A Design Concept for Reliable Mobile Radio Networks with Frequency Hopping Signaling

ANTHONY EPHREMIDES, FELLOW, IEEE, JEFFREY E. WIESELTHIER, MEMBER, IEEE, AND DENNIS J. BAKER

Invited Paper

The design of a packet radio network must reflect the operational requirements and environmental constraints to which it is subject. In this paper, we outline those features that distinguish the High Frequency (HF) Intra Task Force (ITF) Network from other packet radio networks, and we present a design concept for this network that encompasses organizational structure, waveform design, and channel access. Network survivability is achieved through the use of distributed network control and frequency hopping spread-spectrum signaling.

We demonstrate how the execution of the fully distributed Linked Cluster Algorithm can enable a network to reconfigure itself when it is affected by connectivity changes such as those resulting from jamming. Additional resistance against jamming is provided by frequency hopping, which leads naturally to the use of code division mutiple access (CDMA) techniques that permit the simultaneous successful transmission by several users. Distributed algorithms that exploit CDMA properties have been developed to schedule contention-free transmissions for much of the channel access in this network. Contention-based channel access protocols can also be implemented in conjunction with the Linked Cluster network structure. The design concept presented in this paper provides a high degree of survivability and flexibility, to accommodate changing environmental conditions and user demands.

I. Introduction

There are several packet radio (PR) networks, and each has followed a different design route that is dictated by the special requirements imposed upon it. In this paper we consider an example of a packet radio system that is of special interest to the U.S. Navy. For this system, a design concept has been formulated over the last six years, as reported widely in the literature [1]–[7]. It is our intent here to provide a brief but comprehensive description of the main design issues of this network, which is known as the High Frequency (HF) Intra Task Force (ITF) Communication Network. The HF ITF Network is being designed to provide extended-line-of-sight (ELOS; 50 to 1000 km) communication for a Naval task force, and it must use the HF (3- to 30-MHz) band.

The features that distinguish this network from other packet radio (PR) networks stem from the special operational requirements and environmental constraints that it must satisfy. Briefly, these consist of the following:

1) A relatively small number of nodes (2 to 100), which are mobile and whose connectivities may change dynamically as a result of variable propagation conditions and interference, must remain connected.

2) These nodes must be able to communicate to points both within and outside the network under a variety of operating modes and scenarios.

3) The network must be secure, must resist a variety of jamming threats, and must be able to degrade gracefully under conditions of stress.

4) Both voice and data communications must be supported, including point-to-point and broadcast transmissions, while accommodating various precedence levels.

The *number and nature* of the nodes, the need for *transmission security*, and the use of the *HF band* stand out as the three main factors that distinguish the packet radio network from the other systems described elsewhere in this issue.

The relatively small number of platforms that constitute the ITF Network is a basic characteristic that permits design choices that are incompatible with the large and highly variable number of nodes that characterize, for example, the DARPA PR nets and SURAN.

The design of the HF ITF Network has been greatly influenced by the properties of the HF channel, which is a fading and dispersive medium with propagation occurring via both groundwaves and skywaves. The HF ITF Network will rely primarily on the use of groundwaves to connect nodes. In addition to its being more predictable than skywave propagation, groundwave propagation has the advantage of having little dispersion; the use of groundwaves thus permits considerably greater signal bandwidths, and therefore data rates, than does the use of skywaves. HF

Manuscript received December 15, 1985; revised July 20, 1986. A. Ephremides was associated with the Naval Research Laboratory when the research discussed in this paper was performed. His work was supported in part by the Office of Naval Research under Grant N00014-84-K-0614.

A. Ephremides is with the Department of Electrical Engineering, University of Maryland, College Park, MD 20742, USA.

J. E. Wieselthier and D. J. Baker are with the Naval Research Laboratory, Washington, DC 20375-5000, USA.

0018-9219/87/0100-0056$01.00 © 1987 IEEE

Table 1 Comparison of SURAN and HF ITF Network

	SURAN	HF ITFN
Number of nodes	• hundreds to thousands	• tens (<100)
Frequency band	• UHF (300–3000 MHz)	• HF (3–30 MHz)
Link communication range	• LOS	• ELOS
Data rate	• 400 kbits/s	• < 10 kbits/s/link
Spectrum spreading technique	• direct sequence	• frequency hopping
Network structure	• *n*-hop cluster architecture, with superclusters, etc.	• 1-hop cluster architecture (dynamic)
Type of multiple access channel	• single channel	• multiple channel (via CDMA)
Channel access	• random access	• adaptive TDMA and random access

groundwaves are well-suited for ITF communication because they can provide the required ELOS propagation ranges of up to several hundred kilometers. Also, HF groundwave propagation is only minimally affected by severe atmospheric disturbances. The attenuation of HF groundwaves varies as a function of frequency; typically, the achievable communication range decreases as frequency increases. We shall demonstrate in Section II how our approach to networking takes advantage of this apparent shortcoming of the HF medium.

The HF band has relatively small bandwidth compared to the UHF part of the spectrum. Typically, data rates over individual links may be limited to 2400 bits/s. As a result, the communication protocols for channel access and the mode of spread-spectrum signaling chosen for the higher frequency, wider bandwidth nets are inappropriate for the HF ITF Network. How these factors play a crucial role in making fundamental choices of network design will be explained in detail later. For the moment we simply want to emphasize that certain differences in requirements and constraints may lead to substantially different architectures. In Table 1 we juxtapose some of the crucial characteristics of SURAN and the ITF Network. The limited data rate supported by the HF channel will also make it difficult to use packetized voice, and therefore to achieve true voice/data integration. In this paper, we concentrate wholly on data networks, one function of which may be to establish dedicated voice links.

Another aspect of PR network design is that although detailed and accurate performance analysis is difficult (as documented elsewhere in this issue), there is evidence of a critical dependence between performance (in terms of throughput and/or delay) and message traffic statistics. Thus in making certain design choices (such as switching or access protocols), it is useful to have as detailed a traffic pattern as possible. Unfortunately, in the domain of the intended application area of the HF ITF Network, traffic statistics are not readily available. As a consequence, the degree of confidence placed in certain of the design features chosen is necessarily reduced.

Finally, it is worth mentioning some of the other networks that are or have been under design and development, whose purpose and characteristics resemble those of the ITF Network but are *not* discussed elsewhere in this issue. These include the Advanced Mobile Phone Service (AMPS) Network [8], the Battlefield Information Distribution (BID) Network [9], and the Ptarmigan communication system [10]. AMPS is a commercial radio network that is organized into cellular tessellations with a fixed (not mobile) local controller in each cell. BID is based on an adaptation of the PRNET concepts to a military environment. It employs a hierarchical structure and relies on distributed control. Finally, the Ptarmigan system, designed for NATO environments, employs both radio and wire communication links, and assumes infrequent topological changes for certain nodes in the network. These networks are sufficiently different in scope, requirements, and constraints from the ITF Network so that a new design concept approach has been used for the ITF Network.

The remainder of this paper is organized as follows. In Section II, we develop our approach toward the definition of this network organization and control architecture. In so doing we follow the ISO layered concept that permits a modular identification of the design issues and simplifies somewhat the overall task of network design. This modular approach leads naturally to the issue of radio channel access, which is discussed in Section III. Following that we introduce the issue of transmission security and the consequent use of spread-spectrum signaling. We point out the new degrees of freedom (and difficulties) that arise with the use of spread-spectrum modulation and the implications of both channel access and organizational control. We focus attention on the use of spread-spectrum signaling in Section IV and then return to the question of link activation, which is the subject of Section V. We discuss several algorithms for link activation, that actually define the state of the art in that area. We close with Section VI, in which we present some preliminary thoughts about routing and higher level protocol and architectural issues. That section summarizes the design concept in the context of the overall work on mobile radio networks and allows us to draw certain conclusions about the main accomplishments, their limitations, and the natural next steps of our work. Finally, the conclusions Section VII describes the future of the ITF Network project.

II. Network Organization and Control

An HF ITF Network has no *a priori* structure. That is, we do not assume a given connectivity structure nor do we preassign roles to nodes for the purpose of network control. Instead, a network organization algorithm is provided whose purpose is to discover existing network connectivities and dynamically assign the roles that nodes are to assume in network control. In this section we describe how the HF ITF Network implements this design concept.

The ISO seven-layer architecture model is a widely accepted, useful framework for network design and provides a natural basis for the description of the HF ITF Network design concept. The ITF Network consists of mobile nodes that are geographically dispersed over distances that may be beyond line-of-sight, as well as the links that are used to connect these nodes. The only knowledge available to each network member, prior to the execution of the network organization algorithm, is the set of possible network members, each of which has a unique identity. In particular, a node has no *a priori* knowledge of the locations of other nodes, the connectivities of the network, nor even of its own neighbors. Thus the first task of network design is to provide the means by which the existing connectivities among the nodes can be discovered. This is a basic issue that resides in the Physical Layer of the ISO model.

Once this is achieved, we proceed to the next step of determining how to transform the discovered connectivities, which along with the radio equipment of the nodes constitute merely raw transmission facilities, into reliable communication links. This is almost by definition the paramount issue residing in the second or Data Link Layer of the ISO model. Actually, in a radio network in which the communication medium is a multiple-access, broadcast channel the situation is somewhat complicated by the potential interference between simultaneously transmitted signals. This causes the boundaries that define the Link Layer in the ISO model to fade. Thus new issues arise, such as when and how to "activate" or "enable" the links determined by the discovered connectivities; this is to say that switching, which is a level-3 (Network Layer) issue in the OSI model, suddenly seems to merge with the Link Layer issues of how to modulate and code the signals.

Assuming the issue of link utilization gets resolved, the next natural question is how to route messages through the network whose destinations are more than one "hop" away, which is clearly a Network Layer issue.

Following that, the questions of internetting (layer-3), end-to-end connection management (layer-4), session establishment (layer-5), encryption (layer-6), and other higher level issues, must be addressed. In our design concept, we focus on the first (bottom) three layers. As we describe our approach to answering the first question of discovering the network connectivities, we will see that our choices are guided by considerations that reside in higher levels and by overall network requirements. Thus our first algorithm that permits the nodes to discover each other, yields at the same time an overall network organization structure that can be flexibly used to accommodate the overall network operation. The first algorithm achieves not merely "connectivity determination," but is actually an algorithm for network organization and control. Because the interconnection structure that results from this algorithm consists of node-clusters that are suitably interconnected, we call it the *Linked Cluster Architecture*. The details of the Linked Cluster Algorithm (LCA) used to produce this structure have been amply documented in the literature [3]–[5], so that, here, we will limit ourselves to a brief description.

We start from the premise that, as links are discovered, a structure should be formed that will permit:

a) communication between any pair of nodes,
b) network-wide broadcasts,
c) avoidance of the "hidden-terminal" problem [11],
d) robust recovery from node or link losses and other topological changes.

Furthermore, we require that the process of organization starts up in a fully distributed fashion. Even though distributed algorithms (for almost any purpose) often possess little-understood subtleties and hard-to-overcome difficulties, it is of paramount importance for the survivability, robustness, and graceful degradation requirements of the ITF Network to insist on freedom from reliance on any single node for network control.

The Linked Cluster Architecture consists of node clusters that are linked to each other as shown in Fig. 1. Nodes can

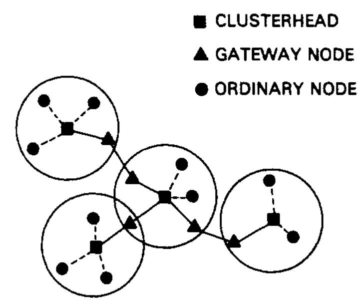


Fig. 1. Example of Linked Cluster organization of a mobile radio network.

assume one of three roles; a node may be a *clusterhead*, a *gateway*, or an *ordinary node*. Each node is associated with a designated clusterhead, called its *own* clusterhead (which may be the node itself), but it may also be within communication range of other clusterhead nodes. The clusterhead *may* act as a local controller for all the nodes belonging to its cluster. For example, intra-cluster communication may be controlled by the clusterhead acting either as a polling agent for noncontention communication, or as a "busy-tone" emitter in a contention-based mode. These choices, however, need not be made now. We are simply noting here the potential nature of the clusterhead node's role. As a result of the procedure used to designate nodes as clusterheads in the LCA, it is not possible for two clusterheads to be directly connected to each other. Gateways are used to provide the communication paths that are needed to connect one cluster to another, thereby providing overall network connectivity. The clusterheads and gateways and the links that interconnect them form the *backbone* network.

The basic philosophy behind the LCA is to invest some initial communication overhead to develop a flexible linked cluster structure that exploits the existing connectivities in a natural way to facilitate multihop path formation, broadcasting, and channel access, while retaining sufficient distributed control to permit local routing around defective or otherwise undesirable parts of the structure.

The LCA consists of two logical steps: cluster formation and cluster linkage. These constitute the "computation" part of the algorithm. To execute that part of the algorithm, it is also necessary to have a "communication" part during which nodes broadcast certain short messages that permit the listening nodes to collect the necessary database to execute the computation part. However, it is logically convenient to describe the computation part first.

1) Computation Part (Cluster formation and linkage): Cluster formation and linkage is best illustrated by an example in which the entire network topology is known to a central controller and the communication range is a fixed constant R for all nodes. The central controller selects arbitrarily one node, say node 1, to be a clusterhead, draws a circle of radius R around that node, and declares all nodes captured in that circle to belong to the first cluster and to have node 1 as their *own* clusterhead. From among the nodes not in the first cluster, the central controller chooses arbitrarily the node with the lowest ID number to be the next clusterhead. The controller draws a second circle of radius R around the new clusterhead and declares all nodes within this second circle to be members of the second cluster. However, only those nodes that were not already in the first cluster are assigned the new clusterhead as their *own* clusterhead. The process repeats until every node belongs to at least one cluster and has its own clusterhead. In practice, a procedure to determine the network structure is implemented in a distributed fashion, without a central controller, without assuming a fixed communication radius R, and without the assumed global connectivity knowledge. We will see presently how this can be done, as we describe the communication part of the algorithm.

Once clusterheads have been determined, gateway nodes can be designated. Two cases need to be considered, namely, overlapping clusters (as in Fig. 2(a)) or adjacent, nonoverlapping clusters (as in Fig. 2(b)). In the latter case,

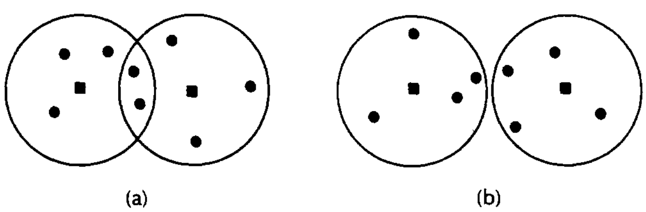


Fig. 2. Examples of the two cases that arise when linking clusters: (a) overlapping clusters, and (b) adjacent, nonoverlapping clusters.

a pair of gateways is needed, i.e., one from each cluster. In either case there is an unambiguous subset of nodes from which gateways can be selected. For example, the node in the intersection of the two clusters in Fig. 2(a) with the lowest ID number can be selected or the pair of nodes in Fig. 2(b) whose sum of ID numbers is the lowest can be chosen gateways (for details see [3]–[5]). Alternatively, all the nodes in the intersection of the clusters shown in Fig. 2(a) can be gateways, and all the node pairs that link the two clusters in Fig. 2(b) can be gateways. Again, the challenge is to implement the selection rules in a distributed way.

To motivate the distributed implementation and to lead toward the communication necessary to achieve it, we note that the cluster formation process just described requires:

a) knowledge of each node's bidirectional connectivities to neighboring nodes,
b) a rule for selection of a clusterhead from a set of candidate nodes,
c) specification of the sequence in which clusters are formed,
d) a rule for gateway selection from a group of candidate nodes.

As it will turn out, each node needs only up to two-hop-away connectivity information (i.e., knowledge of who are its neighbors and its neighbors' neighbors) and knowledge of its neighbors' own clusterheads to implement this procedure in a distributed way.

2) Communication Part (Database collection for distributed implementations): Each node can discover who its neighbors are by the process of probing. When a probe message is broadcast, every node that hears it sends an acknowledgment back to the probing node. A simple strategy for effecting such a probing by every node in the network is to set up two TDMA frames for controlling the transmission of probe and acknowledgment messages. In this approach, each node is assigned its own transmission slot once in each frame, as shown in Fig. 3. During frame 1 each

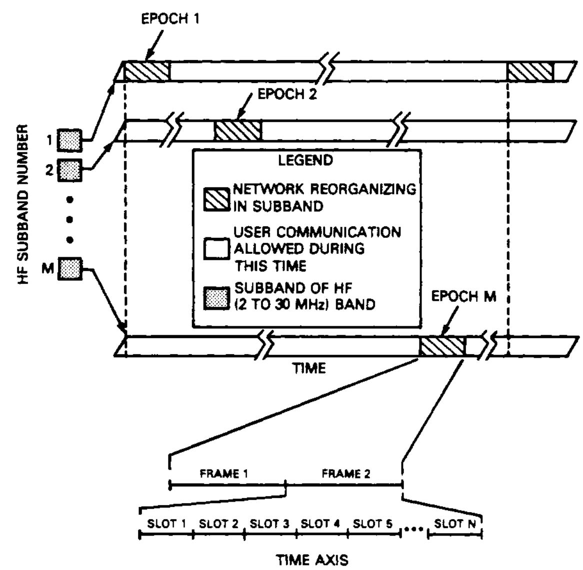


Fig. 3. Timing for network restructuring.

node broadcasts its probe message (for example, announcing its identity) and, at the same time, acknowledges the receipt of previously transmitted probe messages that it has heard (by announcing the IDs of those nodes it has heard from so far). During frame 2 each node's broadcast includes acknowledgments for probe messages that it has received since its own frame 1 transmission. As a result of such probes, each node discovers those nodes to which it is bidirectionally connected [3]–[5]. Only bidirectional links are used to form the backbone network.

Just before its frame 2 transmission, each node has obtained all the information it needs to implement the first step of the Linked Cluster Architecture (that is, cluster formation). It is necessary, however, to include in its frame 2 transmission an announcement of the decision to become a clusterhead, if in fact the node has so decided. A node reveals this decision by including the ID of its own clusterhead in its frame 2 transmission. (Each node selects as its own clusterhead, the lowest numbered head to which it is bidirectionally connected.) A node decides to become a clusterhead if it has not already heard from a clusterhead node to which it is *bidirectionally* connected. Thus node 1 always becomes a clusterhead, and node i ($i > 1$) does so if it is not bidirectionally connected to a clusterhead whose ID is less than i. At the end of the second frame, each node has learned of the existence of all clusterheads that are one hop away and some, but not all, that are two hops away. The execution of this procedure results in the same network structure as that in the centrally controlled example discussed earlier.

In addition to simplicity, an advantage of using a TDMA-based probing message exchange is its contention-free nature and the knowledge of the duration of the organization (or reorganization) period. Network-wide synchronization at the slot level (so that each node knows the slot in which it is to transmit) should be easy to maintain, because the slot duration (perhaps 100 ms) is much greater than the anticipated timing uncertainties throughout the network (several milliseconds). In addition, spread-spectrum synchronization is easier to acquire in contention-free systems than in systems in which contention is permitted. A disadvantage of the TDMA structure is the long duration of the organization period when the number of nodes is large. Therefore, for large networks it may be necessary to develop alternative methods for maintaining linked cluster networks. For example, it may be possible to proceed asynchronously with a contention-based protocol or to allow only a limited subset of nodes to participate in the formation of a backbone network.

With respect to deciding on gateway status, a node faces two difficulties: 1) the identification of the clusters for which the node can serve as a gateway and 2) the identification of all other nodes that can do the same. For nodes that belong to cluster intersections (as in Fig. 2(a)) the selection of a gateway can be made unambiguously, based on the information that it has gathered in frame 2. However, in the case of nonoverlapping but adjacent cluster configurations (as shown in Fig. 2(b)), the algorithms described in [3]–[5] occasionally yield more than a single pair of gateways linking the clusters or possibly an additional "half-pair," that is, one node in one cluster that decides it must serve as gateway along with a node from the other cluster that does not decide the same. The creation of such superfluous gateways is not harmful; it is only a nuisance—their existence has little effect on network operation, and, if desired they can easily be neglected or detected and regulated to nongateway status. So we see that by the proposed simple exchange of messages it is possible to implement in a distributed way the Linked Cluster Architecture for subsequent network operation.

Note that the chosen order of cluster formation produces a set of clusters that is not optimized in any way. It does not achieve, for example, the minimum number of clusters, nor are we sure that such a goal is desirable. Important, however, is that it does produce a connected network, whenever the required link connectivities are present. We also point out that it is not known *a priori* which nodes (other than node 1) will become clusterheads. However, if it is desirable that certain nodes should assume the clusterhead role, then they should be assigned low identification numbers to achieve that goal [3]–[5].

We noted earlier that the communication range of HF groundwaves varies considerably as frequency is varied over the HF band. The proposed network architecture takes full advantage of this apparent shortcoming of the HF band by partitioning the HF band into several subbands (see Fig. 3), each with a bandwidth of a few megahertz over which the communication range for groundwaves is approximately constant. The LCA is run separately for each subband in an assigned epoch, thus producing a set of overlaid connectivity maps that give rise to a set of simultaneously operating networks. The HF ITF Network consists of this set of individual networks that are defined in separate subbands. Typically, there will be a smaller number of larger clusters at the low end of the HF band, where communication range is greatest, and a larger number of smaller clusters at the upper end of the HF band, where communication range is smallest. At most one of these networks will reorganize itself at any time, while the other networks maintain communication using their most recently derived network structure. This capability of preserving a communication structure during the reorganization process adds considerably to the robustness of the HF ITF Network. Also, resistance against jamming is improved, because it would be difficult for a jammer to disrupt all networks simultaneously.

Two additional observations should be made.

a) There is no need to assume anything about communication range; it need not be uniform or symmetric. The algorithm simply discovers which nodes can be heard by a given node in each subband.

b) The algorithm is inherently robust; if a probing message is not received, for example, because of a deep fade or a local jammer, that link is not included in the backbone network. The loss of a potential connectivity during network reorganization does not debilitate the network, and later use of that link is still possible, although not for backbone network communication. If there are undetected errors, however, in the decoding of the received messages about the ID information of neighboring nodes, the algorithm would result in inconsistencies that might reduce the effectiveness of this architecture. However, the use of suitable error detection schemes can render the probability of occurrence of this event negligible [12].

We would like to conclude this section with two comments. First, the proposed distributed algorithm for network operation can appear deceitfully simple or falsely complex. Distributed algorithms are not well understood yet. Often it is not known what are the right questions to ask about them. They can be subtle, and great care must be taken in dealing with them. On the other hand, the direction of network management in military packet radio clearly must go toward distributed implementation, and we feel that our algorithm makes a contribution in this regard. Secondly, now that we have identified the existing connectivities, each of which represents a potential communication link, and have built this identification into an architecture of broader scope, we realize that a highly nontrivial question arises, namely: How should these links be used? How should a node transmit (when, with what signaling format, under what rules) over any of these links, given the distributed nature of the network and the potential of interference? These questions naturally lead us to the second layer of the ISO architecture, which is addressed in the next section.

III. Channel Access

The communication resource available to the ITF Network is the HF channel. This resource must be shared by, and therefore allocated to, the users of the network. The traditional way of allocating bandwidth is via frequency division multiple access (FDMA), under which each user is provided a different portion of the frequency band for its exclusive use. The "time-domain" counterpart of FDMA is time division multiple access (TDMA) under which each user has exclusive rights to the use of the entire channel during predetermined, periodically recurring time slots.

In a multihop network it is possible to reuse the communication resource in distant parts of the network. Thus FDMA allocations may be set up within individual regions of the network, permitting two or more users to use the same frequency channel simultaneously, as long as the other-user signal levels are not great enough to cause destructive interference. Similarly, local TDMA cycles may be established, resulting in what we might call *scheduled transmissions*.

The division of the channel resource in the time domain offers several advantages over doing so in the frequency domain. For example, the intermodulation products that can result from the simultaneous transmission of many signals at the same platform are not a problem when TDMA is used. Also, the use of narrow-band frequency channels would be incompatible with the use of spread-spectrum signaling, which is required for antijam (AJ) performance. In addition, operation in the time domain permits the use of a wide variety of channel access schemes that provide flexibilities that are not possible in the frequency domain.

It is well known that fixed allocation schemes such as FDMA and TDMA do not perform well when traffic is bursty, particularly when the number of users is large. Many of the schemes developed to meet the needs imposed by bursty traffic use contention methods, under which it is possible for two or more users to transmit simultaneously, resulting in destructive interference. The first of these was a pure random access scheme known as ALOHA [13], for which several schemes have since been proposed to stabilize and improve upon the performance [14]. Also, stable collision-resolution schemes have been proposed by Capetanakis [15]. Another approach is the use of reservation schemes [16], which can permit the near-perfect use of the channel resource, as long as some mechanism for making the reservations can be implemented. Finally, hybrid schemes that combine features of two of more of the basic types of schemes have been proposed and studied in the literature [17]. We do not propose to engage here in a comparison of the various classes of methods. Such comparisons abound in the literature [18], and the conclusions are sensitive to various assumptions about traffic statistics.

The Linked Cluster Architecture is extremely flexible in that it permits the use of a variety of channel access methods. In fact, a mixture of two or more schemes may be used; for example, scheduled transmissions over the backbone network and contention-based channel access within the clusters. Such an approach reflects the need to support a robust backbone network, and the fact that backbone network traffic is expected to be relatively *regular*, with clusterheads possibly serving as traffic concentrators. In contrast, much of the other network traffic is expected to be relatively bursty. The division of the channel resource purely in the frequency domain does not offer such flexibility of channel access.

Furthermore, it must be emphasized that there are two very different modes of traffic, i.e., point-to-point (single destination) and broadcast (i.e., multidestination). Clearly, it is best to use scheduled transmissions for broadcast traffic, because if a contention-based channel access scheme were used one would often have to retransmit many times to ensure that each destination received the packet correctly. In contrast, in a scheduled channel access discipline, all the destinations know when the transmission will take place, and thus a single transmission will suffice, as long as the schedules have indeed been generated to ensure that no conflicts occur. One cannot make such a generalization, however, for point-to-point transmissions; the relative performance of scheduled and contention-based schemes will, in general, depend on the network connectivities and traffic requirements. Nevertheless, the task of selecting channel access protocols for the ITF Network is somewhat simplified by the need to use spread-spectrum signaling, as we will presently see.

Although spread-spectrum schemes were invented to enhance jamming resistance and to reduce the probability of interception, it was soon realized that their use also provided multiple access capability. The choice of spread-spectrum signaling and related issues are discussed in greater detail in the next section. Here we want to identify and explore the new degrees of freedom (and accompanying new constraints) with respect to multiple access that are offered by the use of spread-spectrum waveforms.

The multiple access capability of spread-spectrum communications is called code-division-multiple-access (CDMA) and is best explained in the context of the frequency hopping (FH), as opposed to the Direct Sequence (DS), form of spread-spectrum signaling. Consider, for example, the simple case of a certain bandwidth W and N users. We achieve CDMA via FH if we partition W into q frequency bins and assign to each user a frequency-hopping pattern that specifies the sequence, or *code*, of frequency bins in which the user is permitted to transmit during each of the dwell times that make up a time slot (see Fig. 4). If the codes assigned to the different users are orthogonal (i.e., do not overlap in frequency at any time), then CDMA is merely a conceptual equivalent to fixed TDMA or FDMA, except that it requires a much more complex im-

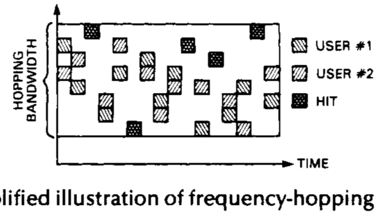


Fig. 4. Simplified illustration of frequency-hopping CDMA.

plementation (and would be justified only because of its antijamming properties). However, CDMA has an inherent flexibility that makes it ideally suited to the needs of the ITF Network. There are only as many orthogonal codes as there are frequency bins. There are, however, virtually unlimited numbers of codes that may overlap to some extent.

In many applications, it is reasonable to assume in the analysis that each of these codes is randomly and independently generated. Forward error control coding is used to handle the loss of data caused by *frequency hits*, i.e., the destruction of the data in individual hops by other users' simultaneous transmission in the same frequency bin. Therefore, it is possible to tolerate several, nonorthogonal, simultaneously transmitted signals without serious performance degradation [19]. Thus a much larger number of users can be accommodated in a CDMA environment than the number of frequency bins would seem to indicate, as long as not too many of them transmit simultaneously. Of course, the penalty to be paid is a gradual reduction in quality that results from the nonorthogonality of the codes. We actually have a *quasi-orthogonality* that is analogous to the true orthogonality that characterizes FDMA and TDMA, and one can now consider operation in the *code domain*. Generally, only a fraction of the theoretically possible total of q users (i.e., one per frequency bin) can transmit at any given time without noticeable degradation. Thus there is a penalty to be paid in the form of efficiency reduction. However, use of spread-spectrum signaling is dictated by the need to provide an antijamming capability; consequently, any degree of multiple access achievable through CDMA is a welcome bonus.

The level of other-user interference that can be tolerated in a CDMA system depends upon the modulation/coding scheme, the receiver implementation, the propagation characteristics of the communication medium, and the criterion that has been established for acceptable performance (e.g., a specified probability of bit error or packet error). For example, in the cases considered in [19], in which Reed–Solomon coding is used to correct errors caused by frequency hits in an otherwise noiseless channel, a reasonable number of users is $q/10$ when all frequency hits are assumed to result in errors, and about twice that when frquency hits can be detected, and the corresponding symbols erased.

We note that even if orthogonal frequency hopping patterns were used, it would generally be impossible to maintain true orthogonality during network operation, unless low hopping rates were used. Fig. 4 actually represents the use of two FH patterns that have been designed to be orthogonal; however, as a result of timing uncertainties at the transmitting platforms, and/or different propagation delays, frequency hits do, in fact, occur.

Let us now compare the properties of the time-domain multiple-access channel with those of the code-domain multiple-access channel. In both cases we shall consider the implications of the choice between scheduled transmissions and contention-based ones. In the time domain, we have a single-user channel. A receiver will not be able to receive a packet correctly if other users in its neighborhood are transmitting, even if their transmissions are destined for someone else. The purpose of scheduling transmissions is to ensure that such destructive interference does not occur, by guaranteeing that at most one packet in a neighborhood transmits at a time. Alternatively, channel access protocols for such channels may be designed to permit such contention, as is often done in bursty traffic environments, in which case packet loss will result, necessitating a mechanism to provide for subsequent packet retransmission.

When we are operating in the code domain, the distinction between scheduled and contention-based channel access gets somewhat blurred. We now have a multi-user channel in which it is possible for several signals using quasi-orthogonal codes to share the channel simultaneously without resulting in destructive interference. There are actually two basic types of other-user interference that must be considered. *Primary conflicts* result when two or more users transmit in the same neighborhood simultaneously using the same code. All packets involved in the collision are normally assumed to be destroyed. In some applications, the capture property of spread-spectrum signaling will permit the first signal to arrive to be received correctly, despite the presence of additional signals using the same code. In an FH system, this implies discrimination against signals that are delayed with respect to the first signal by more than one dwell time. It would be difficult to exploit this property in the HF ITF environment, however, because the hopping rates are relatively low, and as a result the interfering signals will often not be sufficiently separated in time.

Secondary conflicts refer to the other-user interference that results from signals that use codes that are quasi-orthogonal to that of the desired signal. As we have noted, as the number of other users increases, so does the probability of incorrect packet reception. Thus in the code domain, a packet will be received correctly only if it does not suffer any primary conflicts, and if it does not suffer too many secondary conflicts.

Channel access schemes for CDMA systems may actually operate in the joint code/time domain. In that case, the goal of scheduled transmissions should be to avoid primary conflicts and to maintain acceptable levels of secondary conflicts. Contention-based schemes may also be defined, in which the criteria for successful packet reception are as discussed above.

We shall see that operation in the joint code/time domain offers a degree of freedom in the design of channel access protocols that is not possible in the time domain. Again, we may consider channel access protocols that rely on scheduling, as well as those that are contention-based. The choice is no longer clear-cut, because another dimension must now be considered.

In order for a signal that uses a given spread-spectrum code to be successfully received, it is necessary for the receiver, not only to know that code, but also to monitor it at the right time. Thus it is necessary to have a coordination between receiver and transmitter that seems to argue for scheduled transmissions. Actually, the situation is somewhat more complex, so let us explore it further. For the moment, assume that each node is limited to a single transmitter and a single receiver (it may be possible to receive on one code while transmitting on another). This case is not only simpler to consider, but also reflects some realistic physical and equipment constraints. This situation suggests that we consider a third type of interference, which we may call *receiver scheduling conflicts*. If, e.g., two nodes transmit to a common destination simultaneously using different codes (we are assuming here that the destination knows that these nodes are transmitting and also the codes that they are using), the receiver will be able to monitor only one of them, even though there are no primary conflicts and the secondary interference may be sufficiently low. To develop efficient channel access schemes, clearly one must operate in the joint code/time domain, thereby taking into consideration both the times at which packets are transmitted, as well as the codes that are used for this purpose. There are four extreme possibilities of code assignment:

a) Every pair of nodes is assigned an individual code; thus if node 1 wants to transmit to node 2, it must use the specific code assigned to that pair (say $C_{1,2}$) and node 2 must monitor the same code at the same time; a total of $N(N-1)$ codes are needed if there are N nodes.

b) Every node is assigned its own *transmitter-based* code; e.g., when node 1 transmits it always uses code C_1 regardless of the intended receiver's identity; the intended receiver, however, must monitor code C_1 at the same time; a total of N codes are needed.

c) Every node is assigned its own *receiver-based* code; e.g., node 1 always monitors its code (say C_1) regardless of the transmitter's identity, but the transmitter must use the same code C_1 to reach node 1; a total of N codes are also needed.

d) All nodes are assigned the same *common* code; any node wishing to reach any other node uses that code and all nodes monitor the same code all the time; only a single code is needed.

For each of these cases we discuss the issues associated with broadcast and point-to-point traffic and the suitability of scheduled versus contention-based channel access. The intermediate cases of groups of nodes sharing a code in any of the senses mentioned above do not require separate discussion.

The use of codes assigned for pairs of nodes is clearly inappropriate for broadcast traffic. Such codes could be used for point-to-point traffic without scheduling only if users were able to monitor the codes of all their neighbors at all times. However, since we are assuming that only one receiver is available at each node, scheduling is necessary. Furthermore, the use of codes for pairs of nodes cannot handle the case where a transmitting node does not know who can hear it. It does permit, however, simultaneous transmissions between different node pairs, thus taking advantage of the CDMA multiple-user channel capability.

Transmitter-based codes are clearly well-suited for broadcast transmissions, because many nodes can monitor the transmitter-based code; the transmitting node does not even have to know who its neighbors are. Each receiving node must know who is transmitting, however, so that it can monitor the correct code; thus contention-based schemes cannot be implemented, and scheduling of receptions is necessary. It is possible for several nodes to transmit within a neighborhood without causing destructive interference, but each node will be able to monitor only one transmission at a time, unless it has more than one receiver.

Transmitter-based code assignments are also well-suited for point-to-point traffic. Several point-to-point links may be active in a neighborhood simultaneously. As in the broadcast case, scheduling of receptions is necessary to ensure that the receiver is monitoring the correct code.

An important property of transmitter-based codes is that their use precludes primary conflicts, because no two nodes will ever use the same code. Of course, the effects of other users, as secondary conflicts and receiver scheduling conflicts, must be taken into account. Another important property is that of selective reception, which permits a receiver to listen to one of several active transmissions, simply by monitoring the appropriate code. Finally, transmitter-based codes can be used by jammed nodes that can be heard, but cannot hear their neighbors.

The use of receiver-based codes is clearly inappropriate for broadcasting. However, some additional flexibility is offered in the point-to-point case in that either scheduled or contention-based channel access protocols may be defined, unlike the case of transmitter-based codes in which only scheduled access is appropriate. Such code assignments offer a *selective addressing* capability, in that a node directs a transmission to a destination by using the latter's receiver-based code. Simultaneous transmissions by several nodes to different receivers is possible. Primary conflicts occur when two or more nodes transmit to the same destination simultaneously. The use of receiver-based codes cannot handle the case of a jammed node (one that cannot receive) that wants to transmit, unless it knows who its neighbors are; in a dynamic network such information will often not be available.

The use of a single common code is essentially equivalent to operation in the time domain, because a transmission is successful only if no other user in the neighborhood transmits at the same time. (The capture property, mentioned earlier, may sometimes permit one packet to be successful.) Either scheduled or contention-based channel access schemes may be used. Such a code assignment clearly fails to exploit the multiple user capability that is offered by the use of spread-spectrum signaling. However, it may be appropriate for the execution of the Linked Cluster Algorithm, in which each user must monitor the channel throughout the two TDMA frames. As a simple extension of the common code approach, it is possible to associate a local common code with each cluster during normal network operation (i.e., after the organization has been achieved); thus several such codes can be used simultaneously, thereby exploiting the multiple user channel capability of CDMA.

Certainly, hybrid schemes that use combinations of these types of code assignments can be implemented. For example, reservation schemes are often used when it is necessary to maintain high throughput in a bursty traffic environment. First, we may consider the conventional approach, which parallels that in the time domain. Once the reservations are made, either through a contention-based or a contention-free process, the actual data are delivered contention-free (although subject to secondary interference). The code assignments used for both reservation and data traffic will depend on the considerations outlined above. Transmitter-based codes are attractive for the actual data traffic, because their use precludes primary conflicts (e.g., if two nodes think that they are scheduled to transmit at the same time) and permits multidestination monitoring of the transmissions.

Some creative approaches for the design of reservation schemes are possible in the joint code/time domain. For example, Wieselthier and Ephremides [20] have considered a distributed reservation scheme that is especially well-suited for one-way communication, e.g., with a radio-silent destination or one that cannot broadcast reservation schedules because its transmissions are jammed. In another reservation-like approach, Sousa and Silvester [21] have considered a minislotted scheme in which users transmit a preamble using a receiver-based code to get the attention of their desired destination. The transmitter then switches to a transmitter-based code for actual data transfer.

From all these cases it is clear that the transmitter-based code assignment (case b) is the most appropriate for much of the traffic in the ITF Network. It is the only one that permits both broadcast and point-to-point transmissions, handles the case of isolated nodes that cannot hear their neighbors, allows simultaneous transmissions without interference (if scheduled properly), and does so efficiently. Of course, there is no need to make a unique selection of code assignment method for all types of traffic in the entire network. In Section V, where the final choices are presented, it will be seen that a mixture of code assignments is possible and desirable, because it is necessary to incorporate a contention-based channel access mechanism for some of the network traffic. At this point, however, it is clear that we cannot do without the use of transmitter-based code assignments with scheduled transmissions—at least for a good part of the network traffic.

We are at this point ready to focus somewhat closer attention on the issue of spread-spectrum signaling, primarily from the antijamming (AJ) and link communication points of view, rather than from the multiple access point of view. This is done in the next section. Following that, we shall return to the issue of activating (scheduling) the links in the Linked Cluster network structure (Section V).

IV. Spread-Spectrum Signaling and AJ Performance

Spread-spectrum signaling has been proposed for use in the HF ITF Network because it can provide resistance against jamming. In this section we examine the issues associated with the use of spread-spectrum signaling at HF. We show why frequency hopping (FH) is preferable to direct sequence (DS) or hybrid schemes in our application; we discuss the jamming resistance that is achievable over individual links through the use of FH signaling, coding, and diversity; and we demonstrate how the Linked Cluster Architecture provides additional protection against jamming.

Of the two basic alternatives for spread-spectrum communication, namely FH and DS, neither one is universally preferable. There are many comparisons of the two methods in the literature, the most comprehensive of which seems to be available in [22]. Both offer all the advantages of spread spectrum, namely, selective addressing capability, multiple access, privacy and security, signal hiding, high-resolution ranging, and above all, interference rejection (whether caused by jamming, multipath, or other users). Both expand the bandwidth necessary for transmission. They differ in the means by which they achieve it.

In FH systems the transmitter hops from one frequency bin or subband to another, transmitting a narrow-band signal at each hop. Selective addressing can be achieved by the transmitter via the use of the intended receiver's unique FH pattern (or code, as discussed in Section III), and interception is made much more difficult when secure (unpredictable) FH patterns are used. Resistance against jamming is achieved because the jammer cannot put large amounts of energy throughout the entire frequency band, and because it cannot predict the FH pattern. Forward error control coding is used to permit recovery of lost data. (We assume that the hopping rate is sufficiently fast so that repeat-back or follower jamming is not feasible.) In DS systems the data to be transmitted modulate a higher rate, and again unpredictable, pseudonoise code (or chip) sequence, thus producing a signal with an instantaneously large bandwidth. Resistance against jamming is achieved because the jammer's waveform is poorly correlated with the spreading waveform.

The primary criteria by which the selection of spectrum spreading technique should be made are the ability to provide antijamming capability at acceptable data rates, the ability to meet environmental constraints and system requirements, and the practicality of the implementation. In our Network, we also require satisfactory CDMA capability. By considering performance under each criterion, we have concluded that a pure FH system is the most suitable for the HF ITF Network. A complete discussion of the comparison can be found in [7]. Here we summarize the arguments for this choice.

A. Processing Gain—Bandwidth Consideration

The AJ capability of a spread-spectrum system is directly proportional to its "processing gain," which is usually defined to be the ratio between the bandwidth of the transmitted spread-spectrum signal and the data rate in the information baseband channel. We now explain why, when operating at HF, considerably higher processing gains can be achieved in FH systems than in DS systems. The HF groundwave channel can support signal bandwidths that may be as great as 1 MHz; however, the maximum signal bandwidth suitable for skywave paths is usually limited to only about 100 kHz, unless sophisticated equalization techniques are used. Even though the architecture of the HF ITF Network has been designed primarily for the groundwave component, it would be unwise to use a signaling format that could not be used for skywaves. By comparison, FH has no such limitation. Signals can be hopped across the entire HF band. The actual limitation may be of the order of 2 to 5 MHz, however, since we have partitioned the 3- to 30-MHz band into a set of subbands of 2- to 5-MHz bandwidth, each of which supports a separate Linked Cluster configuration as discussed in Section II. Furthermore, the spread bandwidth under FH need not be contiguous, thus permitting certain portions of the band to be allocated for special purposes if necessary—an additional dimension of flexibility.

B. Multipath Considerations

In the HF ITF Network, skywave signals are being considered mainly as a source of multipath interference that is generated unintentionally along with the desired groundwave signal. This interference often results in the fading of the desired signal, as well as in intersymbol interference. Under FH operation, such multipath interference may be eliminated if the hopping rate is sufficiently fast. There are, of course, practical considerations that limit the hopping rate. It turns out [7] that 2400 hops per second (i.e., a rate of about one hop per bit for many applications) suffices to avoid all *F*-layer skywave reflections at ranges of less than 300 km. Since the total elimination of skywave interference cannot be guaranteed in all applications, it is advisable to use a robust signaling scheme that is relatively insensitive to fading; e.g., the use of noncoherent *M*-ary FSK with one *M*-ary symbol per hop and appropriate error control coding is expected to perform well in the presence of this type of interference. Although DS signaling can, in many applications, provide a greater degree of multipath rejection, adequate multipath rejection can be achieved through the use of FH signaling and *M*-ary FSK modulation.

C. Synchronization

It is generally more difficult to achieve synchronization in DS systems than in FH systems. Especially for short-packet traffic, the synchronization overhead incurred because of long preamble bit-strings can be considerable.

D. Multiple Access

Both techniques are capable of providing multiple-access capability through the use of quasi-orthogonal codes and forward error control coding, as discussed in Section III. However, DS systems are highly sensitive to differences in received signal power levels (the "near-far" problem). Most analyses that address the number of DS signals that can share a wide-band channel simultaneously are based on the assumption that the interfering signals are equal in magnitude to the desired signal. If the interfering signals are greater in magnitude, then this number is reduced considerably. In contrast, in an FH system an undesired signal will result in interference only if it simultaneously occupies the same frequency bin as the desired signal.

The considerations we have addressed indicate clearly that FH is to be preferred over DS in the HF ITF Network. We have also considered the use of hybrid FH–DS signaling, which is often used to combine the best features of pure FH and DS, and thereby produce superior results. In such systems the signal transmitted at each hop is no longer narrow-band, but instead it is DS-spread (although by a much smaller factor than in a pure DS system). In our case, the primary benefit would be the improvement of multipath rejection at slow hopping rates. However, a hybrid system is more complex than an FH system, and, like DS, is more sensitive to received power differentials than is FH. We have already noted that *M*-ary FSK signaling is expected to perform sufficiently well, so the added improvement resulting from the use of a hybrid system is not believed to be needed. Therefore, we are led to conclude that pure FH is the most appropriate choice at HF.

The AJ performance of a signaling scheme can be expressed in terms of the maximum jammer to signal power ratio that can be tolerated at the receiver $(J/S)_{max}$, while maintaining acceptable bit error rate (BER). In [7], detailed calculations of AJ performance can be found for signaling parameters that are characteristic of the HF ITF environment. For example, under some reasonable assumptions it is possible to operate links at a jammer to signal power ratio between 25 dB (at a data rate of 2400 bits/s) and 40 dB (at 75 bits/s) for a nominal case of 8-ary FSK signaling, a nonfading channel, 5-MHz spread hopping bandwidth, BER = 10^{-5}, worst case partial band (WCPB) noise jamming, soft decision receiver, and known jammer state (i.e., the receiver can determine which of the received symbols have been jammed). Results for 4-ary FSK are within 0.7 dB of those for 8-ary FSK. When the BER requirement is relaxed to 10^{-3}, and all other conditions are unchanged, $(J/S)_{max}$ is increased by about 1.5 dB. In contrast, when narrow-band (non-frequency hopped) signaling is used with noncoherent FSK signaling, negative $(J/S)_{max}$ values are obtained.

These results indicate that some modest to good AJ capability is possible for most links in the HF ITF Network. However, our estimates of jammer to signal levels at which satisfactory communication can be maintained may have to be reduced somewhat after some issues related to practical implementation are considered [7].

Ultimately, the AJ performance of a link depends not only on $(J/S)_{max}$, but also on the actual received signal and jammer power levels. Therefore, it is difficult to determine what an *adequate* $(J/S)_{max}$ value would be. There is no such thing as a *jam-proof* link, because a jammer with enough power will be able to disrupt any link. The best that can be obtained is jam resistance. Also, the combined effects of other users and jammers must be considered to obtain a more realistic assessment of achievable performance levels.

The overall conclusion from this performance analysis is that although some degree of AJ link capability is practically feasible at HF, the gains are somewhat limited. However, we shall see that in the context of network-wide AJ performance, the modest levels of achievable link protection need not be a limitation. The AJ capability of a link can be enhanced through the addition of relays between communicating nodes. Thus if a link is disrupted by jamming, it may be possible to set up a two-hop path using an intermediate node as a relay. The shorter propagation paths result in a higher signal to jammer power ratio. Such techniques can be used with either narrow-band or wide-band signaling schemes.

An extensive simulation was done with various jamming scenarios and different network configurations, i.e., different placements of nodes. Connectivity in a benign (without jammers) environment was determined by using a standard model for HF groundwave propagation loss and noise values that represent a relatively noisy part of the world. Propagation loss varies considerably with frequency; noise levels vary with both frequency and geographic location. The LCA was then executed to determine the networking structure for several frequencies.

Fig. 5 shows how the effect of a jammer is modeled. We define $d_{j\,min}(d)$ to be the minimum distance at which the jammer does not disrupt communication for a transmitter to receiver distance of d, i.e., the minimum distance at which

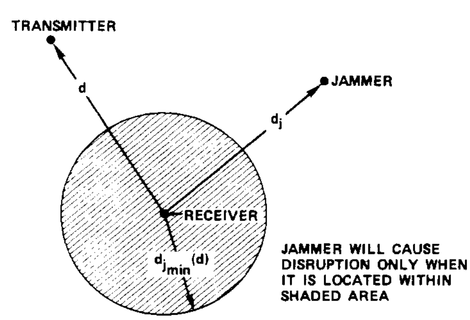


Fig. 5. Transmitter-receiver-jammer geometry.

the specified BER is maintained. A small value of $d_{j\min}(d)$ thus represents a link that is highly resistant to jamming. To determine the size of this region, the $(J/S)_{\max}$ values discussed above are used with the HF groundwave propagation model. In our examples, we have assumed that the jammer, like the desired signal, uses HF groundwave propagation; thus the same propagation loss model is used for the jammer. Figs. 6 and 7 show the relationship between $d_{j\min}$ and d, for the case of groundwave wide-band and narrow-band signaling, respectively, for typical values of system parameters (e.g., modulation, coding, and BER). It is assumed that the jammer's transmitted power is 10 dB greater than that of the desired signal. Despite the greater jammer power, for the case of wide-band signaling we have $d_{j\min}(d) < d$, i.e., the jammer must be closer to the receiver than is the transmitter of the desired signal if communication is to be disrupted; the lower end of the frequency band provides more resistance against jamming in this case. However, for the case of narrow-band signaling we have $d_{j\min}(d) > d$ for the system parameters shown, and the upper end of the frequency band provides more jamming resistance. In general, the relationship between $d_{j\min}(d)$ and d depends on $(J/S)_{\max}$, the jammer's transmitted power advantage, HF signal propagation attenuation, and system losses. These calculations are discussed in greater detail in [6], [7].

It should be emphasized that, in the actual operation of the LCA, none of these calculations are necessary, nor is any model for propagation. The connectivities will be determined by means of the probing messages themselves. When a jammer is present, some previously existing bidirectional links may become one-way links. Such one-way links are considered to be nonexistent in the execution of the LCA, and thus they cannot be part of the backbone network. However, they are clearly useful for some network communication. Protocols for the use of one-way links can be easily superposed on our architecture without changing the basic organizational algorithm; however, we do not address this point further in this paper.

Our simulation studies show graphically the ability of the LCA to combat jamming by dynamically forming a connected network, always structured into node-clusters, using only those links that are not broken by jamming. Furthermore, by operating the algorithm at various connectivities (corresponding to different subbands of HF) we can discover which subbands are more jam-resistant (this turns out to depend on the original configuration and the jamming scenario).

We show here one example (Fig. 8(a)-(i)). More examples and analysis can be found in [6]. For the example shown, we assume 10-MHz operating frequency, 2400-bit/s information rate, and keep the same values for all other signaling parameters as shown on Figs. 6 and 7. Our model gives a communication range of 230 km under benign conditions. The jamming scenario involves simply one jammer traversing the entire region of the task force. We acknowledge that this scenario is not realistic in that a jammer traversing the task force could be easily destroyed. However, this is a good example to consider because it demonstrates the ability of the Linked Cluster Architecture to withstand a threat greater than that which can be anticipated in the real world.

In the absence of jamming, the network discovers through the LCA the connectivities shown in Fig. 8(a). It then organizes itself into the set of node-clusters shown in Fig.

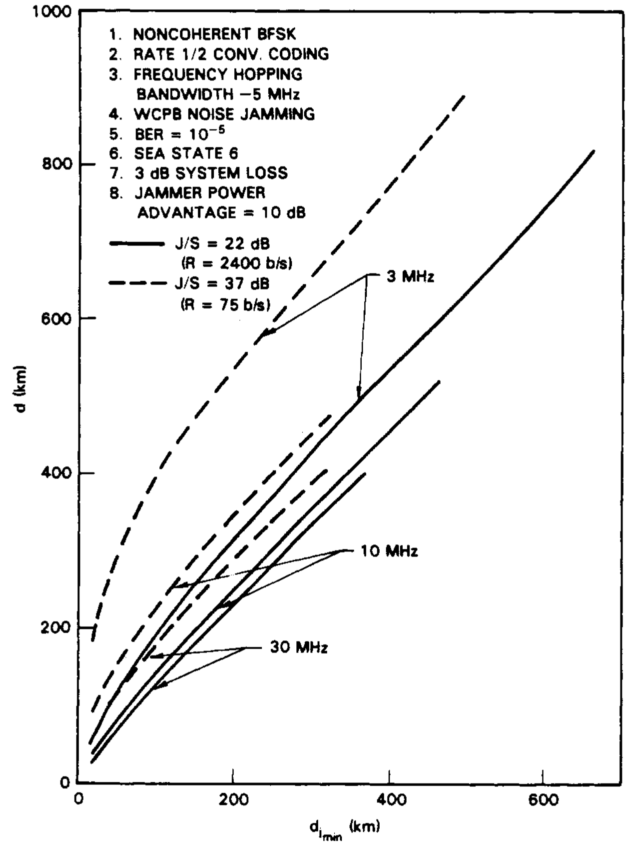


Fig. 6. $d_{j\min}(d)$ versus d for HF groundwave model and wide-band signaling.

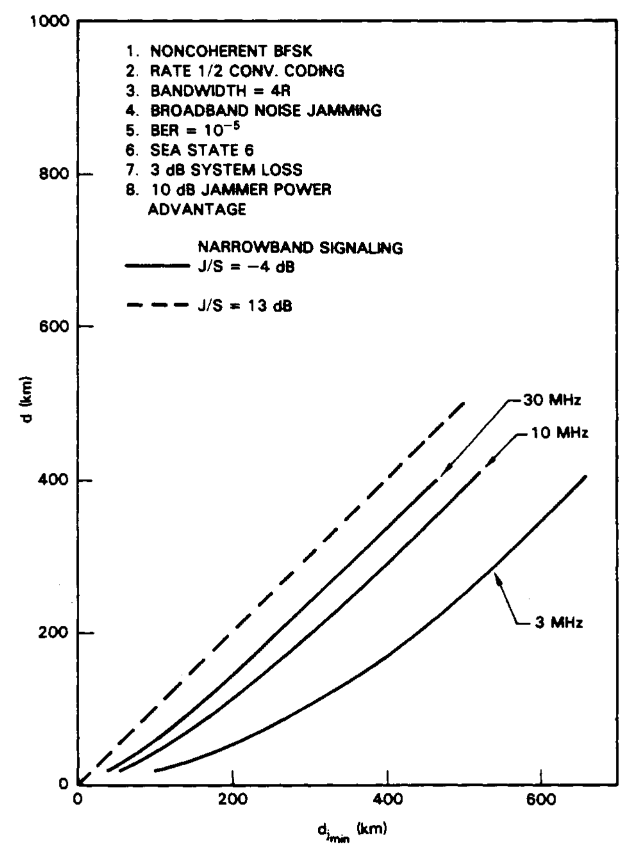


Fig. 7. $d_{j\min}(d)$ versus d for HF groundwave model and narrow-band signaling.

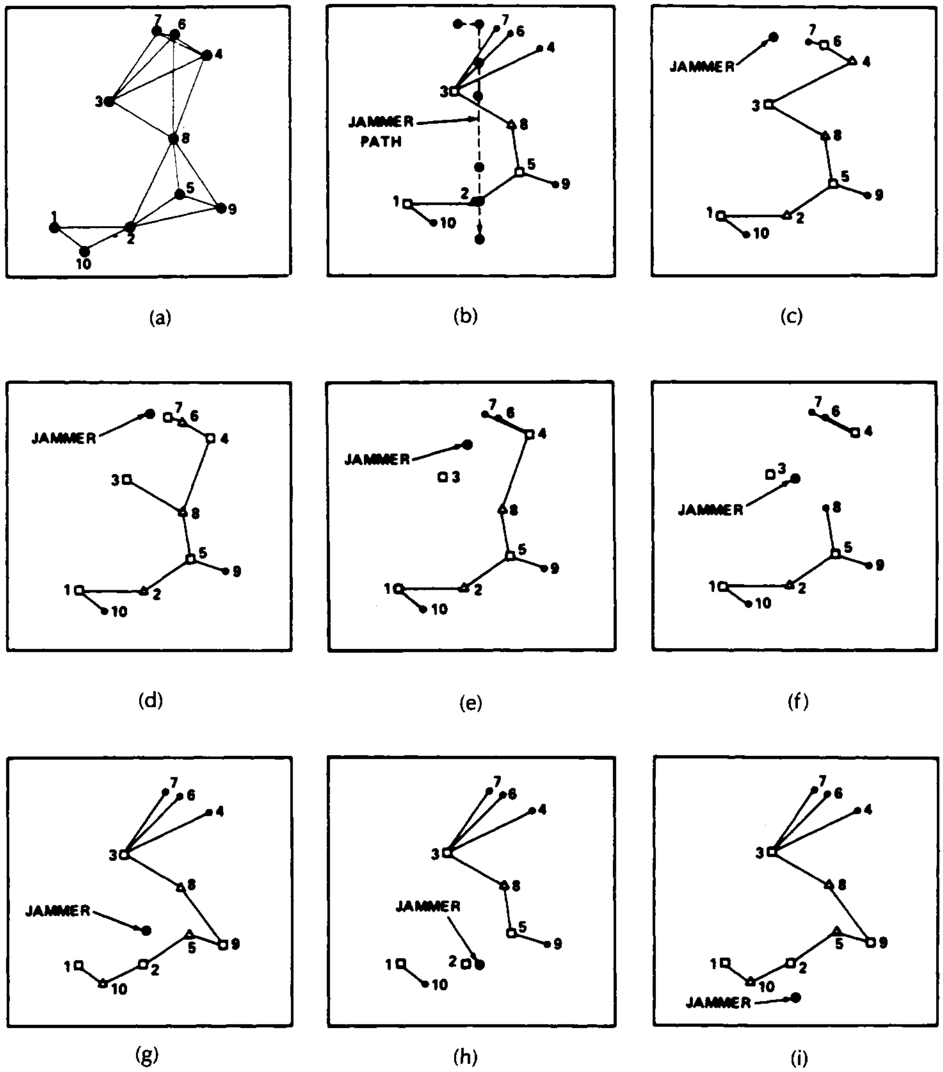


Fig. 8. Example of a network restructuring itself to combat a mobile jammer (10-MHz wide band).

8(b); nodes 1, 3, and 5 have become clusterheads and nodes 2 and 8 have become gateways (relays) to join the clusters, thus forming the backbone network. In that figure we show also the path the jammer is going to follow. The sequence of frames (c) through (i) in Fig. 8 illustrates how the network restructures itself in response to that jammer's changing location. It is assumed that the jammer radiates 10 kW of power as opposed to the network nodes, each of which is limited to 1 kW.

To understand the results shown in Fig. 8, we need only refer to Fig. 6. This figure shows that at 10 MHz, the jammer must be within about 70 percent of the transmitter-to-receiver distance before that link is broken. For example, when the jammer is at the location shown in Fig. 8(c), node 7 is prevented from hearing node 3, and node 6 is prevented from hearing either 3 or 8. Consequently, nodes 6 and 7 are no longer bidirectionally connected to clusterhead 3, as they were in Fig. 8(b). The LCA responds by designating an additional clusterhead, node 6; also, node 4 becomes a gateway to link this new head to the rest of the backbone network.

As the jammer approaches nodes 4, 6, and 7, the link from 3 to 4 is lost. Fig. 8(d) shows that this results in node 4 becoming a clusterhead. This in turn causes 7 to become a clusterhead and 6 to become a gateway. Note that the broken link (3, 4) has been by-passed while the network remains connected.

Approaching node 3 (see Fig. 8(e)), the jammer is successful in isolating this node; however, the rest of the network remains connected.

In Fig. 8(f), the jammer is near the critical node 8. This is a critical node in the sense that its loss will disconnect the network, as can be seen by examining the connectivities shown in Fig. 8(a). Fig. 8(f) shows that splitting does occur. At this point, only a repositioning of the nodes can achieve relaying that will reconnect the network. Of course, if additional link AJ capability can be obtained, for example, by reducing the information rate (by using a lower rate code) or switching to a frequency with more favorable propagation characteristics, then our network structuring algorithm will use the new links to form a connected network.

When the jammer moves away from nodes 3 and 8 to the position shown in Fig. 8(g), the network is able to return to the connected state. However, as the jammer approaches node 2, the ability to use this node for relaying is lost, and the network becomes disconnected again as illustrated in Fig. 8(h). Fig. 8(i) shows that the network returns to the connected state after the separation between the jammer and node 2 becomes large enough.

This example illustrates how the modest link AJ performance (achieved with spread-spectrum signaling and forward error control coding) in combination with the relaying and reconfiguration properties of the Linked Cluster Architecture permit a robust AJ behavior in the ITF Network.

We concluded in Section III that a scheduled-transmission approach to the channel access problem, implemented with CDMA signaling, is appropriate to support much of the communication requirements of the HF ITF Network. Thus, the layer-2 issue of link utilization in the network reduces to the question of efficient scheduling of link activation. The link activation problem addresses the questions of when and on which FH-codes transmitters broadcast and receivers listen.

V. Link Activation Scheduling

Given a network with connectivities established through use of the LCA (as for example in Fig. 8(a)), it is now necessary to determine a schedule for activating or enabling the links corresponding to these connectivities. It is desirable to achieve this scheduling by means of a *distributed* algorithm. Furthermore, it is desirable to come up with efficient schedules (i.e., schedules that do not have too many idle time slots during which no links are enabled) and without conflicts. Ideally, these schedules should also reflect fairness and should be somewhat matched to the traffic flow patterns.

A clarification is needed about the question of conflicts in FH spread-spectrum systems. (Here, it is assumed that each node has only one transmitter and one receiver.) In wide-band networks that use spread-spectrum signaling, it was observed earlier that secondary conflicts (defined in Section III) can be tolerated, *up to a point.* It is difficult to determine that point. As a first approximation, it is often assumed that this point is established as a threshold on the multiplicity of secondary conflicts; that is, if the number of interfering signals that are simultaneously received does not exceed some given number, no disruption of communication occurs, while if it does exceed that number, the level of interference is destructive. More accurately, one needs to look at the microstructure of the signals and evaluate link performance as a function of the interference level. For frequency-hopping signaling, which is relatively insensitive to the amplitude of other-user signals, it is reasonable to evaluate the bit- or packet-error probability as a function of the number of other users that are transmitting simultaneously [19].

Although it is often assumed that no node can transmit and receive simultaneously, we suppress the need to make an assumption on the matter by considering that each link gets activated in a bidirectional (full-duplex) sense, whether this is achieved by time-sharing or by genuine full-duplex operation.

The first algorithm for link activation was developed in [5] and was based explicitly on the use of the two TDMA transmission frames of the LCA. It was a distributed algorithm and was based on heuristics. Under this algorithm each node attempts to make an arbitrary assignment of slots to its neighbors and thus form a TDMA schedule. Obviously, each neighbor will attempt to do likewise, but as it will have different neighbors than the original node and as these nodes set up their own schedules independently, inconsistencies and conflicts will, in general, arise that must be resolved. The systematic way that the original slot allocation and the subsequent removal of conflicts are done constitutes the algorithm that is described in detail in [5].

Briefly, the Link Activation Algorithm (LAA) consists of two activities, namely, the allocation of slots and the resolution of scheduling conflicts. Slot allocation occurs during the first frame of the LAA and resolution of conflicts takes place during the second frame. Thus the link activation schedules are being formed at the same time that clusters are formed and linked. In this fashion, the LAA can ride on the LCA in that it shares the same transmission plan during its communication phase. When a node receives a schedule from its neighbor, the information is stored in its local database. During frame 1, slot allocations are made so that known scheduling conflicts are avoided. Thus a node allocates to a link the earliest slot and considers the various links in arbitrary order.

After its own transmission in frame 1, in which a node broadcasts the current version of its own schedule (which is in the making), it collects new information and updates its schedule so as to resolve any conflicts that arise. During frame 2, the node announces the latest version of its schedule, it collects some additional information, and may make some final changes to it by the end of the second frame, always with the objective of avoiding primary conflicts. However, occasional conflicts (or shared slots, as we call them) may still arise; in this case, the intended receiver must alternately monitor the transmissions of the nodes that share slots. The LAA's chief advantages are its simplicity, its distributed nature, the speed with which schedules are formed, and its compatibility with the LCA operation. Simulation studies demonstrated good slot utilization efficiency as well. Shared slots arose infrequently and idle slots occurred at an average rate of approximately 5 to 20 percent.

Soon after this algorithm was introduced, two things of interest happened. First, in [23] Hajek considered a generalization of the problem in which the traffic matrix was taken into account and slots were allocated to minimize the adverse effects of secondary conflicts. An analytical formulation was provided and it was shown that certain schedules existed (called uniformly most balanced) that were optimal simultaneously under a variety of criteria. It was also determined that finding such schedules is equivalent to the solution of network flow problems with convex costs. Algorithms for the solution were developed that could be decentralized. However, these algorithms were not truly distributed in that the coordination of the node's schedule updating was not ensured. Otherwise, the method could be incorporated into the LAA, generalizing it in the sense of i) secondary conflict control and ii) adaption to input traffic needs.

Second, it was pointed out by Hajek [23] and by Post, Sarachik, and Kershenbaum [24] that the problem of link activation scheduling has strong connections to graph-theoretic problems, such as graph coloring. A classical graph-coloring problem is to partition (or color) the nodes of a network (graph) so that any two nodes that are connected to each other have different colors. A link-coloring problem is to color the links of the network so that all links connected to the same node have different colors. An optimal coloring is one that uses the fewest colors.

Obviously, the link-coloring problem coincides with the link-activation problem in which "color" corresponds to "time-slot." The link-coloring problem in a graph G is identical to the node-coloring problem of its line graph, i.e., the dual graph whose nodes are in one-to-one correspondence with the links of G and are directly connected if and only if their corresponding links in G are incident on a common node. The problem is NP-complete and requires global connectivity information. Thus an efficient algorithm is needed, preferably distributed.

Silvester [25] has developed a scheduling algorithm based on first coloring the largest "clique" of traffic and then filling in the colors (slot schedules) for all remaining links. Post, Sarachik, and Kershenbaum [24] have developed a heuristic, called the "biased-greedy" algorithm, that solves the problem in a way that can take input traffic into account, does so efficiently, and with performance demonstrated via simulations to be close to derived lower bounds of the optimum. However, the algorithm is not distributed in the sense that global connectivity information is required to do the computation.

Based on the ideas of the "biased-greedy" algorithm, its authors proposed a distributed version, known as the Distributed Evolutionary Algorithm [26]. This algorithm gets around the difficulty of global information by: 1) starting from a specific part of the network (the clique that includes, say, node 1), 2) developing consistent schedules locally for that clique, 3) broadcasting connectivity and scheduling to one-hop-away neighbors of the clique, 4) incorporating those neighbors into the schedules, and 5) requiring newcomer nodes to broadcast connectivity and scheduling information to new neighbors (of two-hop distance to the clique) until the outer limits of the network are reached. In other words, global connectivity is acquired in a step-by-step (and distributed) fashion, while at the same time schedules are formed that permit subsets of nodes of the network to communicate. However, the algorithm can produce unsatisfactory schedules if links go down while the algorithm is running (e.g., some groups of nodes may not be able to communicate).

For the ITF Network we not only need schedules that make good use of the communication bandwidth available, but also it is important to form these schedules quickly, with little communication overhead, and in such a way that link and node failures during the execution of the scheduling algorithm do not leave any nodes unable to communicate. The current approach to link scheduling for the ITF Network lies in the so-called Transmission Scheduling Algorithm (TSA), which was developed in [27]. This algorithm makes explicit use of the spread-spectrum signaling realities in the network (including choices on code-assignment methods), is truly distributed, and addresses additional network requirements besides obtaining conflict-free transmission schedules. Schedules are produced by the TSA quickly and with little communication overhead. Evaluated with respect to several performance measures, it is shown to perform reasonably well—even in those situations where nodes and links are failing.

The Transmission Scheduling Algorithm also uses the TDMA transmission frames of the LCA. It does permit secondary conflicts, but tries to keep them under control, without, however, using any specific quantitative criterion for doing so. Moreover, it alleviates some of the shortcomings of strictly scheduled, fixed-TDMA protocols in its construction of schedules by offering the possibility of incorporating new nodes into the system, adapting to varying traffic needs, and making use of the cluster structure to improve on efficiency of channel utilization.

The basic idea is to maintain two sets of parallel schedules, one network-wide fixed TDMA schedule, such as the one used for the two frames of transmissions needed by the LCA, and a set of separate transmission schedules, one for each cluster. Thus if there are M clusters, there are at most $M + 1$ simultaneously transmitting nodes in any one slot. Also, each clusterhead schedules exactly one node at any time slot in its own schedule. (Of course, the possibility exists for a clusterhead to schedule *several* nodes during each time slot—if this is desirable). Since there are generally fewer clusters than there are nodes in the network and since clusters tend to be spread out, the number of secondary conflicts will usually be kept small.

The reasons for having an additional, fixed network-wide schedule are: a) to ensure that each node is aware of at least some of the communication slots of every other node in the net, b) to allow for the addition of new nodes into the network as will be explained later, and c) to enable the separate executions of the LCA in different frequency subbands at the corresponding reorganization epochs.

Since transmission schedules are not coordinated with each other, it is possible for a node to be scheduled to transmit on two schedules in the same time slot. If each node were to use an FH code associated with the schedule on which it is transmitting, then a form of *transmitter scheduling conflict* would arise, since a node can only transmit on one FH pattern at a time. Thus to avoid any transmitter scheduling conflict (there may still arise receiver scheduling conflicts), transmitter-based FH codes are used. Thus in most cases, a transmitter will use its own, unique FH code when transmitting. There are some situations, however (such as when a new node is joining the net), when a node may have to transmit on a code other than its own transmitter-based code.

The TSA is piggybacked onto the LCA in the sense that it shares the same transmission slots. In Table 2 we show the information that is exchanged during the execution of the LCA and the additional information that is passed by the TSA during frame 2 of the reorganization epoch.

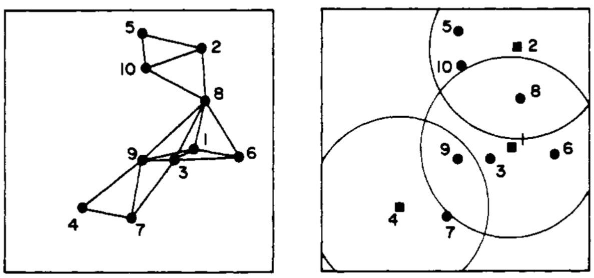
Table 2 Communications During Reorganization Epoch

	FRAME 1	FRAME 2
LINKED CLUSTER ALGORITHM (LCA)	1 … N: [1|0|1| • • • |0|0] NODES HEARD (THIS FRAME)	[1] 1 … N: [1|0|1| • • • |1] BIDIRECTIONAL CONNECTIVITIES; ID. OF OWN CLUSTERHEAD
TRANSMISSION SCHEDULING ALGORITHM (TSA)	NONE	1 … N: [3|0|4| • • • |4] SLOT ALLOCATIONS

As part of the TSA, each clusterhead announces during frame 2, the transmission schedule for the nodes of that cluster. We need not explicitly state the criteria to be used in constructing the schedule (fairness, traffic needs, etc.)

since the basis on which the slots are allocated is expected to differ from one network to the next. Since a node may not hear some of the schedules broadcast by clusterheads connected to its neighbors, each node repeats in its frame 2 transmission the schedule announced by its own clusterhead. This information enables nodes to schedule their receivers when they need to monitor a transmission from a neighboring cluster. The clusterhead selection rule makes sure that nodes that are not clusterheads hear from their own clusterhead prior to transmitting themselves in the second frame. As a result, every node becomes aware of at least two of the transmission schedules of all its neighbors. It knows, of course, of the network-wide fixed TDMA schedule; in addition, for those connected to its own clusterhead, it knows a second schedule directly from the clusterhead announcement, and of those in neighboring clusters it knows a second schedule from the schedule repetition provision mentioned above.

To illustrate the algorithm consider the example shown in Fig. 9. The network whose connectivities are shown in



(a) (b)

TRANSMISSION SCHEDULES

Slot #

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Fixed	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10	1	2	3
Set by 1	8	8	3	9	1	6	3	1	9	3	6	6	9	8	8	1	1	3	3	8	9	9	8
Set by 2	10	8	8	8	8	5	10	2	2	8	10	8	5	2	10	8	8	8	8	5	10	2	2
Set by 4	7	9	4	9	7	9	7	9	9	4	7	9	4	9	7	9	7	9	9	4	7	9	4

(c)

Fig. 9. Example of TSA scheduling. (a) Connectivities of example net. (b) Clusters formed by LCA. (c) TSA transmission schedules.

Fig. 9(a) produces the linked cluster structure shown in Fig. 9(b); the resultant coexisting transmission schedules are shown in Fig. 9(c) (see [27] for details). Two performance measures were computed for this example; namely, receiver utilization and the numbers of secondary conflicts. Receiver utilization here refers to the percentage of time that a receiver is within range of a node scheduled to transmit. For a fixed-TDMA protocol, such as the network-wide schedule, this number can be quite low (e.g., 36 percent for this example). On the other hand, simulations have shown that, for this example, the schedules produced by the TSA result in a much higher receiver utilization (84 percent).

The degree of secondary conflicts that can occur with unscheduled transmissions in this example is at most 5 (occurring at node 8 if all its neighbors transmit simultaneously). That number is reduced to 3 under the scheduled transmissions provided by the TSA. Moreover, 98 percent of the time the degree of secondary conflicts was at most 2 and 81 percent of the time there was at most a single transmission contributing to the secondary conflict. Of course, in highly connected networks the level of secondary conflicts could, potentially, be high. On the other hand, the level of secondary conflicts that result from schedules produced by the TSA remains low even for highly connected networks.

The addition of new nodes in a network operating under a TDMA protocol is difficult in a distributed environment, because *all* nodes at the *same* time must adjust the TDMA schedule to incorporate the new node. The TSA provides, however, the flexibility to permit a solution.

Consider the following modifications to the TSA. The network-wide, fixed-TDMA schedule, which is maintained in parallel with the cluster schedules, is extended to include one or more contention slots. These are contention slots in the sense that a node can freely transmit in them, but must use a separate, common code known to all potential participants in the network (i.e., existing and new nodes). A node that wishes to enter the network must first monitor the channel (using, for example, the fixed-TDMA schedule) for some time to determine which nodes it can hear. At the first opportunity, the new node may transmit a *network-entry* packet during a contention slot, using the special contention code. This packet contains (at a minimum) the new node's ID number plus a list of all the nodes that it can hear. Nodes that receive the network-entry packet, and that are also in that list, are obviously bidirectionally connected to the new node. They save that packet until the next reorganization epoch occurs. At that time, they broadcast that information in their scheduled slot during frame 1 of the LCA. Any clusterhead that receives this information about the new node includes it in its new schedule. Thus the new node becomes part of the network, even though it does not fully participate in the reorganization period and does not have an assigned slot in the network-wide schedule. Fig. 10 depicts the proposed modifications.

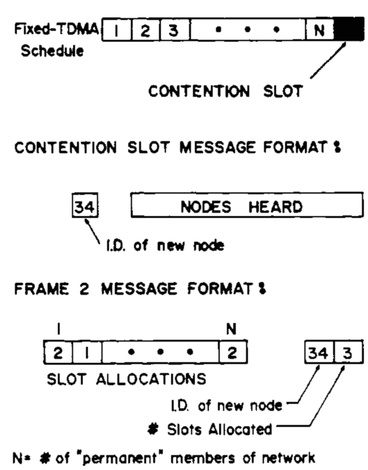


Fig. 10. Modifications of TSA to allow for new nodes.

To summarize, we have proposed a solution to the problem of channel access in survivable networks that use frequency hopping spread-spectrum signaling, such as the ITF Network. The solution is based on scheduled CDMA transmissions. The LCA yields a convenient framework of natural clustering in which to implement our solution. The solution is the fully distributed TSA that rides on the LCA and results in a multischedule system compatible with the Linked Cluster Architecture. Transmitter-based FH codes are used to avoid primary conflicts and transmitter scheduling conflicts. The distributed nature of a multicluster network reduces the number of secondary conflicts when transmissions are so scheduled. The network-wide schedule provides an inflexible, but reliable, component to the communication process. It provides the basis for reconfiguration, it allows the entry of new nodes, and can be relied upon as a backup, albeit inefficient, channel. The cluster schedules, on the other hand, provide efficient and adaptive means of communication and are designed to carry the bulk of the traffic.

Having addressed the important part of the network architecture that corresponds to the layer-2 issues of channel access, signaling, and link activation, we can turn our attention to higher layer issues such as routing, flow control, and some of the remaining aspects of the Link Layer. We do so in the next section.

VI. Other Issues

The task of network design for the ITF Network is far from being complete. In fact, the same can easily be said for almost any other packet radio network. The interrelationships between design factors at all levels are complex. There are tradeoffs between the seemingly most unrelated aspects. For example, the choices discussed in this paper concerning channel access profoundly affect the types of error-control schemes that can be used. The periodic enabling of a link that we have proposed may make ARQ methods inappropriate because of the potentially long delays involved. If this is the case (and it is not clear that it is indeed), error control may have to take place at the higher layers of the ISO architecture. The consequences of that are not easy to evaluate.

Even more mundane issues, such as the throughput analysis in a multihop network that uses CDMA with capture (a well-defined lower layer issue that is extensively discussed in this issue), are far from being accomplished or even well-understood.

Given this state of affairs and, at the same time, given the intense need to actually design, build, and operate radio networks, it is necessary to combine the network design "wisdom" that has accumulated over the last ten years with the art of innovative thinking and propose a design framework that can support a certain, perhaps very suboptimal, network operation with the hope that its shortcomings and imperfections will be identified and eventually overcome.

Taking such a viewpoint, we have proposed a structure for packet network design that has proven, at every step of the design process so far, to accommodate conveniently almost any design choice at various network levels and to satisfy, at least partially, almost any of the ITF requirements. Encouraged by the way this progress has been evolving, we have addressed the question of routing within the context of the same Linked Cluster Architecture.

The problem of routing is perhaps the most well understood (and studied) area in network design. In radio networks that do not involve node mobility, the problem of channel access and interference complicates matters slightly, but for the most part it can be separately solved. Thus routing once again reduces to finding a "best" sequence of links to be traversed by a message on its way to its destination. We are of course talking about multihop networks. The performance measure may be modified a little to reflect the congestion penalty that is paid by too many transmissions in a radio environment, but the solution philosophy is for the most part well cut out. We must find the "best" routes. If the network map and traffic requirements are given and if the performance measure is well-defined, we know what must be done (even though it may not be easy to do it because of computational and/or analytical difficulties, especially when there are variations in the traffic and occasional node or link failures). Actually, the use of spread-spectrum signaling eliminates many of the channel access complications because it makes it possible to think of a given node pair as a point-to-point wire connection rather than as part of a multiple access, broadcast environment.

However, when nodes move often and/or when links fail intermittently because of jamming, it is not clear that there is enough time to accumulate global connectivity information and use it in selecting routes. Especially if these topological changes must be traced by means of distributed algorithms (a subject that is still in the infancy stage of its development), it is clear that any of the known approaches will not work. In the extreme case of chaotic mobility (when topological changes are too frequent to attempt to trace their effects) it is clear what to do. It is, in fact, the only thing that seems possible—flooding. Despite the adverse effects of increased congestion, it is the only means of assuring delivery of information.

What we are interested in studying is the intermediate case in which the network does not change that frequently to make flooding the only feasible solution, and at the same time does not stay reasonably static long enough to make the "shortest route" method acceptable.

A highly adaptive "search-and-discover" approach to routing was preliminarily formulated in [28] for such a case. Under that approach, a node would flood a high-priority short query message through the network to discover a route that would lead to the destination node. Once the first link of that route was determined via a positive acknowledgment mechanism, it would be continuously used until a negative acknowledgment was received that would indicate either that this route is no longer available or that it is heavily congested. In that case, a new query would be generated to relocate the destination node.

Another preliminary approach discussed in [5] involves an idea similar to that of hierarchical routing as used in PRNET and SURAN. Except it makes use of the natural hierarchy provided by the Linked Cluster Architecture. This architecture facilitates routing by:

a) identifying "broadcast centers" (clusterheads) for use in the regional or net-wide broadcasts of messages,
b) defining a backbone network over which inter-cluster communication can be concentrated.

Since all nodes in a cluster are within one hop from a clusterhead, it is sufficient for every clusterhead to broadcast a message to achieve net-wide coverage. Consequently, the linked cluster structure offers a convenient way of sending control information to all nodes. As an example, suppose that we wish to identify the minimum-hop routes between all pairs of nodes immediately following a network reorganization epoch. Since each clusterhead knows the neighbors of its cluster members, it is only necessary for clusterheads to exchange this connectivity information (by using the backbone network, for example) and then to broadcast the resultant connectivity information to its cluster members. As a consequence, every node will obtain global connectivity information that can be used for shortest route computation. These routes may well bypass the backbone network and be much shorter and direct between certain node pairs. Actually, for the case of a two-hop path it may not be necessary to use the linked cluster structure at all to establish routes; since each node knows all its neighbors' neighbors, it can simply forward a packet to a neighbor of the intended destination.

Of course, if the connectivities are continually changing, we are back to the case considered just before. It is possible, however, to consider a two-tier approach, in the sense that normally the ITF Network is expected to maintain a stable configuration for long periods of time in which a shortest path routing mode can be implemented. Once in a while, however, the network may enter a volatile period in which topology is rapidly changing. During these periods, it must switch to a routing regime that is more like the adaptive, "search-and-discover" type that was discussed earlier, or even like flooding for extremely unstable connectivity situations.

Our work on routing is obviously still incomplete. We would like to point out, however, some differences between the ITF Network and networks like PRNET and SURAN that make the design approaches (not limited to the routing question) for these two types of networks necessarily different.

As stated in the Introduction, the three key factors that distinguish the ITF Network from other packet radio nets are:

1) the relatively small number of nodes,
2) the limited capacity (bandwidth) of the HF channel,
3) the use of FH-CDMA signaling.

These differences have necessitated the use of different clustering techniques, channel protocols, and routing strategies.

The cluster architecture proposed for SURAN nets involves clusterheads that may be several hops away from other cluster members [29]. Moreover, clusters are organized into a hierarchy of clusters (involving super-clusters, etc.). That architecture is clearly not suitable for a network with a small number of nodes and with links whose capacity cannot be wasted on multihop transmission of control information. Routing may be facilitated by the multilevel hierarchical structure, but only for networks with thousands of nodes connected via high-capacity links.

In addition, the implementation of the architecture in the ITF Network takes advantage of the FH-CDMA multiple-user channel property by using parallel scheduled transmissions; such schemes are not suitable for most networks with DS spread-spectrum signaling because of the near-far power differential problem. Furthermore, the TDMA schedules used in the execution of the LCA would be too long in large networks. Moreover, in high-capacity links congestion does not build up as rapidly, and therefore it is not necessary to try to avoid it by scheduling.

These fundamental differences among packet radio nets lead naturally to basic choices that point to different directions in the design process. We think that the truth of this statement is clearly demonstrated (and justified) by our analysis and, in a broader sense, by the collection of papers in this special issue.

VII. Conclusions

The work reported in this paper summarizes an on-going effort that started in the late 1970s. Over the years the direction of the project has been influenced by parallel efforts in packet radio network design as well as by developments in other fields, such as channel access, distributed algorithms, spread-spectrum communication, etc. In addition to the authors of this paper, several NRL scientists and research groups from the University of Illinois, UCLA, and the Polytechnic Institute of New York have contributed to this effort.

A simulation testbed is now being developed to perform a networking demonstration that will put the proposed design concept to its first real test. If the proposed architecture and algorithms are successfully tested, the remaining issues of higher level network design must be addressed and the holes must be filled at all levels by fully specifying the hardware and software implementation of the network before it can be subjected to shipboard field testing.

References

[1] J. E. Wieselthier, D. J. Baker, and A. Ephremides, "Survey of problems in the design of an HF Intra Task Force Communication Network," NRL Rep. 8501, Oct. 1981.

[2] J. E. Wieselthier, D. J. Baker, A. Ephremides, and D. N. McGregor, "Preliminary system concept for an HF Intra-Task Force Communication Network," NRL Rep. 8637, Aug. 1983.

[3] D. J. Baker, A. Ephremides, and J. E. Wieselthier, "An architecture for the HF Intra-Task Force (ITF) Communication Network," NRL Rep. 8638, Dec. 1982.

[4] D. J. Baker and A. Ephremides, "The architectural organization of a mobile radio network via a distributed algorithm," *IEEE Trans. Commun.*, vol. COM-29, pp. 1694–1701, Nov. 1981.

[5] D. J. Baker, A. Ephremides, and J. A. Flynn, "The design and simulation of a mobile radio network with distributed control," *IEEE J. Selected Areas Commun.*, vol. SAC-2, pp. 226–237, Jan. 1984.

[6] D. J. Baker, J. E. Wieselthier, A. Ephremides, and D. N. McGregor, "Resistance to HF jamming interference in mobile radio networks by an adaptive, distributed reconfiguration technique," NRL Rep. 8834, Aug. 1984.

[7] J. E. Wieselthier and D. N. McGregor, "Signaling issues in the design of a high-frequency (HF) Intratask Force Communication Network," NRL Rep. 8853, Jan. 1986.

[8] N. Ehrlich, "The advanced mobile phone service," *IEEE Commun. Mag.*, vol. 17, pp. 9–15, Mar. 1979.

[9] A. Nilson, W. Chou, and C. J. Graff, "Packet radio communication system architecture in a mixed traffic and dynamic environment," in *Proc. Comput. Networking Symp.*, pp. 51–66, 1980.

[10] M. Lawson and S. Smith, "Reconfiguration techniques of a mobile network," in *Proc. Int. Zurich Sem. Digital Commun.* (Zurich, Switzerland, 1980), pp. B10.1–B10.5.

[11] F. A. Tobagi and L. Kleinrock, "Packet switching in radio channels: Part II—The hidden terminal problem in carrier sense multiple access and the busy tone solution," *IEEE Trans. Commun.*, vol. COM-23, pp. 1417–1433, Dec. 1975.

[12] E. R. Berlekamp, "The technology of error-correcting codes," *Proc. IEEE*, vol. 68, pp. 564–592, May 1980.

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

[14] B. Hajek and T. van Loon, "Decentralized dynamic control of a multiaccess broadcast channel," *IEEE Trans. Automat. Contr.*, vol. AC-24, pp. 559–569, June 1982.

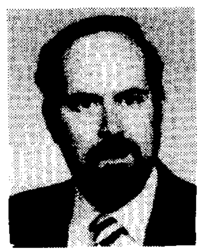
[15] J. I. Capetanakis, "Tree algorithms for packet broadcast chan-

nels," *IEEE Trans. Informat. Theory*, vol. IT-25, pp. 505–515, Sept. 1979.
[16] L. G. Roberts, "Dynamic allocation of satellite capacity through packet reservation," in *AFIPS Conf. Proc., 1973 Nat. Computer Conf.*, vol. 42, pp. 711–716, 1973.
[17] J. E. Wieselthier and A. Ephremides, "A new class of protocols for multiple access in satellite networks," *IEEE Trans. Automat. Contr.*, vol. AC-25, pp. 865–879, Oct. 1980.
[18] F. A. Tobagi, "Multiaccess protocols in packet communication systems," *IEEE Trans. Commun.*, vol. COM-28, pp. 468–488, Apr. 1980.
[19] M. B. Pursley, "Throughput of frequency-hopped spread-spectrum communications for packet radio networks," in *Proc. 17th Annu. Conf. on Information Sciences and Systems* (The Johns Hopkins Univ., Mar. 1983).
[20] J. E. Wieselthier and A. Ephremides, "A distributed reservation scheme for spread spectrum multiple access channels," in *Proc. Globecom'83*, pp. 659–665, Nov.-Dec. 1983.
[21] E. S. Sousa and J. A. Silvester, "A spreading code protocol for a distributed spread spectrum packet radio network," in *Proc. Globecom'84*, pp. 481–486, Nov. 1984.
[22] M. Spellman, "A comparison between frequency hopping and direct spread PN as antijam techniques," *IEEE Commun. Mag.*, vol. 21, pp. 37–51, Mar. 1983.
[23] B. Hajek, "Balanced scheduling in a packet synchronized spread spectrum network," in *Proc. 1983 IEEE INFOCOM*, pp. 56–65, Apr. 1983.
[24] M. J. Post, P. E. Sarachik, and A. S. Kershenbaum, "A 'biased-greedy' algorithm for scheduling multi-hop radio networks," in *Proc. 19th Annu. Conf. on Information Sciences and Systems* (The Johns Hopkins Univ.), pp. 564–572, Mar. 1985.
[25] J. A. Silvester, "Perfect scheduling in multi-hop broadcast networks," in *Proc. 6th Int. Conf. on Computer Communications* (London, England, Sept. 1982).
[26] M. J. Post, A. S. Kershenbaum, and P. E. Sarachik, "A distributed evolutionary algorithm for reorganizing network communications," in *Proc. MILCOM'85*, pp. 133–139, Oct. 1985.
[27] D. J. Baker, J. E. Wieselthier, and A. Ephremides, "A channel access protocol for survivable radio networks with frequency hopping spread spectrum signaling," in *Proc. 22nd Annu. Allerton Conf. on Communication, Control, and Computing*, pp. 50–59, Oct. 1984.
[28] M. L. Bates, "A simulated performance study of some distributed routing algorithms for mobile radio networks," M.S. thesis, Univ. of Maryland, College Park, 1986.
[29] N. Schacham and J. Westcott, "Future directions in packet radio architectures and protocols," this issue, pp. 83–99.

Anthony Ephremides (Fellow, IEEE) was born in Athens, Greece, in 1943. He received the B.S. degree from the National University of Athens in 1967 and the M.A. and Ph.D. degrees from Princeton University, Princeton, NJ, in 1969 and 1971, respectively, all in electrical engineering.

He was Visiting Professor at the National Technical University in Athens in 1978 and at the Electrical Engineering and Computer Science Department of the University of California at Berkeley in 1979. He has been on leave at MIT, Cambridge, MA, and at the ETH in Zurich, Switzerland, during 1985–1986. He is Professor of Electrical Engineering and a member of the Systems Research Center at the University of Maryland, College Park, where he has been since 1971. He has also been a Consultant at the Naval Research Laboratory since 1977. In addition, he is the President of Pontos, Inc., a private consulting firm, and a member of the Advisory Scientific Committee of the University of Crete in Greece. He was Director of the Fairchild Scholars and Doctoral Fellows Program, an academic and research partnership program in Satellite Communications between Fairchild Industries and the University of Maryland from 1980 to 1985. His research interests include communication theory and systems with emphasis on multi-access problems and on communication networks. He was the organizer of the 1983 IEEE workshop on Multi-User Information Theory and Systems and has taught several short courses on the subject under continuing engineering education programs in the United States and abroad.

Dr. Ephremides is a member of the Board of Governors of the IEEE Information Theory Group and President of the Group in 1987. During 1979–1980, he was an Associate Editor on Estimation, and has been an Associate Editor for Networks, Queueing Systems, and Stochastic Control for 1986 and 1987 of the IEEE Transactions on Automatic Control. He was the General Chairman of the 1986 IEEE Conference on Decision and Control in Athens.



Jeffrey E. Wieselthier (Member, IEEE) was born in Brooklyn, NY, on March 14, 1949. He received the S.B. degree from the Massachusetts Institute of Technology, Cambridge, in 1969, the M.S. degree from The Johns Hopkins University, Baltimore, MD, in 1971, and the Ph.D. degree from the University of Maryland, College Park, in 1979, all in electrical engineering.

He was employed at the Naval Surface Weapons Center, White Oak, Silver Spring, MD, from 1969 to 1979. Since 1979 he has been with the Naval Research Laboratory, Washington, DC. His current research interests include multiple access, routing, distributed control, and spread-spectrum issues in communication networks.

Dennis J. Baker was born in Detroit, MI, on July 25, 1940. He received the B.S. degree from the University of Detroit, Detroit, in 1963 and the Ph.D. degree from the University of Michigan, Ann Arbor, in 1973, both in physics.

He served as Scientific Consultant to Convair Division, General Dynamics, from 1968 to 1969. Since 1971 he has been with the Naval Research Laboratory, Washington, DC. His research interests include the simulation and prototyping of distributed systems, communication network design, and software engineering.