Networks of Low-Earth Orbit Store-and-Forward Satellites

JOSEPH P. HAVLICEK, Student Member, IEEE
The University of Texas at Austin

JOHN C. MCKEEMAN, Member, IEEE
IBM Technology Products Division

PERRY W. REMAKLUS, JR., Member, IEEE
Virginia Polytechnic Institute & State University

A slotted multiaccess protocol is proposed for networks of low-Earth orbit store-and-forward communications satellites. Networks of this type would provide communication between low cost geographically distributed Earth stations, and would be particularly attractive in areas where conventional terrestrial communications systems were not available. Applications of this type include data acquisition and remote process monitoring. The proposed protocol incorporates time-division multiplexing (TDM) on the downlink, slotted Aloha with collision resolution on the uplink, and an automatic repeat request (ARQ) algorithm. Since the network connectivity is intermittent, analysis of networks of this type is difficult. Nevertheless, relationships among the performance parameters for a general network are deduced, and performance of three particular network configurations is studied via simulation.

Manuscript received April 1, 1992; revised September 1, 1993.

IEEE Log No. T-AES/31/2/08015.

This work was supported in part by a grant from the IBM Corporation.

Authors' current addresses: J. P. Havlicek, Dept. of Electrical and Computer Engineering, The University of Texas, Austin, TX 78712; J. C. McKeeman and P. W. Remaklus, IBM Microelectronics Division, YM7A/B060, 3039 Cornwallis Rd., Research Triangle Park, NC 27709.

0018-9251/95/$4.00 © 1995 IEEE

I. INTRODUCTION

Satellites at altitudes of 200 km to 1200 km are considered low-Earth orbiting, and are generally referred to as *LEO satellites*, or simply *LEOs*. During the last decade, LEOs have been used as inexpensive platforms for experiments in near-Earth space and for low bandwidth packet switched communications. Here we propose a multiaccess protocol for LEO store-and-forward communication which allows a set of geographically dispersed Earth stations to send messages to one another using a constellation of one or more spacecraft.

In Section II, we discuss the multiaccess protocol in detail. We treat the network analytically in Section III, while a simulation evaluation of the network performance is presented in Section IV. Conclusions are reserved for Section V. The remainder of Section I provides an introduction to this class of spacecraft and an overview of packet switched store-and-forward communications.

A. Introduction to LEO Satellites

LEO satellites are a low-cost means of achieving worldwide communications. They typically have an orbital period of about 90 min, and cover every point on the Earth at least twice each day. The range of commercially useful altitudes for LEOs is constrained below due to atmospheric drag: the expected lifetime of a communication satellite in a 300 km circular orbit is only one to three months. Although the first communication satellites were LEOs, Clarke orbit spacecraft quickly achieved industry dominance. This was largely a consequence of the fact that inviews between low-Earth orbiting platforms and terrestrial stations are inherently short in duration. But LEOs offer two distinct advantages over geostationary spacecraft: they have dramatically lower costs throughout the spacecraft lifecycle and are compatible with cost-efficient terrestrial communication equipment.

Commercially useful real-time communication is not feasible with a single satellite in circular orbit because the time in line-of-sight, or inview duration, is typically on the order of 15 min. Furthermore, the satellite footprint is small. But the savings as compared with geostationary satellites are great, both in the cost of ground stations and in the cost of the satellites themselves. The spacecraft construction and deployment costs of currently existing LEOs have ranged from less than $500,000 to about $2,000,000, and construction time is usually less than one year. A common deployment mechanism has been the get away special cannister, or GAS CAN, available aboard the NASA space shuttle. Satellites deployed by this method are often referred to as CANSATs, and the typical launch cost is on the order of a few tens of thousands of dollars. Since the space shuttle orbits at an approximate altitude of only 300 km, most CANSATs require an onboard propulsion system to reach their final orbit. More information about this launch mechanism is available in [1]. A number of LEOs have also been deployed from expendable launch vehicles. The presentation costs of these vehicles are typically tens of millions of dollars, so it is essential that LEOs deployed in this way be piggybacked on larger missions so that the deployment costs can be limited to the regime of GAS CAN launch costs.

LEO satellites typically weigh between 10 kg and 70 kg, and may be as small as 23 cm on a side. Solar cells and lead acid batteries are normally used for satellite prime power, and typical power budgets range from 5 W to 75 W. While several LEOs have been unstabilized, the more sophisticated ones have employed magnetorquer coils and gravity gradient booms for active stabilization. Due to the nature of their orbits, LEOs experience nodal regression and apsidal rotation which cause the orbit to change slowly with time. Stationkeeping is typically accomplished via an electrothermal propulsion system.

For less than $5,000, an Earth station suitable for communicating with LEOs can be constructed from a small computer, a modem, an RF transmitter, and an omni-directional antenna. Such stations are small enough to be transported by an individual, and can be powered by batteries or solar cells. Since the RF path is much shorter than that associated with geostationary satellites, a transmitter power of 10 W is often sufficient to achieve a bit error rate (BER) on the order of 10^{-5} at a typical data rate of 9,600 bits/s. Hence LEOs are ideal in applications where compact, inexpensive Earth stations are desirable. Such applications include remote sensing of oil and gas pipelines, soil moisture, and snow pack readings, as well as search and rescue communications, navigation, and store-and-forward communications. LEOs are particularly attractive when manned or unmanned Earth stations are required in locations that are characterized by a harsh or hostile environment, especially where conventional terrestrial communication links such as telephone lines are not available.

B. Store-and-Forward Communications

A typical single-hop LEO store-and-forward communication system comprises one spacecraft and one or more geographically distributed terrestrial stations. Some or all of the terrestrial stations may be manned, in which case a subscriber wishes to send messages to and receive messages from other stations in the network. Some or all of the terrestrial stations may also be unmanned remote terminals; their function is to monitor some remote processes and report data to the satellite for dissemination throughout the network. A station wishing to transmit a message through the network must wait until it has an inview with the spacecraft. When this occurs, the message is transmitted across the uplink. The satellite maintains the message in onboard storage until the destination station comes into view, at which time the message can be delivered across the downlink.

Amateur radio enthusiasts have employed LEO systems of this type for packet switched store-and-forward communications since the mid 1980s, and synopses of several of the more notable projects are readily available [2, 3]. These spacecraft have onboard message storage capacities ranging from less than 100 kBytes to over 10 MBytes. Uplink and downlink frequencies are typically in the UHF and VHF bands, and data rates range from 1,200 bits/s to 9,600 bits/s. Very simple uplink multiaccess protocols such as pure Aloha [4] have typically been implemented in order to minimize complexity in the Earth stations.

We reiterate that such single spacecraft LEO satellite systems are inherently incapable of providing real-time communications between geographically dispersed terrestrial stations: the satellite footprints are small as compared with those of geostationary spacecraft, and instances where multiple Earth stations have simultaneous inviews with the spacecraft are usually extremely short in duration. Furthermore, from an economical standpoint it is infeasible to use a LEO satellite network for communicating between (fixed location) stations that are in close proximity to one another. Existing telephone networks offer greater capacity at a lower cost. Clearly, LEO satellite store-and-forward communication systems are of interest only when data must be transmitted between geographically dispersed points where minimization of the end-to-end network delay is not a critical factor.

An obvious extension to the single-hop, single satellite LEO store-and-forward communication network is a network employing multiple spacecraft. Multiple spacecraft LEO networks have been previously hypothesized in the context of dense, survivable networks for real-time packet switched voice and data communications [5, 6]. These networks require large numbers (i.e., hundreds) of highly sophisticated satellites connected through real-time spacecraft-to-spacecraft communication links. Hence the important advantages of having a small constellation of low cost satellites are lost in LEO networks of this type. In this paper, we concentrate on multiple spacecraft single-hop store-and-forward networks. These are essentially no different from single spacecraft store-and-forward networks, except that higher throughput can be achieved with less delay per packet due to the increased number of inviews occurring in the network.

II. NETWORK AND MULTIACCESS PROTOCOL

The nodes in the proposed network are Earth stations and LEO satellites. We assume a set of Earth stations that are widely dispersed geographically and a small number of satellites (i.e., less than ten). We further assume that the communication need is low bandwidth in nature; *viz.* that the total number of bytes to be transmitted is small and that, within reason, minimization of end-to-end delay is not important. These assumptions are certainly unusual in modern communications engineering; some of the situations in which they apply were discussed at the end of Subsection IA.

Fixed size data packets arrive at the Earth stations and are destined for other Earth stations. The task of the network is to deliver the packets to their destinations. The network employs datagram routing where individual packets are routed independently. This implies that the packets of a multipacket message may be received at the destination Earth station out of order and at different times. The task of message accumulation is wholly assumed by the receiving Earth station.

Each satellite has unique uplink and downlink frequencies that are distinct from those of the other satellites. Each Earth station has a transmitter that is capable of being tuned to one of the uplink frequencies at any given time. Each Earth station also has a receiver that may be tuned to any one of the downlink frequencies at any given time. Time is divided into slots, and packets are transmitted only on slot boundaries. The Earth station transmitters and receivers incorporate numerically controlled oscillators so that their tuning can rapidly be changed from one frequency to another on a slot-by-slot basis.

Since LEO satellite signals are subject to significant Doppler shift, as well as rapidly changing propagation delay, maintenance of synchronization between all network nodes is not entirely trivial given that the Earth station receivers must be able to tune to various spacecraft from one slot to the next. Consequently, the Earth stations must store precomputed tables which allow them to track the Doppler shift in real time. Furthermore, the slot times must be long enough to accommodate the longest propagation delay in the network. The slot boundaries must also be padded with enough idle time to accommodate disagreements between the time bases of the individual spacecraft, which can be periodically resynchronized by any one of a number of techniques (including reception of a highly precise, centrally located beacon). Unfortunately, incorporation of a highly accurate (i.e., more accurate than IRIG) time base into each satellite is at odds with our goal of minimizing the spacecraft cost: at the extreme, the cost of a cesium clock could easily exceed that of the rest of the spacecraft! The accuracy of the spacecraft time base must be traded against performance in the network design. Poor accuracy implies that large amounts of idle time must be padded at the slot boundaries. To improve throughput, this means that the packet size should be large. Very large packets will result in greatly reduced throughput when packets are lost due to collisions, nodes going out of view during the transmission of a packet, and uncorrectable random transmission errors. In the worst case, performance of the protocol could be worse than that of unslotted protocols if poor clocks were used.

Earth stations receive arriving packets and place them in a waiting queue in local memory. Each Earth station stores a table of inviews from which it can determine when each satellite is in view. When one or more satellites come into view, the Earth station attempts to transmit packets to a satellite. Transmission scheduling of outgoing packets is according to a FIFO discipline. If multiple Earth stations are simultaneously in view of a satellite, they must compete for the uplink channel bandwidth. If two Earth stations simultaneously transmit a packet to the same satellite, then a collision occurs and both packets are lost.

Satellites also have knowledge of the network instantaneous connectivity, so they can determine at any time which Earth stations are in view. Upon receiving a packet from an Earth station, a satellite stores the packet in its onboard memory until the destination station comes into view. For each Earth station, the satellite maintains a FIFO queue of packets that are waiting to be downlinked to that station. When a destination Earth station comes into view, the satellite will attempt to transmit the packet at the head of the corresponding queue. Downlinked packets are received by the destination Earth station if the station's receiver is tuned to the downlink frequency of the transmitting satellite.

A. Network Protocol

The proposed multiaccess network protocol is discussed in the remainder of Section II. In this subsection, the network ARQ scheme is also described. Since the connectivity of the network is intermittent and continuously changing, fixed assignment time-division multiple access (TDMA) or frequency-division multiple access (FDMA) techniques cannot be employed. Note also that the possibility exists for two Earth stations to be simultaneously in view of a satellite, but yet not able to receive each other's transmissions. This might occur, e.g., if there were a mountain range between the two Earth stations. Hence CSMA techniques also cannot be used. Clearly, some form of demand access protocol with distributed control must be devised.

In terms of ARQ, we note that one favorable feature of the network is that the propagation delays are short compared with those associated with geostationary satellites. We propose a form of the

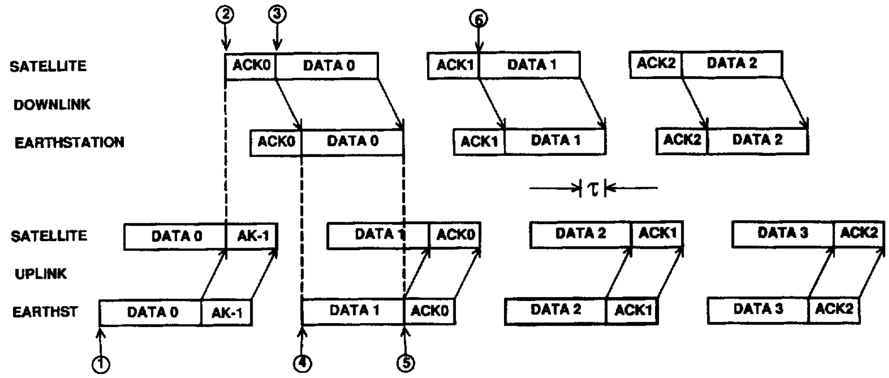


Fig. 1. Space-time diagram of protocol depicting communication between single Earth station and single satellite. At time circle-1, Earth stations send data. After propagation delay of at most τ, satellites receive data by no later than time circle-2, and subsequently send ACKs. Earth stations receive ACKs by time circle-4, and proceed to send data packets. Satellites transmit data packets at time circle-3. These are received by Earth stations by no later than time circle-5, at which time the Earth stations send ACKs. Satellites receive the ACKs by time circle-6, and proceed to send data.

stop-and-wait algorithm for use on both the uplink and downlink [7]. A space-time diagram depicting the network protocol for communication between an Earth station and a satellite is shown in Fig. 1, where transmissions occur simultaneously on the uplink and downlink. The transmission time τ is taken equal to the longest propagation delay in the network. All received data packets are acknowledged with an acknowledgment (ACK) packet that is half the length of the data packets. Upon correctly receiving a data packet in one slot, a node will acknowledge that data packet during the next ACK minislot. Once transmitted, a data packet becomes backlogged until it has been acknowledged. A transmitting node will continue to retransmit a data packet in successive slots until the receiving node goes out of view or until an ACK is received for the packet. Due to the tremendously long packet delay inherent in store-and-forward networks of this type, end-to-end acknowledgments are not feasible.

B. Downlink

The downlink protocol is simple slotted time-division multiplexing (TDM). Downlink frames transmitted by a given satellite are received by all Earth stations that are currently in view of the satellite with receivers tuned to the satellite downlink frequency of the satellite. Each downlink frame contains a data minislot which may hold a data packet addressed to one of the Earth stations that is in view (provided that at least one such packet is stored in the satellite onboard memory). Packets are downlinked addressed to Earth stations selected in round robin order from among those Earth stations which are currently in view and are the destination for at least one packet in the satellite onboard memory. Downlink frames also contain an ACK minislot. If a data packet was correctly received by the satellite during the previous uplink slot, then the ACK minislot contains an ACK for that packet. Prepended to the downlink frame is a header which contains network broadcast information and an announcement identifying the satellite. A trailer containing a cyclic redundancy check (CRC) may also be appended to the end of the downlink frame. The computing time required for processing a CRC on the downlink is not shown in Fig. 1.

A downlink frame is received by the destination Earth station if the receiver of the station is tuned to the downlink frequency of the satellite during the slot in which the frame is transmitted. Normally, for each slot on the downlink, each Earth station randomly chooses one satellite from among those currently in view (with them all equally likely) and tunes its receiver to the downlink frequency of that satellite. However, during the next downlink slot after an Earth station transmits a data packet on the uplink, the Earth station tunes its receiver to the downlink frequency of the satellite to which the data packet was sent.

Note that there can never be a packet collision on the downlink since each satellite has a unique downlink frequency. However, data packets may be lost on the downlink if the receiver at the destination Earth station is tuned to the frequency of a different satellite during the slot in which the packet is sent, if the destination Earth station goes out of view during the transmission of the packet, or if an uncorrectable random transmission error occurs. Besides uncorrectable random transmission errors, ACK packets on the downlink are lost only if the destination Earth station goes out of view, because this station will have sent a successful packet to the satellite in the preceding slot, and hence will have its receiver tuned to the correct frequency and will be awaiting the acknowledgement.

C. Uplink

The uplink multiaccess protocol is slotted Aloha [8] with a modified form of binary exponential backoff for collision resolution. Recall that each Earth station maintains a queue of packets waiting for uplink. If more than one Earth station transmits a packet on a given frequency in a given slot, then a collision occurs and all of the packets are lost. A data packet may also be lost if the receiving satellite goes out of view during the transmission of the packet or if an uncorrectable random transmission error occurs. Successfully received packets are acknowledged in the next ACK minislot on the downlink, so a transmitting Earth station knows whether or not a packet was successfully received after a delay of 2τ plus the transmission time of an ACK packet (see Fig. 1).

Consider the case where only one satellite is in view. For this case, the proposed collision resolution algorithm is no different from ordinary binary exponential backoff [9]. An Earth station with a packet to transmit transmits that packet in the next slot with probability q_r, where each station maintains its own value for q_r. Initially, $q_r = 1$ for all stations. If the downlink slot immediately following the uplink slot in which an Earth station transmits a packet does not contain an ACK for that packet, then the transmitted packet becomes backlogged and q_r is divided by two. Hence q_r is reduced by successive powers of two until the packet is correctly received and acknowledged, or until the satellite goes out of view. When either of these events occurs, the local value of q_r is reset to unity. In the former case, the acknowledged packet is also unbacklogged and removed from the uplink queue of the Earth station. In the latter case, the backlogged packet remains in the queue until the next inview.

With respect to Fig. 1, an Earth station might transmit data packet 0 at time circle-1. Suppose that a transmission error (or collision) occurs. At time circle-2, the satellite will not have received a successful packet. Hence it will not send an acknowledgment. By time circle-4, the Earth station will know that its transmission was unsuccessful. The value of q_r will be halved, and data packet 0 will be retransmitted at time circle-4 with probability q_r.

If more than one spacecraft is in view, an Earth station with a packet to transmit will select one of the currently in view spacecraft at random (where each is equally likely) and transmit the packet at the head of the queue to that spacecraft during the next uplink slot with probability q_r. Suppose that this event occurs in uplink slot k. If the transmission is unsuccessful, then the packet is backlogged, q_r is reduced by $\frac{1}{2}$, and the spacecraft is labeled as being *blocked*. In uplink slot $k+1$, the Earth station will, with probability q_r, again transmit the backlogged packet to the blocked satellite. However, if there is at least one satellite that is in view and not currently blocked, then in uplink slot $k+1$ the Earth station will, with probability $1-q_r$, instead transmit the backlogged packet to one of the currently in view unblocked satellites chosen at random (with each being equally likely). If this results in a successful transmission, then $q_r = 1$. Otherwise, $q_r = \frac{1}{2}$ and the newly chosen satellite is blocked.

Consider a simple scenario involving two satellites (S_a and S_b) and two Earth stations (E_a and E_b). Suppose that E_a and E_b both try to transmit an unbacklogged packet to S_a in slot k. This results in a collision. Under pure binary exponential backoff, both Earth stations would reduce q_r to $\frac{1}{2}$ and attempt to retransmit to S_a in slot $k+1$. Hence in slot $k+1$ one packet would be successful with probability $\frac{1}{2}$, and otherwise no packet would be successful. However, one can easily verify that under the modified algorithm *two* packets are successful in slot $k+1$ with probability $\frac{1}{2}$, and otherwise no packets are successful. The generalization of this analysis to higher numbers of nodes is simple. The reason for blocking satellites where the current packet has been unsuccessful is to prevent occurrence of the event where E_a and E_b both attempt to transmit to S_a in slot k, to S_b in slot $k+1$, to S_a in slot $k+2$, and so forth.

Following every data minislot on the uplink is an ACK minislot. During this ACK minislot an Earth station which correctly received a data packet in the preceding data minislot on the downlink will acknowledge that packet. Hence, like transmitting Earth stations, transmitting satellites know whether or not their packets were successful after a delay of 2τ plus the transmission time of an ACK packet. Of course, a CRC could also be implemented in the downlink frame, and the time for computing the CRC is not shown in Fig. 1.

III. PERFORMANCE

In this section we discuss the performance of the protocol proposed in Section II. We begin by briefly looking at correctness and stability. There are two factors that distinguish this protocol from the usual stop-and-wait approach. First, as was implied in the previous section, a node which sent a data packet in slot k will interpret the absence of an ACK for that packet prior to slot $k+1$ as a NAK. Hence a transmitting node receives either an ACK or a NAK by the next slot after any data packet is transmitted, and timeouts do not have to be implemented. Secondly, we observe that due to lost ACK packets, multiple copies of a single data packet may make their way through the network and end up being received by the destination node. So sequence numbers are essential for correct message accumulation at the destination node. However, use of the customary modulo 2 sequence numbers is not satisfactory due to the intermittent nature of the network connectivity. This presents no problem, however, since a number of schemes (including timestammping the first transmission of a packet) can be used to generate sequence numbers that will allow for identifying multiple copies of a single packet at the destination node. Given a satisfactory algorithm for generating the sequence numbers, correctness of the protocol follows immediately from the usual correctness arguments for stop-and-wait ARQ.

We feel that stability analysis by the usual methods is not particularly informative for this protocol, since some of the assumptions that normally go into such analyses are poorly suited to the physical reality of the LEO store-and-forward network (e.g., no buffering or infinite set of nodes). However, we should point out that in the simplest case and over short time intervals, the uplink protocol could reasonably be cast as being roughly equivalent to normal slotted Aloha with binary exponential backoff. Under the usual assumptions, both of these have been shown unstable for all positive arrival rates (see, e.g., [10]). Nonetheless, both techniques have been used successfully in real networks since their inception.

Next, we consider the evaluation of network performance. We express time in terms of *slot* units equal to the difference between the transmission times of subsequent uplink frames. The length of a slot is time invariant for any given network, and is the same as the difference between the transmission times of subsequent downlink frames. In Fig. 1, the length of a slot is the difference between time circle-4 and time circle-1.

Besides stability, the network performance parameters we are interested in are S, the expectation of throughput expressed in packets per slot, and D, the expectation of delay per packet expressed in slots. These are both examined as functions of the offered load G, which is the expectation of the total number of attempted transmissions per slot by all Earth stations. In addition, we examine G as a function of the aggregate arrival rate λ, which is expressed in packets per slot. If multiple copies of a given packet are received at the destination Earth station, only the first such reception contributes to the throughput S.

Unfortunately, the intermittent nature of the network connectivity renders analytic treatment of these performance parameters a daunting task. To understand why this is so, consider that over various finite non-zero time intervals the network connectivity is fixed. We let T_{start} be the beginning time of network operation and T_{stop} be the least upper bound for the ending time of network operation. For technical reasons, we consider that the network is operating during the half-open time interval $[T_{\text{start}},T_{\text{stop}})$. We let U be the set of all half-open time intervals over which the network connectivity is fixed: $U = \{u = [t_a,t_b) : T_{\text{start}} \leq t_a < t_b < T_{\text{stop}},$ *and the network connectivity does not change during* $u\}$. Now assuming $T_{\text{stop}} < \infty$, we can choose a unique finite disjoint set $O = \{o_1,o_2,\ldots,o_N\}$, $O \subset U$, such that $\bigcup_{i=1}^{N} o_i = [T_{\text{start}},T_{\text{stop}})$. To construct O, we begin by choosing o_1 to be the largest $u \in U$ such that $T_{\text{start}} \in u$ (where a set B is *larger* than a set A if B contains A). Next, we select a time $t_2 \in [T_{\text{start}},T_{\text{stop}})$, $t_2 \notin o_1$, and choose o_2 to be the largest $u \in U$ such that $t_2 \in u$. We repeat this procedure until we have all $t \in [T_{\text{start}},T_{\text{stop}})$ in some $o_i \in O$. By virtue of the existence and finiteness of O, we think of the network connectivity as being piecewise constant in time.

Under a suitable set of simplifying assumptions, we can certainly characterize the network performance during each disjoint time interval in the set O. But in terms of deriving a meaningful characterization of the steady state performance of the network as a whole, there are at least two difficulties with this approach. First, the connectivity is intimately dependent upon the particular configuration of the network under consideration (i.e., upon the number and locations of the Earth stations in a particular network and upon the number and orbital parameters of the LEO spacecraft). Secondly, for any network configuration complicated enough to be of practical interest, the number of elements in O is great enough that we do not know of a means by which readily interpretable analytic results can be obtained. The best we can hope for is to evaluate the performance parameters numerically for a given network. Our feeling is that this approach is no more illuminating than simulation.

Despite the fact that we cannot solve for the network performance analytically, we can deduce a few relationships that should exist between the performance parameters. We enumerate these relationships in the following list of conjectures, each of which is accompanied by an informal supporting argument. In each case we assume a store-and-forward network with some small number of LEO spacecraft and a set of geographically distributed Earth stations such that most nodes do not have an inview during most slots.

C1. *If* $D < \infty$, *then the network is stable. Otherwise, it is unstable.* This is simply the definition of stability.

C2. *If* $S < \lambda$, *then the network is unstable.* Under the hypothesis, the number of packets in the network grows without bound. Instability follows upon application of Little's Theorem and C1.

C3. $S \leq G$. If there are no collisions, no uncorrectable transmission errors, and no packets lost due to satellites going out of view before an uplinked packet can be successfully received and acknowledged, then the average number of packets successfully uplinked per slot is exactly G. Otherwise the number is less than G, since some packets must be transmitted more than once. The average number of packets correctly arriving at their destinations per slot cannot exceed the average number of packets that are successfully uplinked per slot.

C4. *If $G < \lambda$, then the network is unstable.* This follows immediately from C3 and C2.

C5. *$G < \lambda$ when $\lambda > N_I$, where N_I is the average number of Earth stations per slot that are in view of at least one satellite.* This follows immediately from the fact that Earth stations will not attempt to transmit unless at least one satellite is in view. Note that subsequent application of C4 implies instability under the hypothesis.

C6. *If the network is stable, then $\lambda \ll N_e$, where N_e is the number of Earth stations in the network.* In general, most Earth stations are not in view of any satellite during most slots. So $N_I \ll N_e$. Hence, if we do not have $\lambda \ll N_e$, we do not have $\lambda < N_I$. Then the network is unstable by C5 and C4.

C7. *If the network is stable, then $\lambda \ll N_s$, where N_s is the number of satellites in the network.* At most one packet can be successfully downlinked by each satellite in each slot. But since most satellites are not in view of any Earth station during most slots, the actual average number of packets which are successfully downlinked per slot must be much less than N_s. Furthermore, S cannot exceed the average number of packets that are successfully downlinked per slot, and in general S will be less than this number since some successfully dowlinked packets will be duplicates of packets that have already been successfully downlinked. So if we do not have $\lambda \ll N_s$, we do not have $S \geq \lambda$, and the network is unstable by C2.

C8. *If uncorrectable random transmission errors are negligible and the network is stable, then $G \approx \lambda$ for small λ.* Stability and C4 imply $G \geq \lambda$. For λ sufficiently small, the probability of a collision on the uplink is small. Furthermore, if arrival times are independent of slot number, then the probability of a satellite going out of view before an uplinked packet can be successfully received and acknowledged is also small. Since these are the only two factors that tend to increase G, it follows that $G \approx \lambda$.

C9. *If there is a stable region of operation where $G \approx \lambda$, if λ_0 is the supremum of all such λ, and if the network is stable for some $\lambda > \lambda_0$, then there is a stable region of operation where $G > \lambda_0$.* Consider a network operating under the conditions of C8. Then $G \approx \lambda$. By hypothesis there exists $\lambda_1 > \lambda_0$ such that the network is stable. Furthermore, $G \not\approx \lambda_1$ when the arrival rate is λ_1, since λ_0 is at least as large as any λ for which this is true. Then C4 and stability imply $G > \lambda$. Intuitively, we can imagine the network operating under the conditions of C8. A slight increase in λ does not cause the network to become unstable, but does increase the probability of a collision. If we continue to increase λ by small amounts and yet remain stable, we will eventually have a small but sustained rate of collisions. Since any collision implies that some packets must be transmitted more than once, the offered load must exceed the arrival rate in this case.

IV. SIMULATION

In this section we describe simulations that were used to approximate performance for three instances of a representative network. The instances differed in the number of LEO spacecraft they employed. The network configuration is described in Subsection IV-A, while the performance data are presented in Subsection IV-B.

A. Representative Network

The representative network comprised 15 terrestrial stations and up to five LEO satellites. The Earth station locations are shown in Table I; they were chosen solely for their interest to the authors. The station locations were fixed. The presence of mobile terrestrial stations, while in principle supported by the proposed multiaccess protocol, significantly complicates the problem faced by each node in determining the instantaneous network connectivity. Hence increased processing sophistication beyond that proposed here would be required in each network node to support mobile Earth stations (for more on LEO networks with mobile terrestrial stations, see [11]). The orbital parameters for the five satellites in the representative network were taken from five actual LEO store-and-forward spacecraft, and are shown in Table II. The epoch is expressed in days and years, the mean angular velocity in deg/hr, and the altitude in km. Inclination, right angle of ascending node, and argument of perigee are all expressed in degrees, while object number, eccentricity, and mean anomoly are dimensionless. These satellites were chosen because they afford maximal coverage over North America and Europe from among the set of actual deployed LEO satellites that might have been chosen.

Although the units of time used in the performance parameters are slots, simulation of the representative network required that an explicit relationship between slots and real time be fixed. This was because the spacecraft ephemerides had to be expressed in units of real time, and these in turn were required for describing the network instantaneous connectivity. We chose a data rate of 9,600 bits/s for all network links, a data packet length of 134 bytes, and an ACK packet length of 67 bytes. We ignored framing overhead to yield a frame length of 201 bytes.

From Table II, the greatest altitude assumed by a satellite is 887 km. Due to fading and scintillation, transmissions between an Earth station and a LEO satellite cannot begin immediately when the satellite appears over the horizon and is in view of the Earth station. In practice, a reliable communication link can be established only after the satellite reaches some minimum elevation angle, typically 10°. Using a value of 6.37×10^3 km for the mean radius of the

TABLE I
Earth Station Locations

Earthstation	Location	Long	Lat	Altitude (ft)
1	London England	359.92	51.50	86
2	Poughkeepsie NY	286.08	41.70	155
3	Washington DC	284.17	38.83	50
4	Manassas VA	282.53	38.77	321
5	Raleigh NC	281.37	35.78	363
6	Boca Raton FL	279.90	26.28	17
7	Blacksburg VA	279.57	37.22	2033
8	Atlanta GA	275.60	33.82	1050
9	Dallas TX	263.20	32.82	512
10	Boulder CO	254.73	41.00	5350
11	San Jose CA	238.13	37.20	65
12	Fujisawa Japan	139.50	35.33	270
13	Rome Italy	12.50	41.83	230
14	Frankfurt Germany	8.67	50.17	328
15	Madrid Spain	3.67	40.33	2150

TABLE II
Satellite Parameters

Name	OSCAR 9 (UoSAT 1)	NOAA 6	NOAA 10	OSCAR 11 (UoSAT 2)	NOAA 9
Obj. No.	12888	11416	16969	14781	15427
Epoch	122.01634 88	121.08428 88	118.52149 88	122.10823 88	118.46336 88
Incl	97.6192	98.4957	98.6805	98.0570	99.0970
RA of AN	153.5280	124.7956	150.0494	186.0015	92.5302
Ecc	0.00022939	0.0010713	0.0014489	0.00112957	0.0016484
Arg Per	142.3667	238.8091	19.8210	170.9199	42.5035
M Anom	217.7796	121.20141	340.3528	189.22231	317.7413
M Mot	15.32574821	14.25137317	14.22566485	14.62291026	14.115778512
Rev No	36551	45905	8357	22301	17379
Alt km	468-497	805-830	808-846	679-717	845-887
Satellite	1	2	3	4	5

Earth, this gives 2.54×10^3 km as the longest path in the network. At the speed of light in a vacuum, this implies that the network must be designed with value of $\tau \geq 8.5$ ms. This value $\tau = 8.5$ ms was used throughout the simulations, which are described in the next subsection.

B. Results

We simulated one-, three-, and five-spacecraft networks. All 15 Earth stations shown in Table I were present in all simulations. In each case, the period of simulated network activity was at least one week,

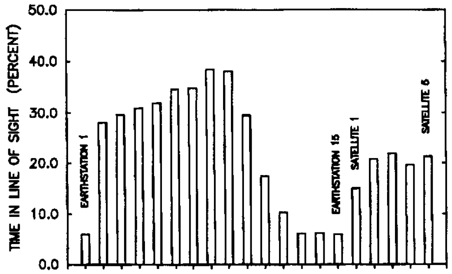


Fig. 2. Percent time in line-of-sight data for five-satellite network. Abscissa coordinates correspond to network nodes, which are listed in increasing order from left to right, Earth stations before satellites. For each node, percent time in line-of-sight is fraction of time that the node is in view of at least one node of opposite type (terrestrial or spaceborne).

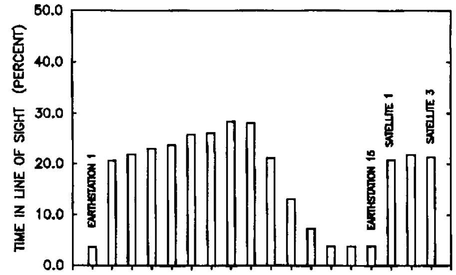


Fig. 4. Percent time in line-of-sight data for three-satellite network. Satellite numbers in this figure refer to designation of each satellite within three-satellite network, and do not correspond to satellite numbers in Table II. Within three-satellite network (and referring to Table II), Satellite 1 is object 11,416, Satellite 2 is object 16,969, Satellite 3 is object 15,427.

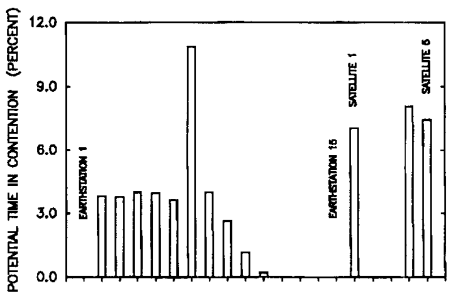


Fig. 3. Potential percent time in contention data for the five-satellite network. Ordinate value for each node is fraction of time that the node is in view of two or more nodes of opposite type (terrestrial or spaceborne).

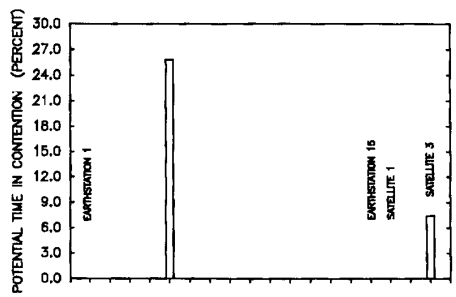


Fig. 5. Potential percent time in contention data for three-satellite network.

beginning at midnight Greenwich mean time on May 9, 1988. We assumed packet arrivals at each of the 15 Earth stations according to 15 independent Poisson processes. These were not identically distributed, and the Poisson parameter for the ith Earth station was determined according to

$$\lambda_i = \frac{\lambda I_i}{\sum_{j=1}^{15} I_j}$$

where λ was the total aggregate arrival rate and I_k was the total length of time during which the kth Earth station had an inview with at least one satellite. The destinations for packets arriving at the ith Earth station were assigned by choosing one of the other 14 stations at random. We assumed uncorrectable random transmission errors to be negligible, so that frames were lost only due to collisions, nodes going out of view during transmission of a packet, and recipient nodes failing to be tuned to the transmitting node's frequency.

Fig. 2 shows the percent time in line-of-sight data for all nodes in the five-satellite network computed for one week of simulated activity. The height of the bar corresponding to a given node is the percentage of the time during which the given node had an inview with at least one node of the opposite type (i.e., Earth station or satellite). Fig. 3 is the potential time in contention data for the five-satellite network. In this case, the height of the bar corresponding to a given node is the percentage of the time during which the given node was simultaneously in view of two or more nodes of the opposite type. The satellite numbers shown in Fig. 2 and Fig. 3 correspond to those in the last row of Table II.

The percent time in line-of-sight and potential time in contention data for the three-satellite network are shown in Figs. 4 and 5, respectively. The satellite numbers in these two figures refer to the designation of the satellite within the three-satellite network, and are not the same as those shown in the last row of Table II. Satellite one in the three-satellite network was (referring to the second row of Table II) object 11,416, Satellite 2 was object 16,969, and Satellite 3 was object 15,427. The significant reduction in the potential time in contention (as compared with the five-satellite case) is a direct consequence of the reduction in the total number of inviews occurring in the network.

The percent time in line-of-sight data for the one-satellite network is shown in Fig. 6. The single spacecraft, designated Satellite 1 within the one-satellite network, was object 16,969. For this network, there are obviously no times when more than one satellite is in view of a single Earth station.

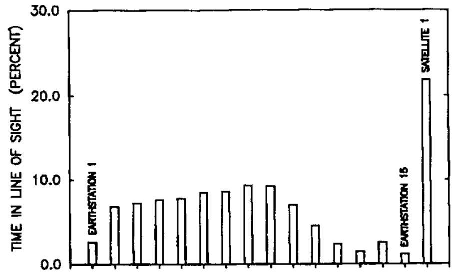


Fig. 6. Percent time in line-of-sight data for one-satellite network. Satellite 1 in one-satellite network is object 16,969 (see Table II).

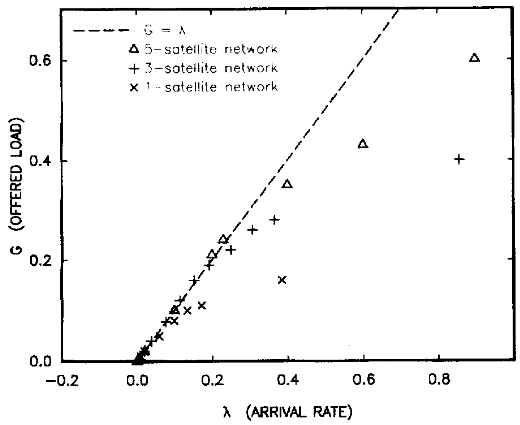


Fig. 7. Offered load as function of arrival rate for all three simulated network instances. Units are packets per slot for both axes. Operation below $G = \lambda$ line is unstable.

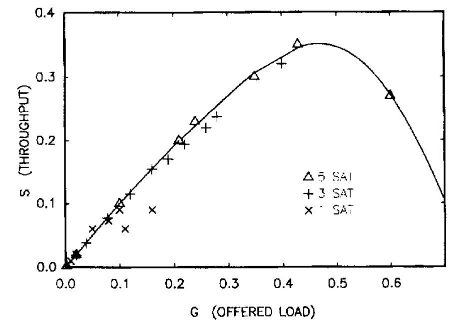


Fig. 8. Throughput as function of offered load for all three simulated network instances. Units of both axes are packets per slot. This characteristic is property of protocol, and does not depend on specific network geometry.

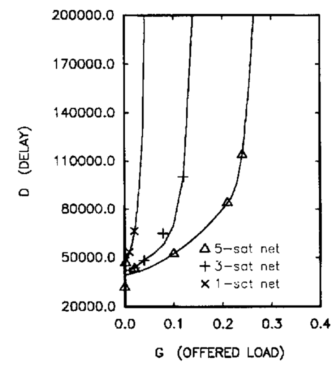


Fig. 9. Delay per packet as function of offered load for all three simulated network instances. Offered load is expressed in packets per slot, while delay is expressed in slots.

The potential time in contention for object 16,969 is unchanged from the three-satellite case, which was depicted in Fig. 5.

Fig. 7 shows G as a function of λ for all three simulated networks. The dashed line is the locus of $G = \lambda$. According to C4 in Section III, any operating point below this line is unstable. In particular, the data corroborate C6 and C7. Although we were not able to simulate arrival rates with fine enough granularity to exactly locate the onset of unstable operation, the data do indicate that the one-satellite network is unstable for all $\lambda \geq 0.060$, the three-satellite network is unstable for all $\lambda \geq 0.193$, and the five-satellite network is unstable for all $\lambda \geq 0.400$.

The three- and five-satellite networks clearly display the three-region behavior predicted by C8, C9, and C5. In particular, $G \approx \lambda$ for small λ. As λ grows, the networks remain stable with $G > \lambda$ until the arrival rate exceeds the maximum that can be supported, at which point the network becomes unstable and operation falls below the $G = \lambda$ line. The one-satellite network goes unstable almost immediately as λ is increased from zero. In all three cases, we expect the λ—G characteristic to asymptotically approach the value of N_l for the network.

The throughput data for all three networks is shown in Fig. 8. Note that throughput as a function of offered load is a property of the protocol, not of the particular network under simulation. Finally, the delay data are shown in Fig. 9. As expected, D increased rapidly as the network approached instability. We did not limit the amount of local message storage in the Earth stations, and this rapid increase was primarily a result of the uplink queue lengths at each of the Earth stations tending toward unboundedness as the arrival rate approached and exceeded the maximum realizable offered load.

V. CONCLUSIONS

We considered the use of networks of LEO communication satellites for providing packet switched store-and-forward communications between geographically distributed terrestrial stations. One advantage of using satellites of this type is that they offer low costs throughout the spacecraft lifecycle. We proposed a slotted multiaccess protocol employing TDM on the downlink and slotted Aloha with modified binary exponential backoff for collision resolution on the uplink. We also proposed the use of stop-and-wait ARQ on both the uplink and the downlink. The slotted scheme requires that an accurate time base be implemented in the spacecraft, and the cost of this time base must be traded against performance in the system design.

Networks of this type are perhaps ideally suited to applications where low throughput communications are needed for compact, inexpensive, unmanned Earth stations performing data acquisition and remote process monitoring in harsh or hostile environments where conventional terrestrial communication systems are not available and minimization of the end-to-end network delay is not critical. Since the RF path associated with low orbit satellites is much shorter than that associated with geostationary satellites, the Earth stations can incorporate low power, low cost transmitters, receivers, and antennae.

Performance analysis of LEO satellite networks is extremely difficult due to the intermittent nature of the network connectivity, which is determined by the exact configuration of the specific network under consideration. Consequently, we did not present a complete analytical characterization of performance. We did, however, deduce several relationships that should exist between the performance parameters of any LEO satellite network employing the proposed protocol.

We argued that such networks can be stable only for arrival rates that are much less than the number of Earth stations in the network and much less than the number of spacecraft in the network. In general, the former condition will be the weaker of these two. We predicted that the network should display stable operation with the offered load about equal to the aggregate arrival rate when the arrival rate is small, that operation should remain stable with the offered load greater than the arrival rate as the arrival rate increases, and that the network should eventually become unstable with the offered load smaller than the arrival rate as the arrival rate is increased beyond the rate at which the Earth stations have opportunities to transmit a packet. The factor which limits the rate of transmission opportunities is the network geometry, since inviews between any given Earth station–satellite pair are sparse in time. That these predictions are reasonable was borne out when we simulated three instances of a representative network. In the simulation results, we observed that network operation was stable for arrival rates approximately 20 times smaller than the number of satellites in the network. In practice the numerical value in this relation will be heavily dependent upon the network configuration; in this case the observed value agreed qualitatively with our general analytic prediction (C6 and C7). Comparison of the slotted scheme with an unslotted approach would make an interesting future study.

ACKNOWLEDGMENTS

The authors wish to gratefully acknowledge numerous valuable consultations with Dr. C. W. Bostian, Dr. T. Pratt, and Dr. W. L. Stutzman of the Bradley Dept. of Electrical Engineering, Virginia Tech, all of whom provided wise opinions and insightful suggestions.

REFERENCES

[1] Megill, L. R., Redd, F. J., and Ridemour, R. W. (1987) Get away special class satellites (CANSATS): design constraints, applications, and recent technology advances at Utah State University. AIAA Paper 87-0583, Jan. 1987.

[2] Karn, P. R., Price, H. E., and Diersing, R. J. (1985) Packet radio in the amateur service. *IEEE Journal on Selected Areas in Communications*, **SAC-3**, 3 (May 1985), 431–439.

[3] Diersing, R. J., and Ward, J. W. (1989) Packet radio in the amateur satellite service. *IEEE Journal on Selected Areas in Communications*, **7**, 2 (Feb. 1989), 226–234.

[4] Abramson, N. (1970) The Aloha system—another alternative for computer communications. In *Proceedings of the Fall Joint Computer Conference, AFIPS Conference Proceedings*, **37**, Montvale, NJ: AFIPS Press, 1970, 281–285.

[5] Brayer, K. (1984) Packet switching for mobile Earth stations via low-orbit satellite network. *Proceedings of the IEEE*, **72**, 11 (Nov. 1984), 1627–1636.

[6] Chakraborty, D. (1989) Survivable communication concept via multiple low Earth-orbiting satellites. *IEEE Transactions on Aerospace and Electronic Systems*, **25**, 6 (Nov. 1989), 879–889.

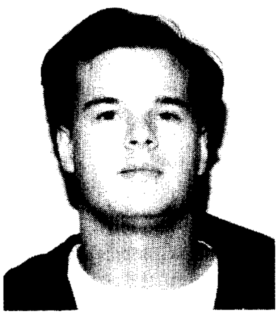
[7] Gray, J. P. (1972) Line control procedures. *Proceedings of the IEEE*, **60**, 11 (Nov. 1972), 1301–1312.

[8] Takagi, H., and Kleinrock, L. (1985) Throughput delay characteristics of some slotted Aloha packet radio networks. *IEEE Transactions on Communication*, **COM-33**, 11 (Nov. 1985) 1200–1207.

[9] Metcalf, R. M., and Boggs, D. R. (1976) Ethernet: distributed packet switching for local computer networks. *Communications of the ACM*, **19**, 7 (Jul. 1976) 395–404.

[10] Bertsekas, D., and Gallager, R. (1987) *Data Networks*. Englewood Cliffs, NJ: Prentice-Hall, 1987.

[11] Richharia, M., Hansel, P. H., Bousquet, P. W., and O'Donnell, M. (1989) A feasibility study of a mobile communication network using a constellation of low Earth orbit satellites. In *IEEE Globecom '89*, Dallas, TX, Nov. 27–30, 1989, 773–777.



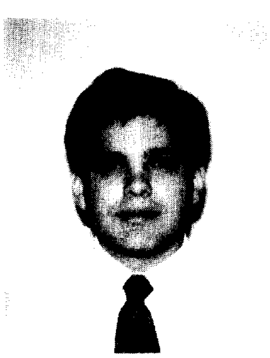
Joseph P. Havlicek (S'84—M'88—S'91) was born in West Lafayette, IN, on April 7, 1964. He received the B.S. degree from Virginia Polytechnic Institute & State University, Blacksburg, in 1986, and the M.S. degree at the same institution in 1988, both in electrical engineering. Since 1990, he has been pursuing the Ph.D. degree at the University of Texas at Austin.

From 1984 to 1987, he was with Management Systems Laboratories, Blacksburg, VA, as a software engineer. From 1987 to 1989, he was affiliated with SFA, Inc., Landover, MD, and since 1987 he has been with the Naval Research Laboratory, Washington, DC, where his work has included high performance signal and image processing. Since 1993, he has also been a programmer–analyst with Ralph Kirkley Associates, Austin, TX, working on-site in the multimedia division of IBM, Austin. In 1990 he was a recipient of the Department of the Navy Award of Merit for Group Achievement, and he was the winner of the 1992 University of Texas Engineering Foundation Award for Exemplary Engineering Teaching while Pursuing a Graduate Degree. His research interests include varied topics in communications systems, signal processing, and computer vision.

Mr. Havlicek is a member of Tau Beta Pi, and Phi Kappa Phi.

John C. McKeeman (S'83—M'87) received the B.S. degree in electrical engineering from Virginia Tech in 1980, the M.S. degree in electrical engineering from the Air Force Institute of Technology, Dayton, OH in 1981, and the Ph.D. degree in electrical engineering from the University of Dayton, Dayton, OH in 1985.

He is currently with IBM's PowerPC Embedded Processors group in Research Triangle Park, NC. From 1991 to 1992 he was with IBM Federal Sector Division, Mannassas, VA, and prior to joining IBM he was an Associate Professor in the Department of Electrical Engineering at Virginia Tech. During 1986 he was affiliated with Harris Corporation, Melbourne, FL, and from 1981 to 1984, he was a signal analyst with the Foreign Technology Division, Dayton, OH. His research interests include computer architecture and satellite communications.



Perry W. Remaklus (S'87—M'91) Was born in Stamford, CT, on June 2, 1965. He received the B.S. degree and the M.S. degree in electrical engineering from Virginia Polytechnic Institute & State University, Blacksburg, in 1987 and 1991, respectively.

From 1983 to 1987, he was a software consultant to Willmann-Dell, Inc., Richmond, VA. From 1989 to 1993 he was with the Satellite Communications Group of the Electrical Engineering Department at Virginia Tech, where his work has included data acquisition and control systems for low-earth orbit and geostationary satellites, as well as computer aided modeling and analysis of atmospheric propagation data for the Olympus and ACTS experiments. Presently, he is with IBM in Research Triangle Park, NC, as an applications engineer in the field of embedded PowerPC RISC controllers.

Mr. Remaklus is a member of Eta Kappa Nu, and Tau Beta Pi.