Experience with Speech Communication in Packet Networks

CLIFFORD J. WEINSTEIN, MEMBER, IEEE, AND JAMES W. FORGIE, MEMBER, IEEE

Abstract—The integration of digital voice with data in a common packet-switched network system offers a number of potential benefits, including reduced systems cost through sharing of switching and transmission resources, flexible internetworking among systems utilizing different transmission media, and enhanced services for users requiring access to both voice and data communications. Issues which it has been necessary to address in order to realize these benefits include reconstitution of speech from packets arriving at nonuniform intervals, maximization of packet speech multiplexing efficiency, and determination of the implementation requirements for terminals and switching in a large-scale packet voice/data system. A series of packet speech systems experiments to address these issues has been conducted under the sponsorship of the Defense Advanced Research Projects Agency (DARPA).

In the initial experiments on the ARPANET, the basic feasibility of speech communication on a store-and-forward packet network was demonstrated. Techniques were developed for reconstitution of speech from packets, and protocols were developed for call setup and for speech transport. Later speech experiments utilizing the Atlantic packet satellite network (SATNET) led to the development of techniques for efficient voice conferencing in a broadcast environment, and for internetting speech between a store-and-forward net (ARPANET) and a broadcast net (SATNET). Large-scale packet speech multiplexing experiments could not be carried out on ARPANET or SATNET where the network link capacities severely restrict the number of speech users that can be accommodated. However, experiments are currently being carried out using a wide-band satellite-based packet system designed to accommodate a sufficient number of simultaneous users to support realistic experiments in efficient statistical multiplexing. Key developments to date associated with the wide-band experiments have been 1) techniques for internetting via voice/data gateways from a variety of local access networks (packet cable, packet radio, and circuit-switched) to a long-haul broadcast satellite network and 2) compact implementations of packet voice terminals with full protocol and voice capabilities.

Basic concepts and issues associated with packet speech systems are described. Requirements and techniques for speech processing, voice protocols, packetization and reconstitution, conferencing, and multiplexing are discussed in the context of a generic packet speech system configuration. Specific experimental configurations and key packet speech results on the ARPANET, SATNET, and wide-band system are reviewed.

I. Introduction

PACKET techniques provide powerful mechanisms for the sharing of communication resources among users with time-varying demands, and have come into wide use for provision of data communications services to the military and commercial communities. The primary application of packet techniques has been for digital data communications where the bursty nature of user traffic can be exploited to achieve large efficiency advantages in utilization of communication resources. Packet networks [1]–[8] using a variety of point-to-point and broadcast transmission media have been developed for these applications, and techniques have been developed for internetwork communication [10], [11] among dissimilar nets.

Packet techniques offer significant benefits for voice as well as for data [15]–[33]. The integration of digital voice with data in a common packet-switched system offers potential cost savings through sharing of switching and transmission resources [30], as well as enhanced services for users who require access to both voice and data communications [59]–[61]. Packet internetworking techniques can be applied to provide intercommunication among voice users on different types of networks. Significant channel capacity savings for packet voice can be achieved by transmitting packets only when speakers are actually talking (i.e., during talkspurts). The silence intervals can be utilized for other voice traffic or for data traffic. Packet networks offer significant advantages for digital voice conferencing in terms of channel utilization (only one of the conferees needs to use channel capacity at any given time) and in terms of control flexibility. A packet network allows convenient accommodation of voice terminals with different bit rates and data formats. Each voice encoder will use only the channel capacity necessary to transmit its information rather than the fixed minimum bandwidth increment typically used in circuit-switched networks. The digitization of voice in packet systems provides the opportunity for security techniques to be applied as necessary to the speech traffic. Secure packet data communication techniques [13] can be applied as well for data users who require this service. Packet networks also provide a system environment for effective exploitation of variable-bit-rate voice transmission techniques, either to reduce average end-to-end bit rate or to dynamically adapt voice bit rate to network conditions.

It has been necessary to address a number of issues in order to develop the techniques required to realize these benefits. The development of packet protocols for call setup and speech transport, and strategies for reconstitution of speech from packets arriving at nonuniform intervals have been required. Other issues include the development of efficient packet speech multiplexing techniques,

Manuscript received April 11, 1983; revised August 5, 1983. This work was supported by the Defense Advanced Research Projects Agency.

The authors are with M.I.T. Lincoln Laboratory, Lexington, MA 02173.

0733-8716/83/1200-0963$01.00 ©1983 IEEE

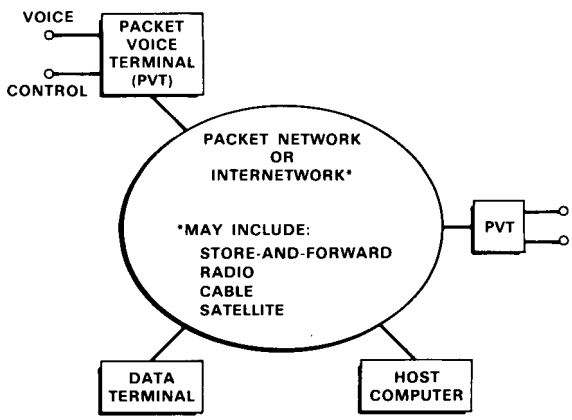

Fig. 1. Generic packet speech system configuration.

and the minimization of packet overhead and effective traffic control strategies to allow network links to be heavily loaded without saturation. System developments have been undertaken to help assess the implementation requirements for terminals and switching in a large-scale packet voice/data system, and efforts continue to drive down the size and cost of system components.

A series of packet speech experiments and system developments to address these issues has been conducted under the sponsorship of the Defense Advanced Research Projects Agency (DARPA). These efforts were initiated in 1973 by Dr. R. E. Kahn of the DARPA Information Processing Techniques Office (IPTO), who has provided leadership and numerous technical contributions through the course of the work. As will be noted in this paper and in the references, numerous individuals in several organizations have made significant contributions to the system development and experiments. The purpose of this paper is to review and evaluate the experience gained so far from these efforts in packet speech systems experiments. The perspectives and conclusions are the responsibilities of the authors and are necessarily influenced by the specific involvement of ourselves and our colleagues at Lincoln Laboratory.

This paper will begin by describing basic concepts and issues associated with packet speech systems. A generic packet speech system configuration will be described, and requirements and techniques for digital speech processing, protocol functions, packetization and reconstitution, conferencing, and multiplexing will be discussed. With this as a point of reference, the experimental system configurations and key results for packet speech on the ARPANET, SATNET, and wide-band system will be described.

II. Packet Speech Concepts and Issues

The purpose of this section is to set a general framework for the descriptions of specific experimental packet speech systems to follow in subsequent sections.

A. Generic Packet Speech System Configuration

A generic packet speech system configuration is depicted in Fig. 1. The interface between the user and the network is provided by a functional unit referred to as a packet voice terminal (PVT) [22]. The PVT may, but need not, be implemented in a single physical unit dedicated to a single voice user. Functionally, the user interfaces with the PVT much as with an ordinary telephone set, and the PVT interfaces with the packet network. In addition to being able to talk and listen, the user is provided with a full range of control and signaling capabilities including dialing and ringing. Both control signals and voice are transmitted from PVT to PVT over the network in digitized packet form. The resources of the integrated voice/data packet network are shared statistically with data traffic among host computers and data terminals as well as with other voice users. The packet network may be of the original store-and-forward type as exemplified by the ARPANET; may utilize packet radio, cable, or satellite techniques; or may be composed of an internetwork combination of these various types of packet nets, connected by gateways.

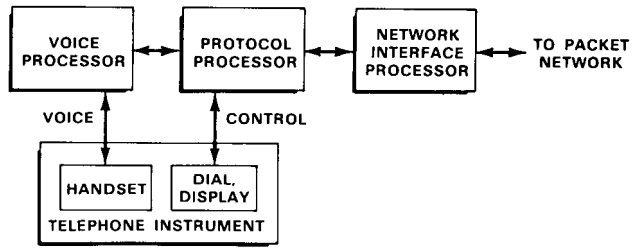

Fig. 2. Functional block diagram of packet voice terminal.

B. Generic Packet Voice Terminal Configuration

A functional block diagram of a packet voice terminal is shown in Fig. 2 which shows three major functional modules each associated with a processor. It is not necessary to use separate processors to achieve the functional modularity, but we have done so in the microprocessor PVT implementation [22] discussed later and find it convenient to use the same terminology here. The voice processor converts between analog and digital speech at digitization rates typically varying from 2 kbits/s to 64 kbits/s, and marks each *parcel* (typically 20–50 ms of speech) as containing either active speech or silence.

The protocol processor is the primary controlling module of the PVT. The protocol processor includes an interface with the user dial/display and must generate and interpret the packets necessary for establishing the call. The protocol processor provides the basic interface between the synchronous voice coding/decoding process, and the asynchronous packet network. The buffering and reconstitution algorithms to produce steady speech to the listener are implemented in the protocol processor.

The network interface processor provides the network-dependent packet transport mechanism. Ideally, all network-dependent hardware and software would be contained in this module. In practice, we have found it difficult to maintain this pure modularity because of a need to incorporate network-dependent elements into the packetization and reconstitution processes in the protocol processor.

The telephone instrument provides the simplest user interface to the PVT. The flexibility of the packet system

allows the possibility of a wide range of user functions and displays, which can exceed the signaling capability of the telephone instrument. In some experiments, computer terminals have been used to augment the user interface.

An important development in the work we will describe on packet voice is the evolution of the PVT from implementation on large general-purpose computers to compact microprocessor-based systems. In our view, this development is essential in making packet voice practical and affordable. We have generally focused on a distributed approach where each separate PVT performs complete voice processing and protocol functions for one user. A more centralized approach is also possible, where a single facility would simultaneously perform the functions of a number of PVT's for multiple users.

C. Digital Speech Processing Functions

The primary voice processing function for packet speech is speech digitization. Two other important voice processing functions are also noted here—speech activity detection and echo control.

1) Speech Encoding Algorithms: Speech is a compressible source [34] that can be coded at rates ranging from 64 kbits/s to below 2.4 kbits/s. Recent packet experiments have made use of the pulse code modulation (PCM) widely used in digital telephony, but all the earlier work described in this paper used encoding techniques [36] such as CVSD (continuously variable slope delta modulation) or LPC (linear predictive coding [37]) to provide data rates low enough for use on the networks that were available for experimentation.

Packet systems offer flexibility for taking advantage of speech encoders at a variety of rates. The PVT may include a variety of (fixed) speech bits rates, which could be selectable at dialup according to network load. More complex coding schemes [42] can be applied which vary transmission rates according to the time-varying compressibility of the speech signal. Or multirate "embedded coding" algorithms [38], [39] can be used to allow rapid adaptation [33] of voice bit rates to network conditions which may vary during a call. Selection of a speech coding algorithm [35], [36] for a given application depends on many factors including network bit rate constraints, speech quality needs, noise or distortions on the input speech, and terminal cost and complexity constraints.

2) Speech Activity Detection: A key advantage of packet speech is the ability to save bandwidth by transmitting packets only during talkspurts. Therefore, accurate discrimination between speech and silence, or speech activity detection (SAD), is an essential voice processing function [43]–[45]. The SAD algorithm must minimize the average percentage activity, but also meet tight constraints on the fraction of lost speech. SAD, in a laboratory or quiet input speech environment, is relatively straightforward. But when the speaker is in a noisy environment, or when the speech originated in the switched telephone network (STN), the design of effective SAD algorithms is more difficult.

In our system model, SAD is performed in the voice processor, which marks parcels delivered to the protocol processor as silence or speech. The protocol processor would normally packetize and transmit only the speech parcels except that it may transmit additional parcels at the beginning and end of a talkspurt to improve speech quality. Such a "hangover" at the end of a talkspurt is commonly used to include weak final consonants in a talkspurt and to bridge across short gaps.. An "anticipatory" parcel at the start of a talkspurt can give a smoother startup and is easy to provide in a packet system since the required buffer space is already present for use in the packetization process.

3) Echo Control: Echo control is not needed in a pure packet speech system in spite of the delays that may be present since the system is fully digital and provides isolation between the two directions of voice transmission for the entire path between sending and receiving handsets. However, echo control becomes an issue if we wish to interconnect a packet network and the common STN. Techniques for controlling echos [46], [47] include 1) echo suppression, generally aimed at passing speech in only one direction at a time; and 2) echo cancellation, which attempts to adaptively cancel echos and maintain full duplex speech. Echo cancellation is generally the preferred, but more costly, technique. Echo canceller chips which reduce the cost are becoming available. If the generic PVT were to be used to interface with the STN, it could be equipped with an echo canceller as part of its voice processor, to cope with echoes caused by the two-wire local loop in the STN. Both echo suppression [57] and cancellation [54] have been used in STN interface experiments on the wide-band network.

D. Packet Speech Protocol Functions

The development of the ARPANET as a packet communication resource was quickly, and by necessity, followed by the development of a set of protocols (i.e., rules for conducting interactions between two or more parties) to organize and facilitate use of this resource for a variety of applications. A network control protocol (NCP) was developed to allow controlled packet communication among processes running in dissimilar host computers [9]. Higher level protocols were developed to serve specific user needs. These included TELNET for terminal access to remote computers and file transfer protocol (FTP) for transmission of large files. Both TELNET and FTP obtained access to the network through NCP. This technique of *protocol layering* to partition and organize the task of providing various levels of communication services has been a fundamental aspect of the development of packet communication systems [12].

The original ARPANET protocols were designed to provide very reliable end-to-end packet delivery either at high throughput (e.g., FTP) or low delay (e.g., TELNET). Both NCP and the basic node-to-node protocols imposed end-to-end flow restrictions which included retransmissions when necessary to reliably deliver all the packets and worked against the simultaneous achievement of high throughput and low delay. But for real-time voice communication, both high throughput and low delay are

needed. Some reliability may be sacrificed, as a small percentage of lost packets is tolerable. Therefore, new protocol developments were needed for packet voice.

The initial work on packet voice protocols focused around the development of a high-level protocol known as the network voice protocol (NVP). Dr. D. Cohen of the Information Sciences Institute (ISI) was the chief architect of NVP [16], [17]. Functions of NVP include

1) call initiation and termination, including negotiation of voice encoder compatibility and handling of ringing and busy conditions;

2) packetization of voice for transmission, with the time stamps and sequence numbers needed for speech reconstitution at the receiver;

3) speech playout with buffering to smooth variable packet delays.

NVP is designed to pass its packets to a lower level protocol for transport across the network to meet real-time speech requirements. In order to avoid NCP's flow restrictions, NVP bypassed NCP for packet transport. In addition, modifications were made to the basic ARPANET transport protocols to provide an "uncontrolled" packet service which reduced packet flow restrictions between IMP's (see Section IV-B). The original NVP used the basic ARPANET (host–IMP and IMP–IMP) protocols directly to deliver its packets, and was independent of and generally incompatible with other protocols (e.g., NCP) in use at the time.

Since the original NVP made use of the ARPANET directly, extension to other networks (e.g., the Atlantic SATNET) required creation of a new protocol for each new network. This motivated the development of a second generation of voice protocols with a more general internetwork-oriented approach and with network-dependent aspects limited to the lowest level. Protocol functions were separated into two levels. The "higher" functions of call setup, packetization, and reconstitution, as well as dynamic conference control features, were incorporated into a second-generation version of NVP. The lower level protocol, which has come to be named "ST," provides an efficient internet transport mechanism for both point-to-point conversations and conferences. The name ST is derived from the work "stream" which refers to the type of traffic load that voice customers offer to a packet network. ST operates at the same level in the protocol hierarchy as IP, the DoD standard internet protocol [11] for datagram traffic. ST is designed to be compatible with IP. NVP may call on IP for delivery of control packets, and on ST for delivery of voice packets.

ST differs from IP in being a virtual circuit rather than a datagram protocol. Transmission of ST packets must be preceded by a connection setup process arranged by an exchange of control messages. During the connection setup, an internet route is established, and gateways along the path build tables pertaining to the connection. The preplanning involved in the connection setup and the existence of these connection-oriented tables allows ST to offer special services and efficiencies.

Fig. 3 illustrates how the current internet packet voice protocols relate to each other and to corresponding data

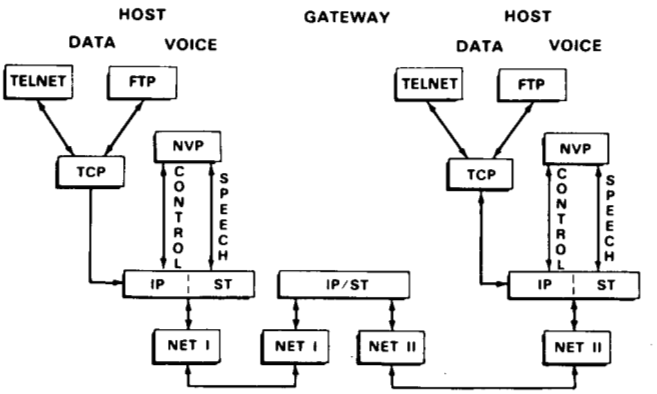

Fig. 3. Protocol hierarchy for internet packet voice and data communication.

handling protocols. Net I and net II designate individual packet networks, and might represent ARPANET, SATNET, or local area cable or radio nets. The situation depicted shows the protocol layers to be traversed in order for voice and data users on net I to communicate (through a gateway) with similar users on net II. The internet data file transfer protocol and the terminal-oriented protocol TELNET utilize a DoD standard transmission control protocol (TCP) for reliable packet delivery. TCP calls, in turn, on IP for packet transport. This is a departure from the original situation in the ARPANET, where FTP utilized NCP, which interfaced directly to the network. Similarly, NVP utilizes both IP and ST for packet transport; IP is used primarily in call setup situations, and ST is used for speech transport.

E. Speech Packetization and Reconstitution

Packet communication necessarily involves both fixed components of delay due to transmission and propagation, and statistically varying components such as queueing delays in network nodes or in gateways. Additional varying delay components are caused by packet retransmissions to compensate for errors in delivery and by the possibility that all packets between a particular source and destination may not follow the same route. In addition to delay effects, some packets may be lost between source and destination. In this regard, a delay versus reliability tradeoff is possible where (for example) delays due to retransmissions can be reduced at a cost of an increase in percentage of lost packets.

The purpose of speech packetization and reconstitution algorithms [31] is to provide speech with 1) minimum overall end-to-end delay and 2) any anomalies caused by lost or late packets basically imperceptible to the listener. Ideally, the overall packet network would provide high enough link bandwidths and sufficient nodal processing power to keep delay and delay dispersion within tightly controlled limits. In such a case, very simple packetization and reconstitution algorithms in the PVT may suffice. However, in some situations where packet speech is required, it may not be possible to control network design. In particular, when there is a need to transmit speech over an existing packet data network, it may be necessary to use more elaborate algorithms.

1) Choice of Packet Size: Resolving the issue of packet size forces us to make some difficult compromises. In order to minimize both the packetization delay at the transmitter and the perceptual effect of lost packet anomalies at the receiver packets should be as short as possible. Experience with lost packet anomalies indicates that individual packets should ideally contain no more than about 50 ms of speech [31]; ideally, we would like packets to be even shorter to minimize packetization delay. On the other hand, in order to maintain high channel utilization, we would like to keep the number of speech bits per packet as high as possible relative to the overhead which must accompany each packet. This tradeoff is particularly difficult for narrow-band speech. For example, 50 ms of 2400 bits/s speech is represented by only 120 bits, which is less than the header size of many existing packet networks. For higher speech bit rates, relative packet overhead is less of a problem. An obvious conclusion is that future packet voice networks should be designed with minimum required header lengths.

The choice of packet size is also influenced by limitations on network throughput in packets/s. For the same user data rate, processing loads on network nodes will generally increase as packet size is decreased. This can force use of longer packets. For example, our typical range of packet sizes for real-time speech transmission across the ARPANET was 100–200 ms, corresponding to 5–10 packets/s because the network could not consistently sustain a higher rate. In some cases it may be desirable to adapt packet size to time-varying network conditions. In speech experiments conducted by SRI on packet radio nets (PRNET's) [20], [21] the radio provides channel availability information to the voice terminal which buffers speech and sends variable size packets depending on the intervals between opportunities for access to the networks.

2) Time Stamps and Sequence Numbers: To assist in the reconstitution process, it is desirable to include a time stamp and a sequence number with each transmitted packet. The time stamp allows the receiver to reconstitute speech with accurate silence gap durations in spite of varying delays between talkspurts. Incorrect gap durations can cause significant perceptual degradation in the output speech, especially for short gaps between syllables, or between words in a phrase. The time stamp also allows reordering of out-of-order packets at the receiver. The time stamp is derived by counting every speech or silence parcel generated by the voice processor. A few bits (we use 12) will suffice to cover a range of relative timing about twice the packet transit time dispersion range of the network.

The sequence number allows the receiver to detect lost packets whereas with a time stamp alone it would not be possible to distinguish silence gaps from packet loss. The detection of lost packets can be used by the receiving PVT to inform the listener (by playing out a distinct audible signal) that some speech has been lost. This can be particularly important if packets contain enough speech to include linguistically significant utterances (such as the word "not"). Detection of lost packets can also be used to allow the terminals to adapt bit rate and/or packet rate to network conditions.

If the network provides service with very short delays and very little delay dispersion, then satisfactory speech can be produced without either time stamps or sequence numbers. However, our experience, both with packet speech experiments and simulations, indicates that both time stamps and sequence numbers should be included.

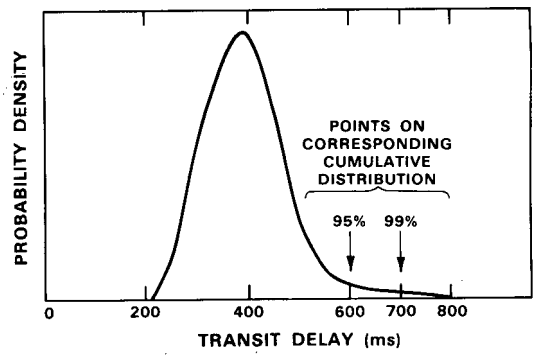

Fig. 4. Illustrative probability density function of transit delays in a packet network.

3) Reconstitution of Speech from Received Packets: The reconstitution algorithm has two major tasks, 1) it must buffer incoming packets and decide exactly when to play them out, and 2) it must decide what to play out when it has finished playing out a packet and the next packet is not available.

Fig. 4 shows an illustrative probability density function for transit delay in a packet network. The delay ranges shown are typical of some of our measurements on 10 hop paths through the ARPANET, but the points to be made are more general. In the case illustrated, 99 percent of the packets experience delays between 200 and 700 ms. Hence, a reconstitution delay (inserted at the receiver) of 500 ms would be sufficient to cover this spread. A 400 ms reconstitution delay would assure playout of 95 percent of the packets. Since some packets may be lost in the net, there is no value of reconstitution delay that can guarantee playout of all packets. Even if all packets did arrive, it would be undesirable to unduly increase delay to account for a few very late arrivals. The network's delay characteristics are generally not known in detail *a priori* and may vary with time. The degree of complexity to be built in to the reconstitution algorithm should be chosen based on the knowledge we do have of the network delays. A fixed reconstitution delay would suffice if network delays and delay dispersion are short. If delays are expected to be large or dispersions vary greatly with the network load, it would be desirable to use an adaptive algorithm (see [31] for an example of such an algorithm) to adjust the reconstitution delay to effect a compromise between packet loss and overall delay.

The other major reconstitution algorithm task is to decide what to play out when it has finished playing out a packet and the next packet is not available. This can result from a late or lost packet or it may simply indicate a pause in the talker's speech. Typically, the reconstitution algorithm has no way to distinguish these cases and should take the same action in either case. A number of fill-in strategies have been tried, including 1) filling with silence, 2) filling by repeating the last segment of speech data, and 3) filling with repeated frames of speech data which are made voice-

less and have energy values which decay with time. The third strategy has generally been found to be the most effective, particularly for framed vocoders such as LPC. However, the best choice of fill-in strategy varies with encoder type, packetization size, and statistics of gaps introduced by the network.

F. Conferencing Techniques

Digital voice conferencing imposes a number of requirements in addition to those required for point-to-point speech. There is a need to set up and control multiple connections and to deliver each talker's speech to multiple destinations. If narrow-band speech vocoding is used, a talker selection technique is generally required. Such vocoders cannot successfully handle more than one voice and the alternative of providing several vocoder synthesizers at each site is both cumbersome and expensive.

Packet techniques offer advantages for digital voice conferencing in a number of areas [28]. Since packets need be sent only when speech is present, they can make very efficient use of network resources in conferences where typically only one participant is speaking at any given time. Because connections to packet networks are multiplexed, it is simple for speech terminals and conference controllers to exchange control information at the same time that speech is being transmitted. This out-of-band signaling capability helps in achieving effective conference control, including the control algorithm which selects a talker to "have the floor" at a given time. The use of packets simplifies the implementation of distributed conference control, an important feature for military applications where its use can enhance survivability.

In order to explore the features and problems of packet voice conferencing in some detail, experimental implementations described in sections to follow have been carried out on ARPANET, SATNET, and the WB SATNET.

G. Statistical Multiplexing of Packet Voice and Data

An important goal for packet voice systems is to achieve efficient statistical multiplexing of multiple voice users, and of voice users with data traffic, on common transmission resources. Much analysis and simulation work has been reported showing potentials and limitations of voice/data multiplexing for various system configurations. One of the goals of packet speech systems experiments is to validate these results or identify practical limitations not shown in the analyses.

Some selected observations related to statistical multiplexing in packet voice systems are noted below. These observations and related analyses or simulations are described in [48]. Similar results have been obtained by a number of other researchers [51].

First, packet speech multiplexing allows a straightforward utilization of the tradeoff between delay and channel utilization (or equivalently between delay and "TASI advantage") [49], [50]. The number of users multiplexed onto a link can be increased at a cost in variable buffering delay at the multiplexer. The relative efficiency improvement offered by buffering is greatest where a small number of users are multiplexed, indicating potential for efficiency in a distributed net where local concentrations may be smaller than required for efficient circuit-switched TASI.

A second observation, based on simulations (as cited in [48]), is that interactive data traffic (characterized by Poisson packet arrival processes) can make efficient use of silence intervals in voice calls. However, the utilization by data traffic of varying capacity due to voice call initiation and termination is not nearly as effective due to the much slower variation in channel capacity used by voice.

A third observation is that local area carrier-sense multiple-access (CSMA) cable networks can be used effectively for voice [23]. The bandwidth utilization of such a CSMA network can be equal to or better than the efficiency obtained using fixed time division multiple access (TDMA). CSMA cable networks have been effectively employed for packet voice and are an important part of the experimental wide-band system.

Finally, variable-rate voice flow control techniques [33] using embedded coding can be employed effectively in situations where we are attempting to maintain link loads close to capacity, and temporary overloads are inevitable. Embedded coding allows immediate response by network nodes to such overloads (by discarding packets), with minimal impact on speech users, since communication can be maintained with a temporary degradation in speech fidelity.

III. Summary of Packet Speech Experiments

A summary of key characteristics of the packet speech experiments conducted under DARPA sponsorship is shown in Table I. More detail on each set of experiments will be presented later; for definition of the abbreviations and acronyms used in Table I, see the Appendix. The first network to be used was the ARPANET which consists of intelligent store-and-forward nodes called interface message processors (IMP's) connected primarily by 50 kbits/s point-to-point leased lines. Later, broadcast nets using satellite, radio, and cable were utilized. Initial internetting experiments were conducted using ARPANET and SATNET. The wide-band system is specifically configured as an internetwork where voice users reside on local nets and access the WB SATNET through gateways. Interoperation with circuit-switched telephone systems has also been introduced in the wide-band system. Such interoperation would be essential in introducing packet speech into an environment dominated by circuit-switched voice users.

The link (point-to-point) or channel (broadcast) bit rates quantitatively indicate the limited capacity available for voice in the earlier experiments as well as the greater capacity of the wide-band system. Because of limited network bit rates, most of the experiments on ARPANET and SATNET used LPC vocoding. A few CVSD experiments (primarily at 9.6 kbits/s) were conducted on ARPANET. Voice bit rates used in the wide-band system have ranged

TABLE I
SUMMARY OF PACKET SPEECH EXPERIMENTS

Networks	Network Types	Link or Channel Bit Rates (Kbps)	Voice Algorithms and Bit Rates (Kbps)	Time Period	Sites	Voice Processors	Protocol Processors
ARPANET	Point-to-Point (PTP) Store and Forward (SF)	50	LPC,LPC (VFR): 2-5 CVSD: 9.6-16	1974-79	CHI, ISI, LL, SRI	AP120 AP120B FDP,LDVT SPS-41	MP32 PDP-11/45 TX-2,PDP-11/45 PDP-11/40
SATNET	Broadcast (B'cast) Satellite	64	LPC: 2.4	1977-79	BBN, NDRE, UCL	LPCM	PDP-11/40
ARPANET + SATNET	PTP/SF + B'cast Sat	50 + 64	LPC: 2.4 LPC: 2.4	1978-79	ISI,LL + NDRE,UCL	AP120B,LDVT LPCM	PDP-11/45 PDP-11/40
PRNET	B'cast Radio	100-400	LPC: 2.4, CVSD: 16	1978-83	SRI	LPCAP, CHI-V	LSI-11 PDP-11/23
WB SYSTEM							
WB SATNET + LEXNET + PRNET + TELEPHONE NETS	B'cast Sat + B'cast Cable + B'cast Radio + Circuit-Switched	772-3088 1000 100-400 ---	LPC: 2.4, CVSD: 16, ECVSD: 16-64, PCM: 64	1980-83	LL, ISI, SRI, DCEC	CLPC AP120B CHI-V	8085 PDP-11/45 PDP-11/23

TABLE II
PACKET CONFERENCING EXPERIMENTS

NETWORKS	CONTROL TECHNIQUES		PACKET ADDRESSING	DEMONSTRATED	SITES
ARPANET	CENT	PB	PTP	1976	CHI, ISI SRI, LL
SATNET	DIST	PB	B'CAST	1978	NDRE, UCL, BBN
ARPANET + SATNET	CENT + DIST	PB	PTP + B'CAST	1979	ISI, LL + NDRE, UCL
SATNET	DIST	VOICE	B'CAST	1979	NDRE, UCL, BBN
WB SYSTEM	DIST	VOICE	B'CAST	1982	ISI, SRI, LL, DCEC

CENT = CENTRALIZED
DIST = DISTRIBUTED
PB = PUSH BUTTON
PTP = POINT-TO-POINT
B'CAST = BROADCAST

from 2.4 to 64 kbits/s. Accommodation of 64 kbits/s PCM is important in allowing convenient interoperation with digital circuit-switched systems which use PCM as a standard.

As indicated, a large variety of narrow-band voice processors and protocol processors have been used in the packet speech experiments. Voice processors range from special laboratory-built programmable signal processors (e.g., FDP, AP120, LDVT), to very compact LPC units (CLPC). Protocol processors include general purpose network host computers (e.g., PDP-11/45) and small microprocessor-based units (e.g., 8085). The trend through the course of the program has continually moved toward smaller size, weight, and power.

The large number of site organizations involved, as well as the associated time periods, are indicated in Table I.

Conferencing has been of major importance in the packet speech experiments, and Table II summarizes features of conferencing experiments which have been carried out. Both centralized and distributed control techniques have been used for conference setup and for determination of which speaker has the floor at a given time. In ARPANET and SATNET, a conferee indicated his desire to talk by pushing a button, and indicator lights were used to inform the conferee that he had the floor. In later systems, a conferee could try to gain the floor by beginning to talk. A voice-controlled floor controller provided arbitration among multiple talkers. The voice-control strategy gener-

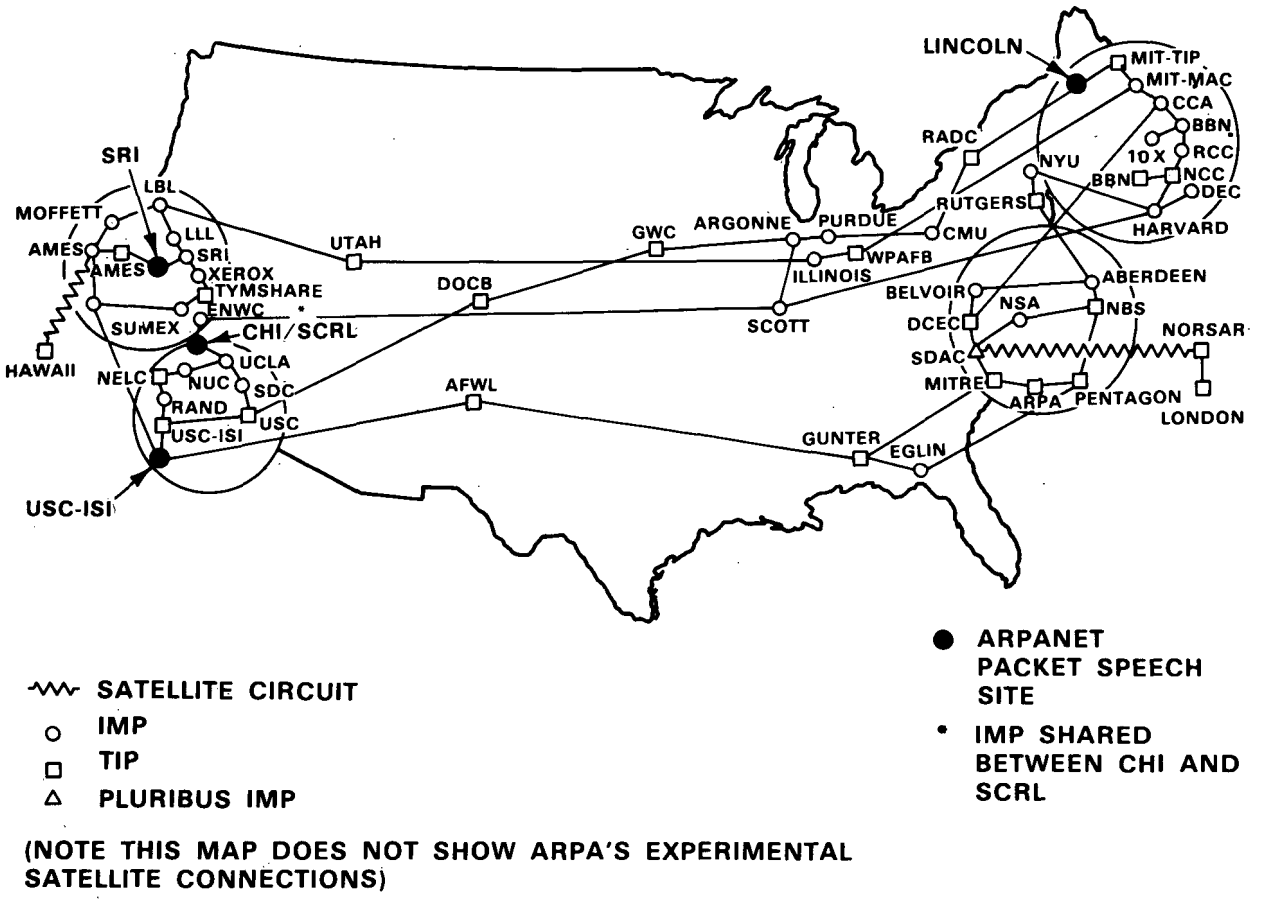

Fig. 5. Geographic map of the ARPANET, as of June 1975, showing locations of ARPANET packet speech sites.

ally gave more satisfactory performance from a human factors point of view [64]. The packet addressing mode is also important in conferencing. A broadcast mode avoids replication of voice packets or of conference control packets for multiple receivers.

IV. Packet Speech on the ARPA Network

A. ARPANET Characteristics

The ARPANET is a large store-and-forward packet-switching network [1] which interconnects computer facilities at a variety of locations. The network has been growing and evolving constantly since its initial four-node operation late in 1969. A June 1975 network map, representative of the topology in effect when most of the packet speech experiments were performed, is shown in Fig. 5. The network sites involved in the speech experiments were CHI, ISI, and SRI on the West Coast, and LL on the East Coast. The intersite distance among these locations (in number of hops on the shortest path) generally varied from 5 to 10.

Each ARPANET node generally consists of a communications processor called an interface message processor (IMP) developed by BBN. The IMP's are connected by 50 kbit/s lines according to the indicated topology. Host computers connected to the IMP's at each site deliver "messages" to the network with headers indicating the destination address. Depending on the number of bits in the message, it will be transmitted across the network by the IMP's as one or more ARPANET packets. The IMP's route each packet independently to the destination. As packets travel through the net on the lines between IMP's, they carry a packet header of approximately 160 bits. The maximum amount of user data that can be carried with each such packet is approximately 1000 bits.

B. Speech Transport in the ARPANET

The ARPANET characteristics lead to upper bounds on speech throughput due to the 50 kbit/s links and the transmission overhead, and lower bounds on delay due to the multiple hops generally required between source and destination. In addition, the original protocols developed for the ARPANET included reliability and flow control features which were designed appropriately for data communication, but which caused undesirable and unnecessary limitations on the throughput and delay for real-time speech. These limitations were present both in the packet delivery service provided by the IMP subnet between source and destination host, and in the original host/host or network control protocol (NCP) used in the ARPANET. Because of these limitations a new host/host protocol (NVP) was developed for speech and a new type of "uncontrolled" packet delivery service (suggested by Dr. R. E. Kahn) was introduced into the ARPANET.

The original NCP protocol implementations [9] generally allowed only one "message" at a time to be in flight

between a pair of processes in a source and destination host. The next message would not be sent until an acknowledgment, known as a request-for-next message (RFNM), was received from the destination. One motivation for the message-at-a-time limitation was to prevent a single user process in a multiuser host from dominating the host/IMP line. This "fairness" criterion was in conflict with the need to provide priority service to speech users. Messages could include up to 8063 bits of user data. Any message larger than the maximum packet size of 1008 bits would be broken up by the source IMP into a multipacket message to be transmitted across the net and reassembled by the destination IMP. High throughput could be attained by sending large multipacket messages. This is reasonable transport service for file transfers but sending multipacket messages for speech results in an undesirably large packetization delay. On the other hand, single-packet messages allow lower delay but result in severe throughput penalties, particularly for a path containing many hops. For example, a typical minimum round-trip time to send a 1000 bit single-packet message across a 10 hop ARPANET path and to receive a RFNM is about 0.3 s. The resulting peak throughput for the "message-at-a-time" protocol is $1000/0.3 = 3333$ bits/s with the average being significantly lower. Because of these restrictions NVP bypassed the NCP protocol modules which were available at the time when the network speech experiments were initiated and instead interfaced directly to the IMP subnet through the host/IMP protocol.

But the IMP subnet itself imposed important limitations on speech traffic. First, a restricted number of messages was allowed to be in flight between source and destination IMP's without a RFNM being received. When speech experiments started this number was 4; it was increased to 8 late in 1974. This restriction was imposed by IMP buffer space. More fundamentally, the IMP subnet provided reliable in-order end-to-end delivery of messages. If any message was lost and had to be retransmitted, all subsequent messages would be delayed to wait for the successful retransmission. This characteristic was reasonable for data terminal or file transfer traffic, but for speech it caused an occasional late packet to result in lengthy glitches. Fortunately, the rarity of packet errors in ARPANET did allow some successful speech communication despite this error control and sequencing.

For the above reasons, the new "type 3" packet delivery service was incorporated into the ARPANET by BBN on an experimental basis late in 1974. This new service allowed single-packet messages to be transmitted between selected hosts without end-to-end error control, without sequencing, and without a restriction on the number of packets in flight. This mechanism was used for most of the ARPANET packet speech experiments. Most of those experiments were conducted in conditions of light network loading. Use of type 3 packets in heavy load conditions was restricted to avoid the possibility of ARPANET congestion affecting all users.

Fig. 6 shows a comparison of cumulative round-trip delay distributions for type 0 (ordinary service with controls as described above) and type 3 messages between Lincoln and ISI (10 hops at the time the data were taken), taken in June 1975. Each message consisted of a single packet with approximately 1000 data bits. Packet rate was 8.6/s for net user bit rate of 8.6 kbits/s. At a 1 percent lost packet rate, type 3 is seen to provide about a 0.4 s advantage in overall delay. For higher rates type 0 became unusable whereas it was possible at the time to support 16 kbit/s CVSD with type 3 packets (but only during hours when network load was light). For lower rates, such as 2.4 kbits/s, the difference between type 3 and type 0 diminished. Currently, the ARPANET is much more heavily loaded and the results would change accordingly. Additional measurement results on ARPANET speech transmission are reported in [19].

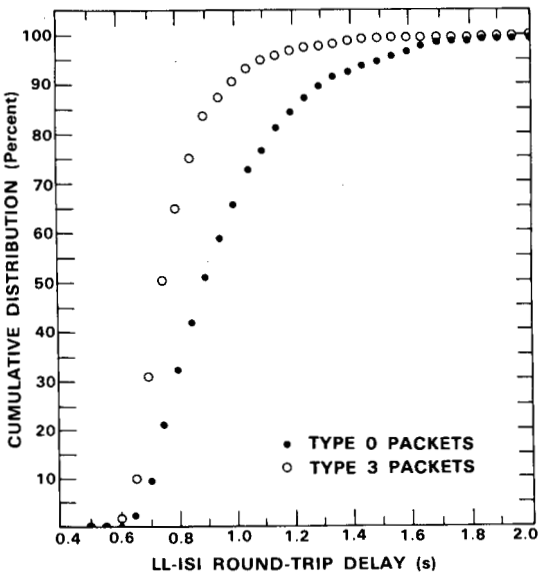

Fig. 6. Comparison of cumulative distribution of round-trip times for type 0 and type 3 ARPANET packets between LL and ISI (10 hops), measured in June 1975. Packet rate was 8.6 packets/s, with 1000 data bits/packet. Minimum around-trip delay on the path was observed to be about 0.6 s.

C. *ARPANET Speech System Implementations*

ARPANET speech systems were implemented at four sites, as indicated in Fig. 5 and Table I. All sites used different equipment but worked to a common NVP [16] specification. The success in bridging the gap among the systems was an important result in packet voice protocol development, and was achieved through the cooperation of many people in the ARPA packet speech community. The ARPANET speech systems were implemented in minicomputers such as the DEC PDP-11/45 to handle protocol processing with attached programmable signal processors to implement the speech encoding algorithms. Computer terminals were used for controlling call setup, and at some sites high-quality microphones and headphones were used instead of the conventional telephone handset. All sites had measurement software to record system performance. Much of the effort involved in these implementations went into programming the LPC encoding algorithms which were being developed during the same period. Several versions

of LPC at data rates from 5.0 kbits/s down to about 2.0 kbits/s were implemented and tested. An ARPANET cofferencing system was implemented with centralized floor control under a CHAIRMAN program running at one site. Conferees had pushbuttons to indicate desire to talk and lights to indicate when they had obtained the floor.

D. Milestone ARPANET Speech Experiments

The earliest packet-speech-related experiments on the ARPANET were conducted by Lincoln Laboratory in 1971 [14] using the TX-2 computer. Speech was not actually transmitted over the ARPANET, but an arrangement was set up whereby two persons could converse while experiencing in real time the effects of packetization and ARPANET delays. Speech was digitized (PCM) and stored. ARPANET delays were introduced by forming messages corresponding to blocks of speech and transmitting to a "fake host" at some IMP in the ARPANET. The fake host would discard the message and return an acknowledgment. Receipt of the acknowledgment was used to indicate that the corresponding block of data could be reconstituted at any time thereafter. Simulated speech bit rates from 2400 to 16 000 bits/s were used. Tests were performed on the effects of vocoder rate, block size, network distance (in hops), and reconstitution strategy. It was concluded that packet speech in a system with characteristics similar to a lightly-loaded ARPANET could be quite satisfactory from a human factors point of view.

The initial milestone in actual packet speech communication across the ARPANET was between ISI and LL, using 9.6 kbits/CVSD, in August 1974. CVSD quality at 9.6 kbits/s is quite poor, but the ARPANET was not capable of supporting 16 kbits/s at that time (Type 3 packets were not yet available.), and narrow-band vocoders were not available for use. In this and all other experiments, the average bit rate was reduced by transmitting packets only during talkspurts. In December 1974, the first LPC speech was communicated at 3.5 kbits/s over the ARPANET between LL and CHI. LPC conferencing at 3.5 kbits/s was first demonstrated in January 1975. Sites involved were CHI, ISI, LL, and SRI; all used different speech processors and host computers (Table I). In April 1978, LPC conferencing was demonstrated using a variable-frame-rate LPC [42] operating in the 2–5 kbits/s range. A 2.4 kbit/s LPC-10 for the ARPANET, first implemented at LL in 1979, was used for ARPANET/SATNET experiments and was later used for LPC experiments in the wide-band system. In addition to the real-time packet speech tests, a variety of experiments [59], [60] were also conducted in person–computer interaction by voice over the ARPANET.

V. Packet Speech on the Atlantic Packet Satellite Network

A. SATNET Characteristics

The Atlantic packet satellite network (SATNET) [4] is a packet-switched network that utilizes a distributed-control demand-assignment multiple-access (DAMA) algorithm called priority-oriented demand assignment (PODA) [2] to share a 64 kbit/s INTELSAT channel among earth stations in the United States and Europe. PODA in SATNET provides several important services for packet voice. First, it offers a type of service called a packet stream which can provide a guaranteed (except for priority preemption) data rate independent of network load. The stream service allows high utilization of the channel and minimizes the effect of network congestion on speech quality. Second, multiaddress packet delivery is provided in SATNET. This reduces the communication costs associated with voice conferencing by avoiding the need to send multiple copies of speech packets. Finally, a datagram service is provided in addition to the stream service. Data service involves the sending of a reservation request message via the satellite. As a result, packets with datagram service experience a cross-net delay at least 250 ms longer than that seen by packets traveling in streams. This type of service was used for control packets to avoid conflicts with the voice stream.

B. SATNET Speech System

Packet speech efforts on SATNET focused on voice conferencing [28] to take advantage of the multiaddress delivery capability. LPC speech at 2.4 kbits/s was used due to the limited bandwidth. The SATNET conferencing programs were designed to use the above features and also to explore the potential for distributed conference floor control. In a satellite net, distributed floor control achieves a delay advantage over centralized control of at least one satellite roundtrip.

In SATNET conferencing, the conference control programs (CCP's) at each site shared a common uplink stream to minimize use of capacity. On the downlink, stream packets were addressed simultaneously to all CCP's including the sender. The CCP's controlled access to this stream on a distributed basis. Communication of control packets was carried out via broadcast datagrams. Datagrams among CCP's were also used to resynchronize the conference when control errors occasionally caused two or more talkers to collide in the shared stream. Such collisions would be detected by the CCP receivers and recovery would be initiated.

Participants in the initial SATNET conferences were provided with a conference-control box equipped with push buttons and lights. A participant desiring to talk would push a want-to-talk (WTT) button which would cause a WTT message to be broadcast to all CCP's. On receiving that message, each CCP would add the participant to a WTT queue. Pushing a DONE-TALKING button would relinquish the floor by sending a control message in the voice stream. All CCP's assumed the head talker in the WTT queue to be the next speaker. Pushing the DONE-TALKING button would also remove a waiting participant from the WTT list.

A later version of SATNET conferencing employed voice control using SAD. A participant was allowed to transmit speech packets when none had been received within the last half second. A preassigned priority was used to resolve

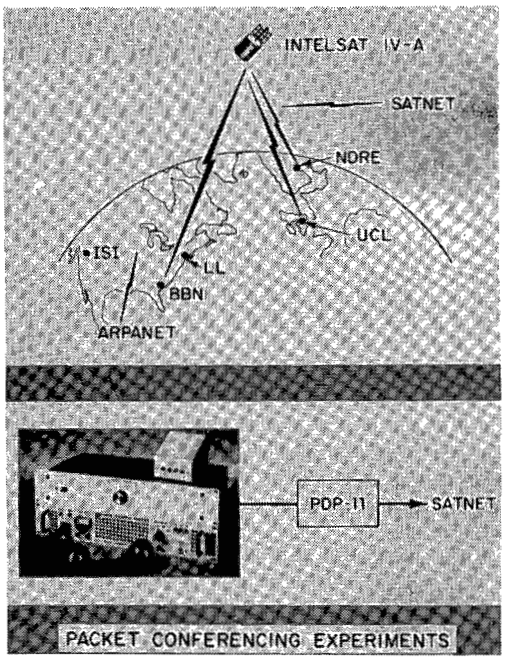

Fig. 7. Configurations of sites and equipment for SATNET and SATNET/ARPANET packet speech experiments.

collisions. Human factors studies [64] have concluded that voice control is preferable since it is easier to learn. Also, the queue associated with the push-button control sometimes leads to a "town meeting" effect where participants join the queue and then rehearse their speech instead of listening.

Hardware and software to support SATNET conferencing were developed by Lincoln Laboratory and installed at NDRE, UCL, and BBN. Hardware included a linear predictive vocoder [40], a PDP-11 interface, and a conference control box, all shown in Fig. 7. Voice protocol and conferencing software were implemented in PDP-11 SATNET host computers residing at the sites. Fig. 7 also shows the locations of SATNET conferencing sites and of the sites involved in SATNET/ARPANET internetwork conferencing.

C. SATNET/ARPANET Internetwork Speech System

To support internet conferencing, software was written for the SATNET host computer at BBN which also served as a gateway to ARPANET. The software made the BBN PDP-11 act as a special conferencing gateway. It functioned both as a participant and as the central controller in an ARPANET conference and as a participant in a simultaneous SATNET conference. Vocoder programs were written for the ARPANET sites to match the hardware vocoders at the SATNET sites. This internet system deomonstrated operation of a combination of centralized control and point-to-point packet delivery in the ARPANET with distributed control and broadcast delivery in SATNET. However, it pointed out the need for a more general approach to internetting since it was necessary to have very specialized software running in the gateway to deal with the different protocols in effect in the two nets. The new voice protocols developed for the wide-band network eliminate much of this specialization.

D. Experimental Results and Milestones

SATNET conferencing among the three sites using push-button control was first demonstrated in May 1978. The later version using voice control became operational in November 1979. Internet conferences were first carried out in September 1979 with SATNET participants at NDRE and UCL and ARPANET participants at LL and ISI. These systems have demonstrated the technical feasibility of packet voice conferencing in existing packet networks. SATNET conferencing, in particular, has demonstrated that the survivability advantages of distributed control can be achieved with little loss in conferencing performance.

VI. Packet Speech on the Experimental Wide-Band System

A. Introduction and System Overview

An experimental wide-band satellite-based packet system [52], [53] has been implemented to develop and demonstrate techniques for integrating packet voice with data in a realistic large scale network. The system is designed around a satellite channel with a capacity of 3.088 Mbits/s, in order to support many simultaneous voice connections. Whereas the ARPANET and SATNET were fundamentally data networks, on which limited speech experiments were performed, the wide-band system was designed specifically to accommodate speech. The wide-band system is configured as an internetwork where voice users reside on local networks and obtain access to the wide-band packet satellite network (WB SATNET) through gateways. This introduces a useful multiplexing hierarchy where traffic from local sources is first multiplexed by local nets and gateways, while the WB SATNET nodes in turn multiplex the satellite channel among aggregated traffic sources from the gateways at all the nodes.

The wide-band packet speech system development and the experimental program are sponsored by DARPA and involve a cooperative effort among a number of organizations as cited below. The Defense Communication Agency (DCA) has sponsored the satellite network development along with DARPA, and is utilizing the WB SATNET for a set of experiments supporting the development of the future defense switched network (DSN) [62], [63]. One of the four original network nodes is located at the Defense Communications Engineering Center (DCEC) in Reston, VA.

B. The Wide-Band Packet Satellite Network

The WB SATNET is a higher performance version of the Atlantic SATNET described in Section V-A. It uses the same DAMA algorithm (PODA) to share a 3.088 Mbit/s channel. The channel on the WESTAR III satellite and the earth stations are leased from Western Union, Inc. WB SATNET differs from the Atlantic SATNET in the use of earth stations with smaller antennas and has link budgets that result in bit error rates at 3.088 Mbits/s that require forward error correction of control packets to maintain

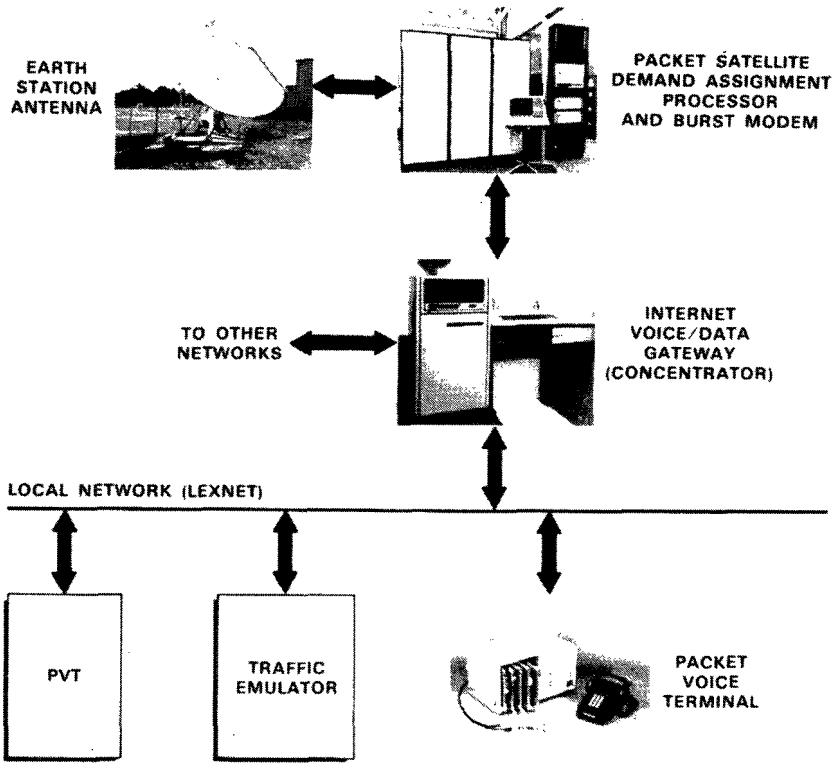

Fig. 8. Equipment configuration for typical wide-band network site.

synchronization of distributed PODA controllers. The earth station interface equipment provides multirate error correction to support this requirement. This error correction can also be applied to user data at the option of the user with consequent reduction in net data rate. The WB SATNET equipment at each site includes three major subsystems, a satellite earth station, a flexible burst modem called an ESI (earth station interface, developed by Linkabit, Inc.), and a packet satellite DAMA processor called a PSAT (pluribus satellite imp, developed by BBN) [55]. These WB SATNET subsystems are illustrated in Fig. 8 which also shows a traffic concentrator (i.e., a gateway) and a local net at the Lincoln site.

Features of the WB SATNET which are of interest for packet speech experiments are: 1) a sufficiently wide-band channel to support multiple voice users, even without narrow-band speech coding; 2) the capability for multiple coding rates to accommodate the different bit error rate requirements of speech and control packets; 3) stream reservations on the channel to provide guaranteed data rate and minimum (i.e., one hop) delay for speech; and 4) broadcast capability for efficient voice conferencing.

C. Wide-Band System Speech Facilities and the ST Protocol

Fig. 9 shows a map of the wide-band internetwork system, focusing on the primary local area facilities at each site. Internet voice/data gateways (G) based on a DEC PDP-11/44 minicomputer have been developed by Lincoln Laboratory and have been used for most of the wide-band system experiments. These gateways (Fig. 10), also referred to as "miniconcentrators," support both the experimental ST protocol and the DoD standard IP protocol. Key speech-related ST functions include obtaining satellite channel stream allocation based on local user bit rate requirements and concentrating speech packets from local terminals into aggregated packets for the WB SATNET. Table III lists major requirements for efficient packet speech transmission along with the approach used in ST to meet these requirements. Satellite channel allocation requests are ideally set on a statistical basis taking account of the fact that voice is transmitted only during talkspurts. The development of ST has been a major facet of the wide-band program. Although ST operates at an internet level in the wide-band system, the approach is valid for an individual network [29]. Gateway ST functions would be performed by network nodes in an individual net.

The PDP-11-based gateways are multiported and can provide simultaneous connections to more than one local net. Measurements have indicated an available throughput of 600–900 packets/s depending on packet lengths. More than one gateway can connect to a PSAT; the LL and ISI sites have both miniconcentrator gateways and a BBN-developed very high throughput multiprocessor concentrator/gateway referred to as the voice funnel [56].

Local broadcast cable networks (referred to as LEXNET's, for Lincoln Experimental Networks) [22]–[24] were developed at LL to efficiently support local packet voice and data traffic. LEXNET's have been installed and operated at all four sites. LEXNET is a 1.0 Mbit/s baseband cable network with distributed control, which uses a carrier-sense multiple-access protocol with collision detection (CSMA/CD) similar to that used in Ethernet. It utilizes a distributed algorithm for randomized retransmission which is specialized for voice traffic and which has been shown by simulation studies to provide high channel utilization for voice. The algorithm estimates competing

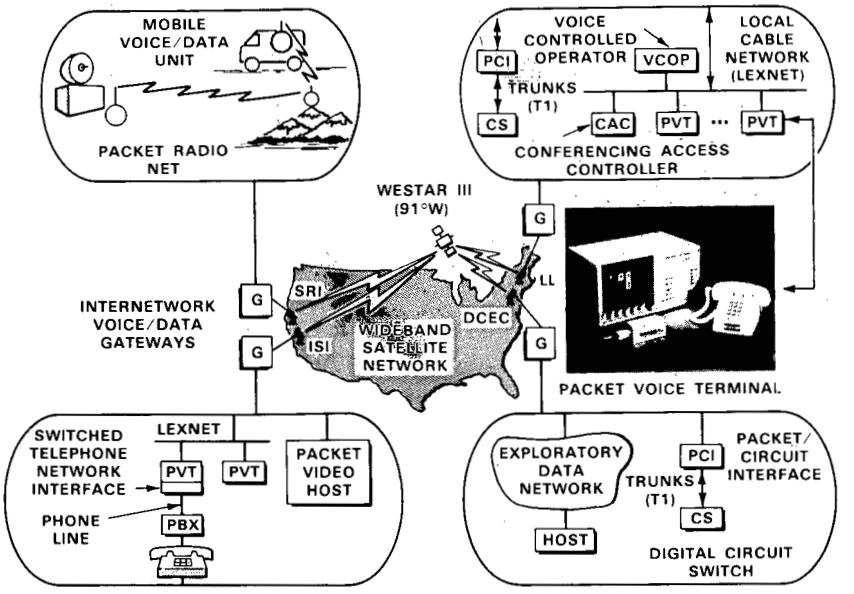

Fig. 9. Wide-band internetwork packet voice/data system, with illustration of primary local area facilities at each site.

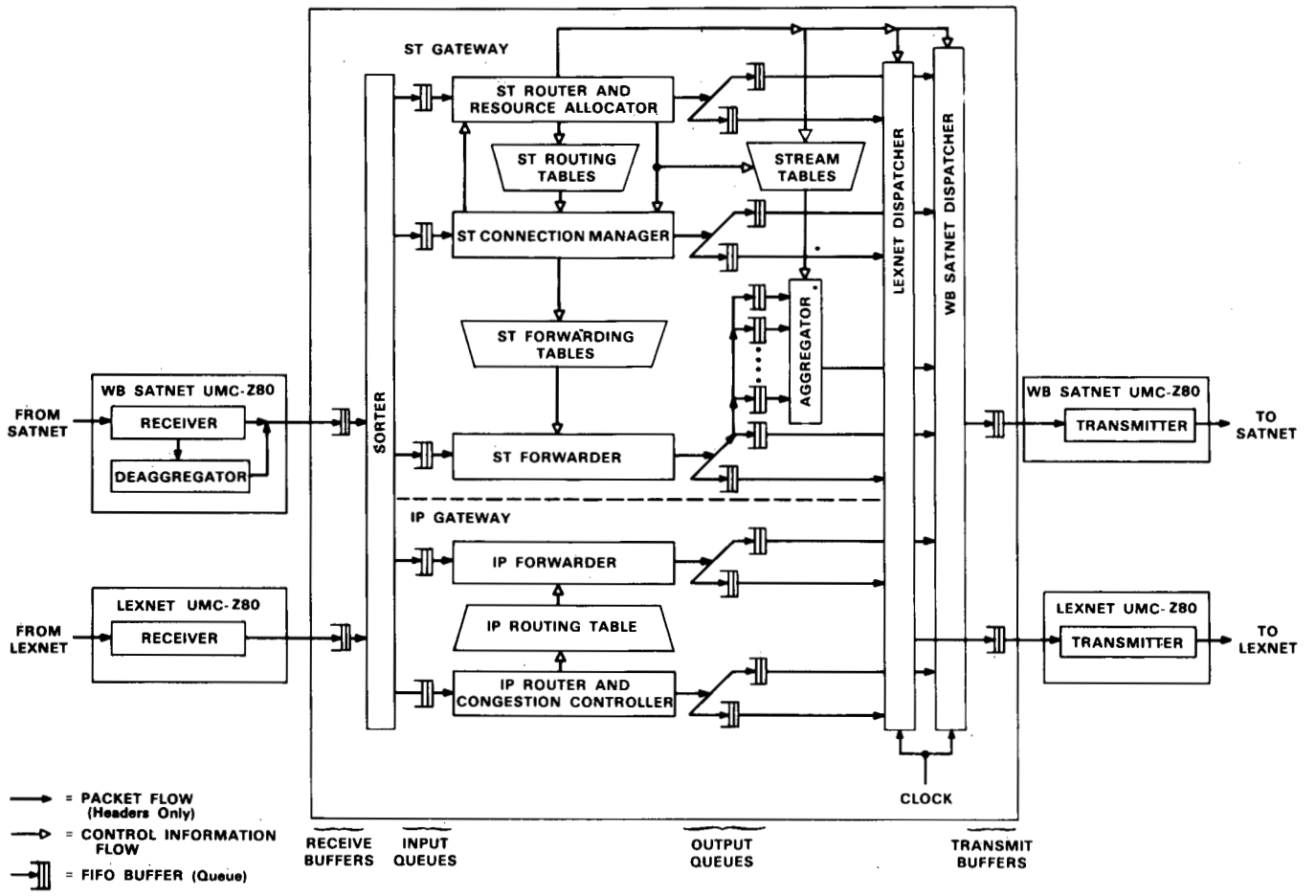

Fig. 10. Block diagram of miniconcentrator gateway. A PDP-11 central processor is used, and network interface processors (UMC-Z80 boards produced by Associated Computer Consultants) are included with special hardware interfaces for each attached network.

network activity and adjusts its retransmission interval based on the fact that voice terminals produce periodic packets during talkspurts. LEXNET's are populated by compact, microprocessor-based packet voice terminals (PVT's) [24] which provide full voice processing and protocol functions (see Fig. 11). The PVT's support 64 kbit/s PCM voice digitization or a choice of lower rate plug-in vocoders. In particular, Lincoln-built single-card 2.4 kbit/s LPC [41] and 16–64 kbit/s embedded CVSD (ECVSD) [39] units are available for experiments.

Conferencing using the second-generation voice protocols requires the services of a central access controller to

TABLE III
THE ST PROTOCOL FOR PACKET SPEECH

Packet Speech Requirements	ST Approach
1) Guaranteed data rate.	1) Know requirements in advance. Request reserved network resources when available (e.g., PODA streams). Assign loads to links statistically in routing virtual circuits.
2) Controlled delay (predictable dispersion).	2) Prevent congestion by controlling access on a call basis.
3) Small quantity of speech per packet.	3) Set up virtual circuit routes so that abbreviated headers can be used. Aggregate small packets for efficiency.
4) Efficiency equal to or better than circuit switching without TASI.	4) Abbreviated headers for packet efficiency. Goal of high link utilization with effective traffic control.
5) Efficient use of broadcast media.	5) Control multiaddress setup for conferencing and replicate packets only when necessary.

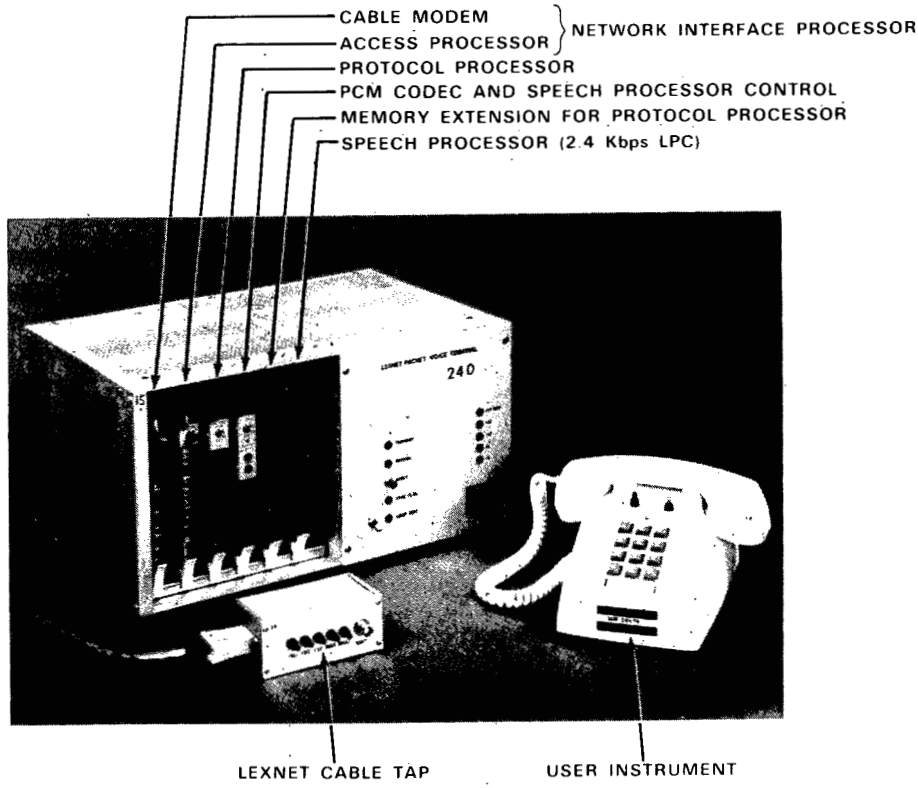

Fig. 11. Lincoln packet voice terminal. The three primary functional units are each controlled by an Intel 8085 microprocessor. The LPC unit utilizes three high performance signal processing microcomputers for analysis, synthesis, and pitch detection. The protocol processor supports NVP and ST and has a general interface to the access processor to allow adaptation to other networks. The user instrument has an 8085 which controls ringing, dial tone, etc. The PVT package is composed of approximately 200 integrated circuits, consumes 40 W, and occupies 0.75 ft^3 of volume.

assure uniqueness of conference connection identifiers throughout the network and to regulate access to particular conferences according to instructions provided by the conference originator. These functions are performed by the conferencing access controller (CAC) that resides on the LL LEXNET (the CAC address is assumed to be known to all PVT's and need not be dialed by users). The CAC is involved only in the process of setting up and taking down conferences and plays no part in the dynamic control of the conference "floor." It is implemented using PVT

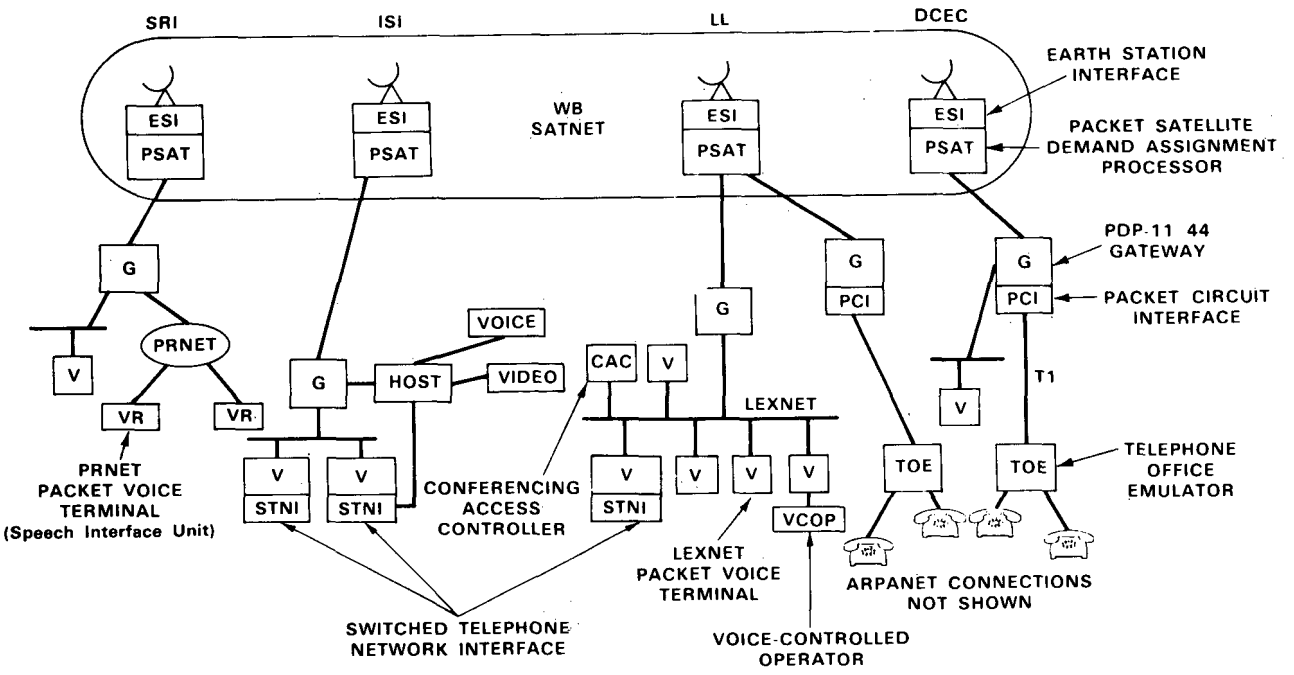

Fig. 12. Wide-band packet speech experiment status—September 1982.

hardware with special CAC software running in the protocol processor. A voice-controlled operator (VCOP), which allows conference setup via dialog with speech recognition and synthesis devices, is also resident on a LEXNET at LL [61].

The packet radio network (PRNET) located in the San Francisco Bay area [7] includes both fixed and mobile units, and both voice and data terminals. PRNET voice terminals [20], [21] include PDP-11/23-based speech interface units (SIU's) which implement voice protocols; speech coding is accomplished via 16 kbit/s CVSD units or CHI-5 2.4 kbit/s vocoders. Packet routing from the mobile PRNET to SRI can switch automatically as required from line-of-sight to double connectivity via hilltop repeaters. The PRNET, primarily designed for data, can support only limited voice traffic. But PRNET voice experiments have led to definition of a new PRNET type of service for better service of real-time voice [21]. In particular, voice service can be improved by allowing voice routes to change more rapidly than routes for data traffic.

Two kinds of interfaces are shown between the packet-switched network and circuit-switched systems. ISI has developed a switched telephone network interface (STNI) [57] which allows connection from individual telephone lines to the wide-band packet system. The STNI takes the form of a card which resides in a LEXNET PVT and allows the user to dial into the wide-band system from any ordinary telephone by first calling the STNI, which provides a second dial tone and accepts dialed digits addressing other PVT's. The STNI card handles translation of dialing and analog voice between the PVT and the public net, provides PCM digitization, and includes echo suppression. STNI's are currently installed at LL as well as at ISI. A packet video facility has also been developed by ISI to support low rate packet video experiments.

The packet/circuit interface (PCI) was developed by Lincoln under DCA sponsorship [54] to allow communication between packet switches and digital circuit switches in the T1 digital carrier format used for multiplexing of interswitch trunks in digital telephony. Telephone office emulators (TOE's) are provided to simulate the traffic from local digital circuit switches. The PCI is primarily being used for experiments in which a DAMA satellite is used as an overlay to a terrestrial circuit-switched net. These experiments are being carried out under DCA sponsorship to develop networking techniques applicable to the planned defense switched network which will utilize a mix of satellite and terrestrial media to provide survivable and economical telecommunications for DoD subscribers. The PCI/TOE facility has also been used to demonstrate interoperability between circuit-switched users (i.e., telephones on a TOE) and packet voice users on LEXNET PVT's. Each PCI provides up to four 64 kbit/s PCM trunks. In translating from T1 to packet format, the PCI must implement a subset of NVP and ST. The PCI thus performs the functions of a multiuser PVT, and in fact, carries out the protocol functions (both call setup and transport) for four simultaneous users. Special four-wire phones are provided at each TOE, but a COMSAT telesystems echo canceller is provided for access from standard two-wire phones. At DCEC, a gateway connection to an exploratory packet data network (EDN) is provided to help support packet data experiments in the wide-band system.

D. Experimental Results and Milestones

A snapshot of the wide-band internetwork packet speech system, as configured in September 1982, is shown in Fig. 12. All the internet packet speech capabilities implied by that figure have been demonstrated [53]. These include: multiple simultaneous PTP calls using PCM, ECVSD, and LPC; PCM and LPC conference calls using distributed floor control; voice internetting among LEXNET's, PRNET, and cirucit-switched systems; and conference

setup using VCOP. The new internet ST protocol has been implemented and tested successfully both in gateways and in terminals. Interoperation between miniconcentrator and voice funnel gateways has been demonstrated. Compatible LPC voice processing and NVP/ST protocols (both point-to-point and conferring) have been implemented in LEXNET PVT's and in PRNET SIU's.

The earliest major milestone in the achievement of the packet speech internet testbed capability occurred in November 1981 when two simultaneous PCM conversations were carried over WB SATNET between LL and ISI using PVT's on LEXNET's. One of these calls originated at an ordinary telephone extension at ISI and entered the wide-band system through an STNI. During 1982, the other capabilities were demonstrated: circuit-to-packet interconnection via PCI's in March; communication with a mobile PR terminal and multisite conferencing in June, and voice-controlled conference setup in October.

Current efforts are focused on performance measurements on the wide-band system, building on the basic demonstrated capability for internetting multiple voice users. A combination of real and emulated voice and data traffic is being applied to assess performance breakpoints in local nets, gateways, and the WB SATNET itself.

VII. Discussion and Conclusions

The successful system implementations and experiments described here strongly support the conclusion that packet communication is a practical technique for real-time speech communication. In cases where a user has already invested in a packet data communication network, adding a speech service to this network may well be a more economically attractive alternative than providing a separate speech service.

The great deal of interest in packet speech being shown by telecommunications companies, as evidenced by a number of current publications, including those in this current Special Issue of the IEEE Journal on Selected Areas in Communications, attests to the potential long-term advantages of packet techniques for integrated voice and data communication.

The work described here has provided a practical demonstration of the feasibility of packet speech in a large variety of packet network and internetwork environments. These system implementations have provided stimulus for the definition of packet speech requirements and for the successful development of speech processing techniques, voice protocols, packetization and reconstitution strategies, digital voice conferencing, and voice/data multiplexing. In addition, some of the advanced services possible through integration of voice and computer communication in the same network have been demonstrated, including voice interaction between computers and people in the network environment.

The vast investment in circuit-switched systems currently in existence makes it unlikely that packet techniques will soon become the dominant method for speech communication. However, as illustrated by the circuit/packet interoperability experiments described here, a useful coexistence of circuit-switched and packet-switched speech systems can be achieved. Meanwhile, the use of packet speech can be expected to grow over the next few decades.

Appendix
Acronyms and Abbreviations

AP120	—an early array processor developed by CHI, used for ARPANET speech
AP120B	—commercially-available array processor developed by Floating-Point-Systems, Inc.
BBN	—Bolt, Beranek and Newman, Inc., Cambridge, MA
CAC	—conference access controller
CCP	—conference control program; used for distributed control of SATNET packet speech conferences
CHI	—Culler-Harrison, Inc., Goleta, CA; now known as CHI Systems, Inc.
CHI-V	—array processor developed by CHI
CLPC	—compact LPC; single-card unit developed by Lincoln Laboratory
DEC	—Digital Equipment Corporation
ESI	—earth station interface; developed by Linkabit, Inc.
Ethernet	—CSMA/CD packet data cable network developed by Xerox
FDP	—fast digital processor; digital signal processing computer developed by Lincoln Laboratory
IMP	—interface message processor; the nodal processor in the ARPANET, developed by BBN
INTEL 8085	—microprocessor developed by INTEL Corporation
ISI	—Information Sciences Institute, Marina Del Rey, CA
LDVT	—Lincoln digital voice terminal; a programmable signal processing computer
LEXNET	—Lincoln experimental packet voice network
LL	—Lincoln Laboratory, Lexington, MA
LPC-10	—tenth-order linear predictive coding
LPCAP	—LPC array processor; an LPC voice processor developed by CHI
LPCM	—LPC microprocessor; an LPC vocoder developed by LL
LPVT	—LEXNET packet voice terminal; developed by LL
MP32	—host computer used at CHI for ARPANET packet speech
NDRE	—Norwegian Defense Research Establishment, Oslo, Norway
NVP	—network voice protocol
PCI	—packet/circuit interface; developed by LL
PDP-11	—a family of computers (programmable data processors) manufactured by DEC
PRNET	—packet radio network

PSAT	—multiprocessor packet satellite IMP developed by BBN for WB SATNET multiprocessor
PTP	—point-to-point
RFNM	—request-for-next-message; an acknowledgment message in ARPANET
SATNET	—the Atlantic packet satellite network
SCRL	—Speech Communications Research Laboratory
SF	—store-and-forward
SIMP	—satellite IMP; developed by BBN for SATNET
SIU	—speech interface units; developed by SRI
SPS-41	—a signal-processing computer developed by Signal Processing Systems, Inc.
SRI	—SRI International, Menlo Park, CA
ST	—stream protocol; an internet transport protocol for speech and other real-time traffic
STN	—switched telephone network
STNI	—STN interface; developed by ISI
T1	—standard digital carrier format used in telephony; operates at 1.544 mbits/s and carries 24 channels
TASI	—time-assigned speech interpolation; technique for saving bandwidth by transmitting only during talkspurts
TOE	—telephone office emulator; circuit switch emulator developed by LL
TX2	—host computer used at LL for early packet speech experiments
UCL	—University College, London
UMC-Z80	—a microprocessor-based input-output board used in the LL miniconcentrator gateway
VFR	—variable-frame rate; refers to vocoders operating at variable rate
WB SATNET	—the wide-band packet satellite network

Acknowledgment

The packet speech and wide-band network experiments and system developments described here were initiated by Dr. R. E. Kahn, DARPA Information Processing Techniques Offices, who has provided leadership and numerous technical contributions throughout the course of the work. Since 1978, Col. D. A. Adams, DARPA, has provided guidance, support, and technical contributions to the packet speech efforts. As noted in the text and references, the packet speech developments described here are the result of the efforts of many individuals at a number of organizations. For their contributions to the packet speech system developments and experiments, we specifically wish to cite the following individuals: D. Cohen, S. Casner, and R. Cole of ISI; E. Craighill of SRI; M. McCammon of CHI; and H. Heggestad, W. Kantrowitz, C. McElwain, and G. O'Leary of Lincoln Laboratory. We would like to acknowledge the technical contributions and cooperative efforts of these and many other colleagues who have made this paper possible.

References

Packet Networks

[1] L. Roberts and B. Wessler, "Computer network development to achieve resource sharing," in *Proc. Spring Joint Comput. Conf., AFIPS Conf.*, vol. 36, Montvale, NJ: AFIPS Press, 1970, pp. 543–549.
[2] I. M. Jacobs, R. Binder, and E. Hoversten, "General purpose packet satellite networks," *Proc. IEEE*, vol. 66, pp. 1448–1467, Nov. 1978.
[3] R. E. Kahn, "The introduction of packet satellite communications," in *Nat. Telecommun. Conf. Rec.*, Nov. 1979, pp. 45.1.1–46.1.6.
[4] W. W. Chu *et al.*, "Experimental results on the packet satellite network," *Nat. Telecommun. Conf. Rec.*, Nov. 1979, pp. 45.5.1–45.5.12.
[5] R. M. Metcalfe and D. R. Boggs, "ETHERNET: Distributed packet switching for local computer networks," *Commun. Assoc. Comput. Mach.*, vol. 19, pp. 395–404, 1976.
[6] D. D. Clark, K. T. Pogran, and D. P. Reed, "An introduction to local area networks," *Proc. IEEE*, vol. 66, pp. 1497–1516, Nov. 1978.
[7] R. E. Kahn, S. A. Gronemeyer, J. Burchfiel, and R. C. Kunzelman, "Advances in packet radio technology," *Proc. IEEE*, vol. 66, pp. 1468–1496, Nov. 1978.
[8] *Proc. IEEE*, Special Issue on Packet Commun., Nov. 1978.

Packet Network Protocols

[9] S. Carr, S. Crocker, and V. Cerf, "HOST–HOST communication protocol in the ARPA network," in *Proc. Spring Joint Comput. Conf., AFIPS Conf. Proc.*, vol. 36, Montvale, NJ: AFIPS Press, 1970, pp. 589–597.
[10] V. G. Cerf and R. E. Kahn, "A protocol for packet network interconnection," *IEEE Trans. Commun.*, vol. COM-22, May 1974.
[11] J. B. Postel, "Internetwork protocol approaches," *IEEE Trans. Commun.*, vol. COM-28, pp. 604–611, Apr. 1980.
[12] Special Issue on Computer Network Architectures and Protocols, *IEEE Trans. Commun.*, vol. COM-28, Apr. 1980.
[13] S. T. Kent, "Security in computer networks," in *Protocols for Data Communications*, F. F. Kuo, Ed.

Packet Voice

[14] J. W. Forgie, "Semiannual technical summary on graphics to the Advanced Research Projects Agency," M.I.T. Lincoln Lab., Lexington, MA, DTIC AD735-326, Nov. 30, 1971.
[15] —, "Speech transmission in packet switched store and forward networks," in *Proc. NCC*, 1975.
[16] D. Cohen, "Specifications for the network voice protocol," Univ. Southern California Inform. Sci. Inst., Rep. ISI/RR-75-39, Mar. 1976.
[17] R. F. Sproull and D. Cohen, "High level protocols," *Proc. IEEE*, vol. 66, Nov. 1978.
[18] D. Cohen, "Packet communication of online speech," *AFIPS Conf. Proc. NCC*, vol. 50, May 1981.
[19] S. L. Casner, E. R. Mader, and E. R. Cole, "Some measurements of ARPANET packet voice transmission," *Conf. Rec. Nat. Telecommun. Conf.*, Nov. 1978, pp. 12.2.1–12.2.15.
[20] P. Spilling and E. Craighill, "Digital voice communication in the packet radio network," in *Int. Conf. Commun.*, June 1980, pp. 21.4.1–21.4.7.
[21] N. Schacham, E. J. Craighill, and A. A. Poggio, "Speech transport in packet radio networks," submitted to *IEEE Trans. Commun.*
[22] G. C. O'Leary, P. E. Blankenship, J. Tierney, and J. A. Feldman, "A modular approach to packet voice terminal hardware design," *AFIPS Conf. Proc. NCC*, vol. 50, May 1981.
[23] D. H. Johnson and G. C. O'Leary, "A local access network for packetized digital voice communication," *IEEE Trans. Commun.*, vol. COM-29, pp. 679–688, May 1981.
[24] G. C. O'Leary, "Local access facilities for packet voice," in *Proc. 5th Int. Conf. Comput. Commun.*, Oct. 1980, pp. 281–286.
[25] E. R. Cole, "Packet voice: When it makes sense," *Speech Technol.*, pp. 52–61, Sept./Oct. 1982.
[26] D. K. Melvin, "Voice on ETHERNET—Now," in *Proc. Nat. Telecommun. Conf.*, New Orleans, LA, 1981.
[27] J. W. Forgie, "ST—A proposed internet stream protocol," unpublished memorandum.
[28] —, "Voice conferencing in packet networks," in *Conf. Rec. Int. Conf. Commun.*, June 1980, pp. 21.3.1–21.3.4.
[29] J. W. Forgie and A. G. Nemeth, "An efficient packetized voice/data network using statistical flow control," in *Conf. Rec. Int. Conf. Commun.*, June 1977.
[30] I. Gitman and H. Frank, "Economic analysis of integrated voice and data networks: A case study," *Proc. IEEE*, vol. 66, pp. 1549–1570, Nov. 1978.
[31] J. W. Forgie, "Network speech implications of packetized speech,"

Annu. Rep. Defense Commun. Agency, M.I.T. Lincoln Lab., Lexington, MA, DTIC AD-A45455, Sept. 30, 1976.
[32] C. K. McElwain, "Protocol software for a packet voice terminal," M.I.T. Lincoln Lab., Lexington, MA, Tech. Rep. 633, 1983, to be published.
[33] T. Bially, B. Gold, and S. Seneff, "A technique for adaptive voice flow control in integrated packet networks," *IEEE Trans. Commun.*, vol. COM-28, pp. 324–333, Mar. 1980.

Voice Processing

[34] B. Gold, "Digital speech networks," *Proc. IEEE*, vol. 65, pp. 1636–1658, Nov. 1977.
[35] J. L. Flanagan *et al.*, "Speech coding," *IEEE Trans. Commun.*, vol. COM-27, Apr. 1979.
[36] L. R. Rabiner and R. W. Schafer, *Digital Processing of Speech Signals.* Englewood Cliffs, NJ: Prentice-Hall, 1978.
[37] J. D. Markel and A. H. Gray, Jr., *Linear Prediction of Speech.* New York, NY: Springer-Verlag, 1976.
[38] G. Kang, L. Fransen, and E. Kline, "Multirate processor (MRP)," Naval Res. Lab. Rep., Sept. 1978.
[39] J. Tierney and M. L. Malpass, "Enhanced CVSD—An embedded speech coder for 64–16 kbps," *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Atlanta, GA, Mar. 30–Apr. 1, 1981.
[40] E. M. Hofstetter, J. Tierney, and O. Wheeler, "Microprocessor realization of a linear predictive coder," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-25, pp. 379–387, Oct. 1977.
[41] J. A. Feldman and E. M. Hofstetter, "A compact, flexible LPC vocoder based on a commercial signal processing microcomputer," presented at Electro '82, Boston, MA, May 25–27, 1982, Session 22/5.
[42] V. R. Viswanathan, J. Makhoul, R. M. Schwartz, and A. W. F. Huggins, "Variable frame rate transmission: A review of methodology and application to narrow-band LPC speech coding," *IEEE Trans. Commun.*, vol. COM-30, pp. 674–686, Apr. 1982.

TASI

[43] P. T. Brady, "A statistical analysis of on-off patterns in 16 conversations," *Bell Syst. Tech. J.*, vol. 47, Jan. 1968.
[44] K. Bullington and J. Fraser, "Engineering aspects of TASI," *Bell Syst. Tech. J.*, vol. 38, Mar. 1959.
[45] S. J. Campanella, "Digital speech interpolation," *COMSAT Tech. Rev.*, vol. 6, Spring 1976.

Echo Control

[46] M. Sondhi and D. Berkley, "Silencing echos on the telephone network," *Proc. IEEE*, vol. 68, p. 991, Aug. 1980.
[47] D. G. Messerschmitt, "Echo control in the switched telephone network interface to a packet speech terminal," unpublished memorandum.

Multiplexing

[48] T. Bially, A. J. McLaughlin, and C. J. Weinstein, "Voice communication in integrated digital voice and data networks," *IEEE Trans. Commun.*, vol. COM-28, pp. 1478–1490, Sept. 1980.
[49] C. J. Weinstein and E. M. Hofstetter, "The tradeoff between delay and TASI advantage in a packetized speech multiplexer," *IEEE Trans. Commun.*, vol. COM-27, pp. 1716–1720, Nov. 1979.
[50] W. Kantrowitz, C. J. Weinstein, and E. M. Hofstetter, "Prediction and buffering of digital speech streams for improved TASI performance on a demand-assigned satellite channel," M.I.T. Lincoln Lab., Lexington, MA, Tech. Note TN1979-75, DTIC AD-A081593/6, Nov. 1979.
[51] J. G. Gruber, "Delay related issues in integrated voice and data networks," *IEEE Trans. Commun.*, vol. COM-29, pp. 786–800, June 1981.

Experimental Wide-Band Network

[52] C. J. Weinstein and H. M. Heggestad, "Multiplexing of packet speech on an experimental wideband satellite network," *Proc. AIAA 9th Commun., Satellite Syst. Conf.*, San Diego, CA, Mar. 1982.
[53] H. M. Heggestad and C. J. Weinstein, "Experiments in voice and data communications on a wideband satellite/terrestrial internetwork system," in *Conf. Rec. Int. Conf. Commun.*, Boston, MA, June 1983.
[54] Annual Report to the Defense Communications Agency on Network Speech Syst. Technol., M.I.T. Lincoln Lab., Lexington, MA, Sept. 30, 1981.
[55] G. Falk, S. Groff, W. M. Milliken, and R. Koolish, "PSAT technical report," Bolt, Beranek and Newman Inc., Rep. 4469, May 1981.
[56] R. Rettberg *et al.*, "Development of a voice funnel system," Bolt, Beranek and Newman Inc., Rep. 4816 and 4817, Mar. 1982.
[57] I. H. Merritt, "Providing telephone line access to a packet voice network," Univ. Southern California, Inform. Sci. Inst. Rep. ISI/RR-83-107.
[58] V. J. Sferrino, "A multiport buffer memory for an internet packet voice/data gateway," M.I.T. Lincoln Lab., Lexington, MA, Tech. Rep. 612, DTIC AD-A119157, July 1982.

Speech Recognition in Packet Voice Systems

[59] D. H. Johnson and C. J. Weinstein, "A phrase recognizer using syllable-based acoustic measurements," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-26, Oct. 1978.
[60] —, "A user interface using recognition of LPC speech transmitted over the ARPANET," in *EASCON Rec.*, Sept. 1977.
[61] Semiannu. Tech. Summary Defense Advanced Res. Projects Agency on Packet Speech Syst. Technol., M.I.T. Lincoln Lab., Lexington, MA, Sept. 30, 1982.

Defense Switched Network

[62] C. J. Coviello and R. H. Levine, "Considerations for a defense-switched network," in *Nat. Telecommun. Conf. Rec.*, Nov. 1980.
[63] R. P. Lippmann, "Steady-state performance of survivable routing procedures for circuit-switched mixed-media networks," M.I.T. Lincoln Lab., Lexington, MA, Tech. Rep. 633, DTIC AD-A126213, 1982.

Voice Conferencing Studies

[64] J. W. Forgie, C. E. Feehrer, and P. L. Weene, M.I.T. Lincoln Lab., Lexington, MA, Final Rep. Voice Conferencing Technol., DTIC AD-A074498/7, Mar. 31, 1979.

Clifford J. Weinstein (S'66–M'69) was born in New York, NY, on April 7, 1944. He received the S.B., S.M., and Ph.D. degrees in electrical engineering in 1965, 1967, and 1969, respectively, all from the Massachusetts Institute of Technology (M.I.T.), Cambridge, MA.

Since 1967 he has been with the M.I.T. Lincoln Laboratory, Lexington, MA, where he is currently Leader of the Speech Systems Technology Group whose activities include programs in packet voice, integrated voice/data networks, speech processing algorithm development, and signal processor architecture and implementation. His major technical activities have been in communication networks for voice and data, automatic speech recognition, speech bandwidth compression systems, and digital signal processing.

Dr. Weinstein is a Past Chairman of the IEEE Acoustics, Speech, and Signal Processing Society's Technical Committee on Digital Signal Processing and is a member of Eta Kappa Nu, Tau Beta Pi, and Sigma Xi.

James W. Forgie (S'50–A'52–M'56) was born in Washington, PA, in 1929. He received the B.S. degree in electrical engineering in 1951 from the Massachusetts Institute of Technology (M.I.T.), Cambridge, MA.

He joined the M.I.T. Lincoln Laboratory, Lexington, MA, as a Research Assistant in 1951, and has remained with that organization where he is currently a Senior Staff Member of the Speech Systems Technology Group. His research activities have involved digital computer hardware and software design (Whirlwind I and TX-2), computer networking, automatic speech recognition, voice conferencing, and packet voice communications.

Mr. Forgie is a Fellow of the Acoustical Society of America, a member of the Association for Computing Machinery, and Sigma Xi.