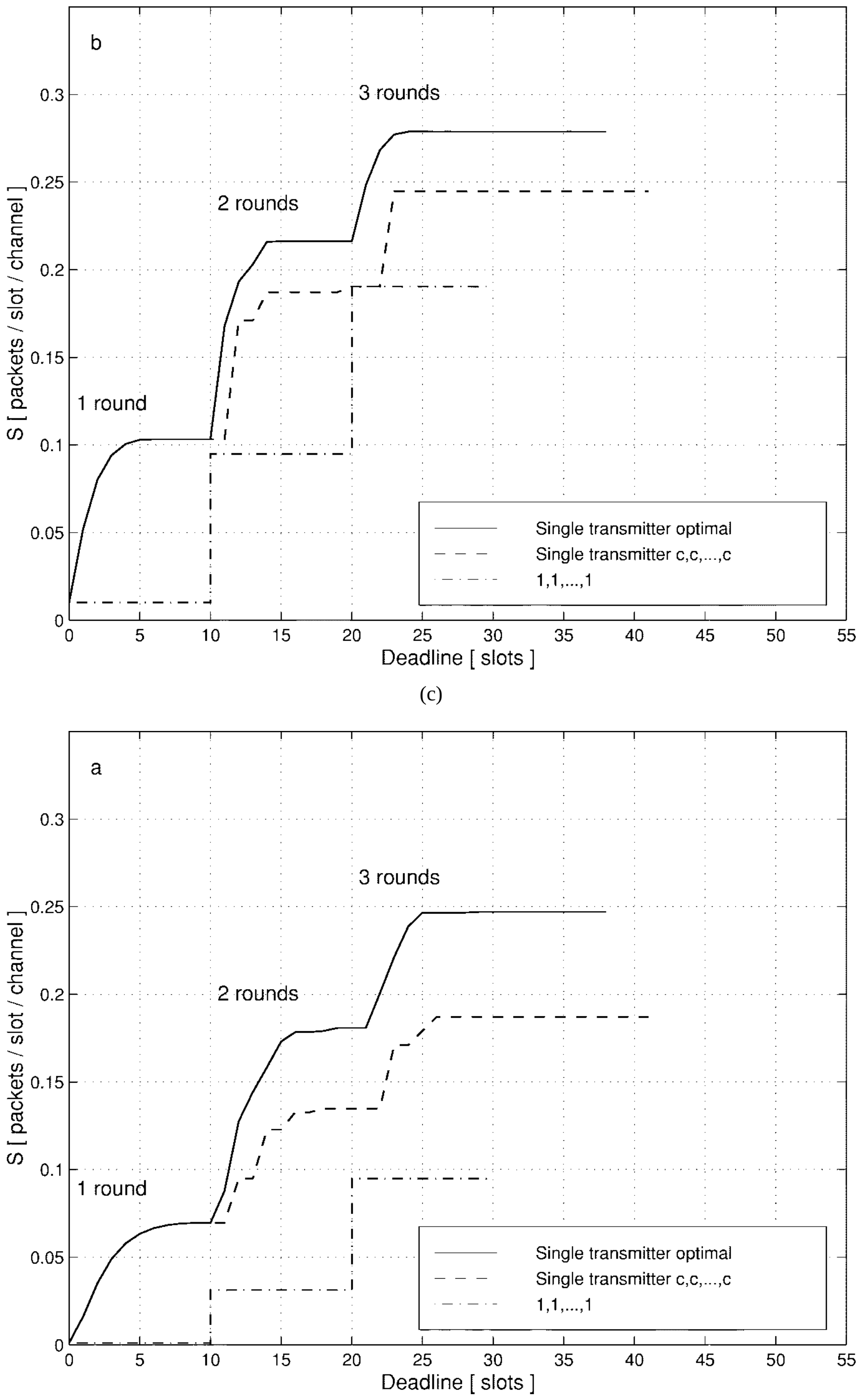


(a)

(b)

Fig. 4. Capacity versus deadline with the optimal single-transmitter replication-based scheme, optimized multicopy ALOHA (single transmitter), and the baseline (nonreplicated) multichannel ALOHA. (a) $P_e = 0.001$, round trip = 10; (b) $P_e = 0.01$, round trip = 10.

Estimation of Offered Load: A problem often associated with optimized operation of random-access schemes is that of continuously estimating the offered load. For example, retransmission policies need this information in order to minimize the expected delay for any given throughput. With the user-oriented performance measures employed in this paper, however, this is not the case: all that matters is maximization of attainable throughput subject to exceeding a specified deadline with a probability that does not exceed a specified value; there is no reward for reducing delay. This, combined with the fact that the probability of collision of any given copy increases with an increase in offered load, implies that it suffices to use a fixed transmission policy, which is tuned for the operating point at which the maximum capacity (subject to the constraints) is attained. Whenever throughput is below that (for lack of generated packets), the constraints will definitely be satisfied; the fact that we could do even better (e.g., in the sense of mean delay) in these situations



(c)

(d)

Fig. 4. (*Continued.*) Capacity versus deadline with the optimal single-transmitter replication-based scheme, optimized multicopy ALOHA (single transmitter) and the baseline (nonreplicated) multichannel ALOHA. (c) $P_e = 0.001$, round trip = 20; (d) $P_e = 0.01$, round trip = 20.

is irrelevant! This observation greatly simplifies the use of our scheme.

VI. Conclusions

We have introduced a performance measure along with a corresponding optimization goal: maximization of capacity subject to keeping delay below a specified deadline with at least a specified probability. This realistic measure captures both the user view and system-architect view of a satellite-based communication system used for transaction processing. Moreover, we have shown how to judiciously exploit redundancy in order to substantially increase the capacity of a multichannel ALOHA network for any given deadline and the permissible probability of not meeting it.

For a single round of transmissions on a multichannel, we focused on a scenario that is typical of geostationary satellites and their VSAT ground stations. We showed that the combination of replicated preambles and lower overhead error correction for the data portion results in a dramatic increase of the attainable throughput subject to a required probability of success. This is an adaptation of redundant dispersity routing to this situation. As an example, a throughput of 0.18 (per channel) with 99% probability of success was shown, as compared with a throughput of 0.01 for conventional ALOHA. Further improvement of this result is possible by optimizing the code selection.

Optimal replication-based multiround retransmission policies were devised for the multichannel. Most important, it was shown that the best policy in the case of a deadline is to transmit one or very few copies of the packet at a time until the last transmission attempt. Then, a burst of packets is transmitted. This method sharply increases the attainable throughput for any given deadline and the permissible probability of failing to meet it, and the relative increase is greater when the permissible probability of failure is smaller. We addressed the constraint of a limited number of transmitters per station, focusing on the most practical case of a single transmitter. We showed how "round stretching" can be employed to substitute time for transmitters, achieving with a single transmitter results that come close to those of multiple transmitters.

Both the single- and multiround schemes are most applicable to transaction-oriented applications using high-bandwidth satellites, in which the permissible delay is much larger than the transmission time of a packet. Both schemes require the use of receivers on all channels, thus requiring a hub. Satellites that can receive all transmissions and process them on board would obviate the need for a terrestrial hub, thereby cutting in half the round-trip propagation delay, substantially reducing the required spectral bandwidth, and doubling the permissible number of rounds; as seen in the numerical results, doing so could substantially increase the attainable throughput while adhering to the same deadline. Our schemes can thus greatly benefit from such satellites.

One interesting direction for continued research on this topic entails the application of more general error-correction techniques to the multiround case. A first step would entail the application of those to individual transmission attempts (rounds); however, they could also be applied across rounds, allowing the receiver to accumulate subpackets of the same original data packet.

Finally, it is important to stress that the schemes presented in this paper are optimized for user-oriented performance measures. Moreover, the replication-based single-transmitter multiround scheme can be implemented very easily. In view of this, the fact that the performance improvements are very substantial, and since the simplifying assumptions appear to have a minor impact, the schemes suggested in this paper may be of practical value in addition to their academic merit.

REFERENCES

[1] L. G. Roberts, "ALOHA packets, with and without slots and capture," *Comput. Commun. Rev.*, vol. 5, pp. 28–42, 1975.

[2] L. Kleinrock and S. S. Lam, "Packet switching in a multi-access broadcast channel: Performance evaluation," *IEEE Trans. Commun.*, vol. COM-23, pp. 410–423, Apr. 1975.

[3] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error Correcting Codes.* Amsterdam: North-Holland, 1978.

[4] Y. W. Leung, "Generalized multi-copy ALOHA," *Electron. Lett.*, vol. 31, pp. 82–83, Jan. 19, 1995.

[5] E. W. M. Wong and T.-S. P. Yum, "The optimal multi-copy Aloha," *IEEE Trans. Automat. Contr.*, vol. 39, pp. 1233–1236, June 1994.

[6] Y. Birk and Y. Keren, "Redundant transmissions and retransmission scheduling for improved throughput–delay in ALOHA networks," CC-PUB 209 (EE-PUB 1109), Technion, Dec. 1997.

[7] ——, "Optimal inter-copy delay for dual-copy transmissions in ALOHA with no feedback," CC-PUB 234 (EE-PUB 1134), Technion, Jan. 1998.

[8] N. Abramson, "The ALOHA System—Another alternative for computer communications," in *AFIPS Conf. Proc., 1970 Fall Joint Comput. Conf.*, vol. 37, pp. 281–285.

[9] N. F. Maxemchuk, "Dispersity routing," in *Proc. Int. Conf. Commun.*, 1975, pp. 41.10–41.13.

[10] M. O. Rabin, "Efficient dispersal of information for security, load balancing, and fault tolerance," *J. ACM*, vol. 36, pp. 335–348, Apr. 1989.

[11] Y. Birk and N. Bloch, "Prioritized dispersal: A scheme for selective exploitation of redundancy in distributed systems," in *Proc. 8th Israeli Conf. Comput. Syst. Software Eng*, (*ISySE'97*), June 1997, pp. 77–85.

[12] Y. Birk, "Random RAID's with selective exploitation of redundancy for high performance video servers," in *Proc. NOSSDAV '97*, St. Louis, MO, May 1997, pp. 77–85.

[13] E. S. Sousa and J. A. Silvester, "Spreading code protocols for distributed spread-spectrum packet radio networks," *IEEE Trans. Commun.*, vol. 36, pp. 272–281, Mar. 1988.

[14] S. W. Kim and W. Stark, "Optimum rate Reed-Solomon codes for frequency-hopped spread-spectrum multiple-access communications systems," *IEEE Trans. Commun.*, vol. 37, pp. 138–144, Feb. 1989.

[15] M. B. Pursley, "Frequency-hop transmission for satellite packet switching and terrestrial packet radio networks," *IEEE Trans. Inform Theory*, vol. IT-32, pp. 652–667, Sept. 1986.

[16] E. Lutz, "Slotted ALOHA multiple access and error control coding for land mobile satellite networks," *Int. J. Satellite Commun.*, vol. COM-10, pp. 275–281, 1992.

[17] L. Cooper and M. W. Cooper, *Introduction to Dynamic Programming.* New York: Pergamon, pp. 31–44.

[18] A. B. Carlelial and M. E. Hellman, "Bistable behavior of Aloha-type systems," *IEEE Trans. Commun.*, vol. COM-23, pp. 401–409, Apr. 1975.

[19] W. A. Rosenkrantz and D. Towsley, "On the instability of the slotted ALOHA multiaccess algorithm," *IEEE Trans. Automat. Contr.*, vol. AC-28, pp. 994–996, Oct. 1983.

Yitzhak Birk (S'82–M'86) received the B.Sc. (cum laude) and M.Sc. degrees from the Technion in 1975 and 1982, respectively, and the Ph.D. degree from Stanford University in 1987, all in electrical engineering.

From 1976 to 1981, he was a Project Engineer in the Israel Defense Forces. From 1986 to 1991, he was a Research Staff Member at IBM's Almaden Research Center, where he worked on parallel architectures, computer subsystems and passive fiber-optic interconnection networks. From 1993 to 1997, he also served as a Consultant to Hewlett Packard Labs in the areas of storage systems and video servers. His research interests include computer systems and subsystems, as well as communication networks. He is particularly interested in architectures for information systems, including communication-intensive storage systems (e.g., multimedia servers) and satellite-based systems, with special attention to the true application requirements in each case. The judicious exploitation of redundancy for performance enhancement in these contexts has been the subject of much of his recent work.

Yaron Keren received the B.Sc. (suma cum laude) and M.Sc. degrees in electrical engineering from the Technion—Israel Institute of Technology, in 1993 and 1998, respectively. His thesis was entitled "Judicious use of redundancy for improved performance in ALOHA networks."

From 1993 to 1999, he served an Engineer in the Israel Defense Forces. His current research interests include communications systems.