ESD-TR-77-178

Annual Report

Network Speech System Implications of Packetized Speech

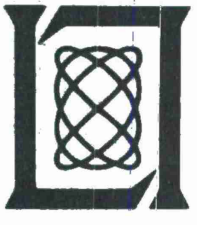
30 September 1976

Prepared for the Defense Communications Agency
under Electronic Systems Division Contract F19628-76-C-0002 by

Lincoln Laboratory

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

LEXINGTON, MASSACHUSETTS



The work reported in this document was performed at Lincoln Laboratory, a center for research operated by Massachusetts Institute of Technology, for the Military Satellite Office of the Defense Communications Agency under Air Force Contract F19628-76-C-0002.

This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER

Raymond L. Loiselle, Lt. Col., USAF
Chief, ESD Lincoln Laboratory Project Office

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
LINCOLN LABORATORY

NETWORK SPEECH

SYSTEM IMPLICATIONS OF PACKETIZED SPEECH

ANNUAL REPORT
TO THE
DEFENSE COMMUNICATIONS AGENCY

1 JANUARY – 30 SEPTEMBER 1976

ISSUED 16 SEPTEMBER 1977

LEXINGTON MASSACHUSETTS

ABSTRACT

The effects, examined in parametric fashion, on the overall voice quality, acceptability, and communicability of speech packetization and its transmission through a packet-switched network. Speech processed through a number of real-time simulation programs developed to create anticipated anomalies (glitches) in packet speech systems were evaluated by informal listening, communicability experiments, and formal acceptability testing. Depending on system design parameters, test results indicate that packet-system speech quality varies from essentially perfect (no packet-related anomalies) to unusable. Guidelines are provided for an acceptable packetized speech communication system.

CONTENTS

ACKNOWLEDGMENTS

The network and anomaly simulation programs described are the work of J. D. Drinan. The reconstitution algorithm was programmed by C. K. McElwain. The speech encoding and decoding programs were contributed by E. M. Hofstetter and S. Seneff. The report was prepared by J. W. Forgie.

NETWORK SPEECH

I. REPORT PLAN

This report is the output of a study undertaken to examine in parametric fashion the effects that packetization of speech and its subsequent transmission through a packet-switched network would have on the overall voice quality, acceptability, and communicability in such a system. While the feasibility of packet speech communication has been demonstrated in experiments carried out with the ARPANET,[1] the practicality of the technique under other network conditions has not been fully demonstrated. Since integration of voice and data traffic in the same network could be an economical way to handle both modes of communication, and packet technology is one approach to such integration, it was decided to undertake a systematic study of the effects of packetization on voice quality to provide basic data for use in further network simulation and analysis studies.

Lincoln Laboratory personnel developed a number of real-time simulation programs for this study that can create the kinds of anomalies (glitches) that can be expected to occur in packet speech systems. Speech processed through these simulation programs has been evaluated by informal listening, communicability experiments, and by formal acceptability testing. Parameters for the simulations permit speech quality to vary from essentially perfect (no packet related anomalies) to unusably bad. In addition, a full duplex network simulation was developed to test the effectiveness of algorithms for reconstituting speech at the receiver of a packet speech stream and to explore the interaction among such parameters as buffer sizes and network delay characteristics. Parameter choices again produce the same extreme results.

Since the study did not focus on any particular set of network characteristics, the outcome is not a yes/no judgment as to the acceptability of packetized speech communication. Rather, the outcome is a set of guidelines, which, if followed, can result in an acceptable speech communication system.

The report plan then is: Section II discusses those aspects of a packet-switched speech system that are different from conventional circuit-switched speech systems. Section III discusses the types of anomalies that can be expected to occur occasionally in packet systems and describes their subjective effects. Section IV describes the programs developed to simulate the anomaly situations. Section V describes the full duplex network simulation created to explore communicability and discusses an adaptive reconstitution algorithm in some detail. Section VI presents the results of limited formal acceptability tests and summarizes the main conclusions drawn from the study. Other results and observations are provided in the discussion of anomaly effects (Section III) and the rationale for the adaptive reconstitution algorithm (Section V-B).

II. PACKET SPEECH SYSTEM CONSIDERATIONS

A basic requirement for a packet speech system is that the network be capable of handling packet streams for each active speech subscriber at the data rates required by the subscriber's speech encoder. If overall network load rises to the point where the average data rate available to a subscriber falls below the requirement of the encoder, conversational speech communication is not possible for that subscriber. For satisfactory speech, a packet network must embody flow control and/or priority mechanisms that guarantee that, once accepted by the system, a subscriber is permitted either an adequate data rate or his connection is preempted by a higher priority need. These mechanisms are not inherent in packet networks; for example, they are lacking in the ARPANET. For this study, we assumed that a network has such mechanisms, and concentrated on such other aspects of packet networks that can affect the acceptability of a packet speech system: variability in network delays, action of speech activity detectors, and behavior of reconstitution algorithms.

A. PACKET NETWORK DELAYS

When speech is transmitted in a packet-switched network, the voice signal is digitized, the resulting bit stream is chopped into segments, address bits are added, and the resulting packets are given to the network for delivery to the addressee. Each packet travels through the network as a unit, forwarded from node to node according to some routing algorithm and experiencing delays caused by other traffic in the net as well as bit rates and propagation delays introduced by communication links in the network. Delays will vary from packet to packet, enough so that in some cases packets may arrive at the receiver in an order different from that in which they were generated at the transmitter. If bit errors occur during transmission, the affected packets may fail to arrive or arrive very late due to retransmission by the network.

Overall delay in a packet speech system has three components. First, the time required to accumulate a packetsworth of speech at the transmitter. This component is determined by the ratio of the packet size to the bit rate of the speech encoder. Second, the transit time of the network, which depends on packet length, communication link capacities and propagation times, network topology and control algorithms, and other network traffic. Interactions among these factors have been discussed at length in other reports,[2,3,4] and the discussion is not repeated here. Third, the delay that must be added at the receiver to smooth the jitter in packet arrivals in order to play out speech of acceptable quality. This component, called the "reconstitution delay," should be adjusted so that almost all the packets arrive as needed. The sum of the second and third components equals the worst-case transit delay that a packet can experience and still contribute to output speech.

Early packet speech experimentation was carried out with the ARPANET. Significant round-trip delays (between 1 and 2 seconds) occurred due to a combination of 50-kbps lines and many node-to-node hops. These long delays are not inherent in packet technology, but rather, result from the particular combination of network characteristics used in ARPANET. Other studies[5] indicate that overall packet network delays can be comparable to fixed-TDMA, digital circuit switching if satisfactory speech transmission is an established goal that is set when the network is designed.

B. SPEECH ACTIVITY DETECTION

The advantage of packet switching over circuit switching for data communications results from the ability of a packet network to effectuate statistical multiplexing of many data sources

to make use of high peak-to-average data rates that are characteristic of these sources, thus handling more active subscribers than would be possible in the same network using circuit-switching techniques. It is desirable to attempt to gain a similar advantage when speech is transmitted in a packet net by sending voice packets only when voice energy is present at the transmitter. Experience with TASI[6] shows that a packet network would gain about a factor of two in increased capacity so long as communication links in the net have sufficient capacity to handle 50 or more conversations. The advantage for voice traffic with links of lower capacity would be less, but the unused capacity could be used by data traffic. We assumed that a packet-speech transmission system would use speech activity detectors to take advantage of the otherwise wasted channel capacity.

C. RECONSTITUTION ALGORITHMS

When voice packets arrive at their destination they are generally held in a buffer memory until played out. The process of buffering incoming packets and deciding exactly when to play them out is governed by a "reconstitution algorithm." Since packets can be lost in the net, there is no value of reconstitution delay that can guarantee that all packets will have arrived when needed. A practical algorithm must be designed to accept some probability of packet loss (due to real loss or late arrival). If the network can deliver packets out of order, the transmitter must attach a packet sequence number and/or time stamp permitting the reconstitution algorithm to correct the order on playout.

The degree of complexity in a reconstitution algorithm depends on the network's delay characteristics. If network delays are short, as would be the case if link capacities were high and packet sizes kept short, a fixed reconstitution delay could be used. If delays are long or dispersion varied greatly with average network loads, it would be desirable to use an adaptive algorithm that could adjust the reconstitution delay to effect a compromise between packet loss and overall delay. (See Section V-B for a description of such an algorithm.)

Another 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 situation can result from a late or lost packet or it can simply mean that the talker has paused and there are no more packets on the way. It might seem that the transmitter could have marked the last packet of a talkspurt so that the receiver could distinguish that case, but the transmitter does not know when it sends a packet that it will not be sending a successor. Even though speech energy was below an activity threshold at the end of a packet, it may go above the threshold during the next packet interval. As a result the reconstitution algorithm should take the same action in either case. (When the receiver "knows" a packet is missing because a successor to the missing packet has arrived, the action taken may be different.)

Two basic alternatives are available to deal with the absence of new speech data for playout. The first is to send a bit pattern to the speech decoder that results in no output. This alternative fills gaps with "silence" which is correct if the gap is a natural pause, but which produce a detectable anomaly if the gap is due to a lost or late packet. The second alternative is to repeat the last segment of real speech data until new data become available. This alternative works reasonably well for a framed encoding technique such as a linear predictive coding (LPC) or channel vocoder, particularly if the last frame is unvoiced. If the gap is very short, it may be less noticeable when filled in this manner than when filled with silence. If longer, however, and particularly if the last real frame had considerable energy, a gap filled in this way will be much

more noticeable than when filled with silence. For a real pause there is a good chance that repetition will be satisfactory since the last frame has very low energy and its repetition will approximate silence. Repetition will probably be less satisfactory for a frameless encoding scheme such as pulse code modulation (PCM) or continuously variable slope delta (CVSD) modulation. For such schemes, repetition of the last n samples produces a periodic signal that is likely to sound unnatural, particularly for a long gap.

We adopted a mixed strategy, filling long gaps with silence and short gaps with repeated frames that are always made voiceless and have energy values that decay with time. The exact techniques used vary with the encoding scheme (Section IV-A).

III. PACKET SPEECH ANOMALIES

Packet speech anomalies result when packets fail to arrive in a timely fashion or when the speech activity detector fails to function as intended. Four types of anomalies that can result from variations in network delay characteristics plus that expected from speech activity detector inadequacy are discussed. A real network might exhibit all types in varying degrees, but it is best to assess their subjective effects independently.

A. LOST PACKETS

The most frequently observed anomaly is the gap caused by a lost or late packet (Fig. III-1). Time is represented horizontally; however, network delays and packet lengths are not to scale. The top rectangles indicate equal-size packets being transmitted when filled. The upper arrows represent transmission through the net. Their lack of parallelism reflects the dispersion of network delays. The lower arrows represent time spent in the receiver's buffer waiting for the reconstitution algorithm to call for playout of the packet. The dashed arrows represent the two causes for output gaps.

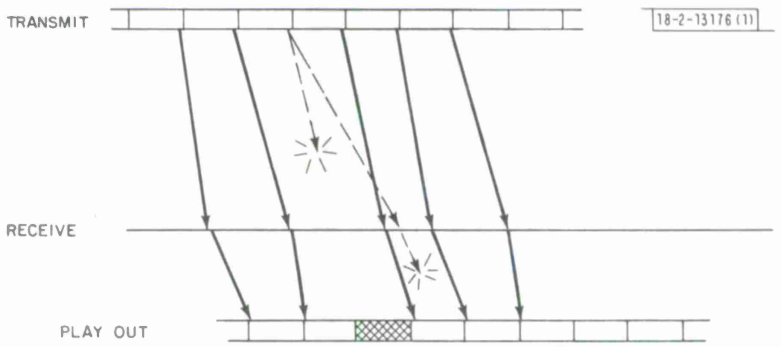


Fig. III-1. Lost packet anomaly: lost in the network or delivered too late.

Obviously, a late packet could arrive during its nominal playout time instead of entirely too late (Fig. III-1). The reconstitution algorithm could then reduce the size of the gap by playing out the remaining portion of the packet (see Section V-B). So long as the overall timing of the talkspurts is retained, such gaps are considered to be lost packet gaps.

The subjective effect of lost packet anomalies depends on the duration and frequency of the gaps. If the gaps are short (20 to 50 msec) and infrequent (one every ten seconds or so) they are likely to pass completely undetected in conversational situations. If short gaps become more frequent (one per second or more often), the reconstituted speech takes on a rough, chewed up quality, but speech intelligibility is not seriously reduced until the gaps are frequent enough to destroy one quarter to one half of the packets.

If the gaps are long (100 msec or more) the gap can cause the loss of complete syllables, words, or even phrases. Such gaps are clearly perceived as gaps, and not as a quality degradation. Where they occur determines sentence intelligibility. A long gap could seriously alter the meaning of a sentence. In one experimental run, a lost packet gap neatly removed the word "not," inverting the meaning of the sentence. The silence that replaced the "not" did not even cause an unnatural-sounding pause to indicate that an anomaly had occurred.

For lost packet anomalies of intermediate length, quality effects and loss of intelligibility are observed. The damage depends on where the gap falls in relation to the speech material.

B. BUFFER UNDERFLOW

If during a talkspurt the average delay through a network increases gradually or abruptly so that the reconstitution delay is no longer sufficient to smooth the output speech, an adjustment in the timing of the output speech is required to allow for the increased network delays. We call this condition "buffer underflow" because the receiver runs out of speech to play out. In an example of a double gap and a timing shift caused by such a situation (Fig. III-2), the difference between the slopes of the dashed lines indicate the timing shift. No speech is lost with this type of anomaly.

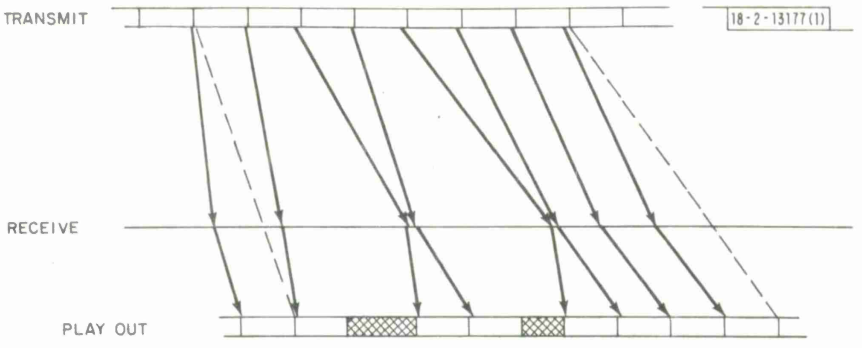


Fig. III-2. Underflow anomalies caused by increasing network delays.

The subjective effect of an underflow anomaly of given duration tends to be greater than that of a lost packet of the same duration. The increased effect appears to be due to the shift in timing of the continuation of the talkspurt. This is, to some extent, a surprising result because no speech is lost with such an anomaly, but the prosodic information carried by speech tempo is sufficiently great to cause the large effect. When the anomaly occurs, it produces a hesitation in the output speech that sometimes seems natural because it occurs at a time when the speaker might have hesitated, but at other times seems very unnatural, coming perhaps in the middle of a vowel or consonant. While an underflow anomaly is unlikely to affect the intelligibility of a sentence directly, it could cause a subtle shift of meaning such as suggesting an element of uncertainty when, in fact, the speaker had no such intent.

Underflow anomalies need not occur with any great frequency in a packet speech system, but they are probably unavoidable in a network where average delays can increase suddenly due to the necessity to change routes to bypass failed links or nodes or due to peak load problems that cause a sequence of packets to be abnormally delayed. When such an increase in short-term average delay occurs, the reconstitution algorithm should increase reconstitution delay to avoid a burst of lost packet gaps. If the shift in timing cannot be carried out during a natural pause, an underflow anomaly is produced. If the increase is somewhat gradual, as might be the case if it resulted from increasing network traffic, the reconstitution algorithm can prepare for the event by observing the increasing lateness of packet arrivals relative to expectations. In such case, an adjustment can be made without any speech loss. If the increase is sudden, as could be the case in the event of link or node failure, some packets may be treated

as lost before the algorithm concludes that an underflow condition exists and takes corrective action.

C. BUFFER OVERFLOW

Buffer overflow anomalies can occur when short-term average delays are decreasing, causing packets to be delivered at a higher rate than the speech is being played out. If buffer space at the receiver is inadequate to contain the buildup, speech is lost and there is a corresponding shift in speech timing. Such anomalies should not occur in a well-designed packet speech system, except under extreme circumstances. While decreases in short-term average delays can be expected under normal conditions, buffering is usually sufficient to permit the corrective delay adjustment to take place during a natural pause in the speech, in which case, no anomaly results.

Overflow anomalies are more damaging to reconstituted speech than either lost packet or underflow anomalies since they cause both a loss of speech and a timing shift. We explored two ways to handle speech loss: (1) Throw away a contiguous speech segment of duration equal to the overflow, and (2) spread the loss over a period of time by throwing out every other or every third frame of vocoded data to effect an apparent speedup of the output process. The second approach reduces the subjective effect somewhat, but not enough to suggest that a designer is likely to accept even a one-tenth percent probability of the occurrence of these anomalies in order to save the cost of adequate buffering.

D. BUFFER UNDERFLOW PLUS OVERFLOW

It is not unreasonable to imagine that a network could accumulate packets behind some blockage in the net and then deliver the packets in a burst. A listener would experience an underflow anomaly and then an overflow anomaly. As might be expected, the subjective effect of the combination is greater than either one occurring alone. In addition to the jarring effect of two timing shifts in rapid succession, there is likely to be loss of intelligibility due to the overflow. Obviously, adequate buffering could reduce this to the underflow case if the network delivered the blocked packets, but the combined anomaly may be forced on the receiver if the blocked packets are discarded within the net. Our listening experience suggests that a network causing such events on a regular basis would be unsatisfactory for speech communication. However, there is no way for a designer to guarantee that one or all of these anomalies will not occur. Good design should be able to avoid underflow plus overflow anomalies except in the hopefully rare case of node or link failures.

E. SPEECH ACTIVITY DETECTION FAILURES

We assumed that speech activity detection in a packet speech system would make use of simple energy measurements to decide the presence or absence of speech. This technique is used in TASI systems and appears to operate satisfactorily under field use. Obviously, if background noise reaches amplitudes comparable to the voice signals entering the system, the noise is considered to be speech, encoded and transmitted. It is possible to improve the selectivity of a speech activity detector with sophisticated techniques,[7] but such techniques are likely to be too expensive for general use and were not included in our study.

A speech activity detector can fail either by rejecting speech that should have been accepted or accepting background noise that should have been rejected. Anomalies produced by false rejection are more damaging since they can cause loss of sentence intelligibility. There are two types of failure that result in loss of speech information. One is the clipping of weak sounds at the beginning and end of talkspurts. This difficulty can be largely overcome by the proper use of anticipatory and hangover delays in the activity detector. If buffering is available at the transmitter to remember the last few tens of milliseconds of a silent interval, switching thresholds can be set to operate on relatively strong vowel energy. When the threshold is crossed coming out of silence, the information in the buffer is included in the talkspurt, permitting the detector to "anticipate" the switching action and include a possible weak initial consonant in the talkspurt. As little as four milliseconds are sufficient for this purpose.[8] The buffering required to accumulate a packetsworth of speech in a packet system would often be sufficient.

At the end of a talkspurt, a 200-300 msec hangover delay is often used to extend the talkspurt to include final weak consonants as well as to prevent unnecessary switching in short pauses, many of which occur in normal conversations. In at least one TASI system,[9] the hangover delay was adjusted depending upon the peak amplitude of the last syllable considered to be above threshold in the talkspurt. If the peak was low the delay was increased to give better performance for speakers who tend to trail off at the end of utterances.

The other type of failure that can cause loss of speech information occurs if the speech activity threshold is set too high in an attempt to exclude noise. A similar situation results if system gain, prior to the threshold stage, falls due to equipment malfunction. This failure can cause loss of whole words or phrases. This system problem can be handled by proper design and maintenance procedures with perhaps some help from adaptive threshold adjustments. It is not a sort of anomaly particularly related to packet speech systems or amenable to the techniques used in this study.

Failures of a speech activity detector in the opposite direction – accepting noise as speech – would appear to cause anomalies no more damaging to speech quality than the effects of noise in the absence of speech activity detection. Of course, if noise is accepted as speech, a packet network would experience a heavier load of speech packets than would be observed in the absence of the noise. In addition, there is a more subtle anomaly effect that may have a greater effect on speech quality than one would expect. When a speech detector switches on a noise burst during a nominally silent interval, the listener is likely to hear a perceptually greater effect than the noise burst would have had if transmission had been continuous through the interval. This effect results from the difference in the sound of the artificial "fill silence" synthesized in the absence of any transmitted signal and the real background "silence" that surrounded the noise burst. We noted this effect in informal listening tests, but have not yet subjected it to formal quality rating tests. We suspect that it may be a good idea to adjust the amplitude and perhaps the spectral shape of the "fill silence" to approximate the background noise at the transmitter. By so doing the difference between the fill silence and noise burst just above threshold would be minimized.

IV. ANOMALY SIMULATION PROGRAMS

To generate the anomalies discussed in Section III in a controlled way, a family of simulation programs were written. The programs run in part in LDVT (Lincoln Digital Voice Terminal) processors and in part in our DEC PDP-11/45 computer under the DEC RSX-11M operating system. The LDVTs handle the speech encoding and decoding algorithms; the PDP-11 packetizes the encoded bit streams and simulates the network delay effects. Speech activity detection is done in the LDVTs or PDP-11 depending on the encoding algorithm in use. The programs were developed in modular fashion. Programs to handle three different speech encoding algorithms were integrated with five different network simulation programs.

The general form of all simulation programs is the same (Fig. IV-1). For simple simulations to show the effect of lost packets, buffer underflow, etc., the simulation program is realized in the half-duplex form (Fig. IV-1). The same LDVT processor handles both speech encoding and decoding and the network delay and reconstitution algorithms are arbitrarily set to

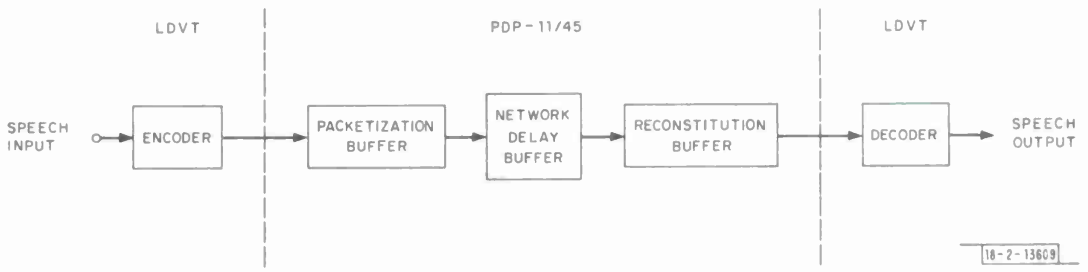


Fig. IV-1. Data flow for a half-duplex network simulation.

produce the desired type of anomaly. For simulation of a real packet speech communication system, a full-duplex simulation is obtained by repeating the configuration (Fig. IV-1) using two LDVTs to handle the two speech streams. Then the PDP-11 handles simulation of both network paths, but delays and reconstitution algorithms are completely independent.

All of our speech encoding simulations use a framed coding scheme. Framing is essential to the interpretation of LPC and adaptive predictive coding (APC) bit streams, but has no significance in CVSD encoding. We use the concept uniformly because the time base for the PDP-11 part of the simulation is provided by interrupts from the LDVT speech encoders and it is convenient to use interrupt rates on the order of one every 20 milliseconds, the natural frame rate of the LPC and APC encoders. Faster interrupt rates, compatible with the CVSD algorithm, would tend to overload the PDP-11 with no substantial advantage for the network simulations.

The once-per-frame interrupt from the LDVT provides all timing for the simulations. A clock word is incremented on each interrupt and the value of the clock word provides a time stamp for the data bits in the frame. The word "parcel" refers to the data bits in the frame. In our simulations, packets always contain an integral number of parcels. Following the technique used in the ARPANET packet speech experiments, each transmitted packet is given a time stamp equal to that of the first parcel in the packet. The time stamps of the other parcels are not transmitted.

Whether the speech activity detector algorithm indicates speech or silence all parcels are transferred from the LDVT to the PDP-11 and stored in the packetization buffer. If silence is

indicated for some time no packets are transmitted. When speech activity is first detected following such a period the last n parcels, where n corresponds to the anticipatory delay in effect for the speech activity detector, are marked as ready for transmission. Depending on the relation between n and the packet size in effect for the network simulation, one or more packets can be handed to the network simulation for transmission.

Except for the startup of a talkspurt when more than one packet might be transmitted in rapid succession, packets are sent at a uniform rate of one every k parcels where k is a run-time parameter for the simulation. When speech energy falls below threshold, a hangover delay of m parcel times is entered and packets continue to be transmitted until either the delay times out or the threshold is recrossed indicating the resumption of speech activity. The final packet of a talkspurt may contain less than k parcels.

When a packet is placed in the network simulation buffer it is given a network delivery time value that is computed in parcel time units according to the network delay or anomaly simulation algorithm in effect. When the input clock reaches the delivery time stored with the packet, the packet is moved from the network simulation buffer to the reconstitution buffer. According to the reconstitution algorithm in effect, it is then either played out or part or all of it discarded according to the value of its time stamp in relation to a playout clock maintained by the reconstitution algorithm. The network simulation and reconstitution algorithms and their interactions vary with the simulated conditions.

When the simulation programs require a random value for a network delay or the time to the next occurrence of an anomaly event, a subroutine capable of producing random variables with an arbitrary distribution is used. The subroutine uses the linear congruential method[10] to obtain a random number with a uniform distribution. This number is, in turn, used to obtain a number with the desired distribution by referring to a table whose values represent the desired cumulative distribution function. The tables in our experiments had 100 values and used linear interpolation where the table would yield values that differed by more than one parcel time. The tables contain normalized values that are multipled by run-time parameters to yield the actual values used.

The three types of speech encoding algorithms used, their associated speech activity detectors, the four kinds of specific anomaly simulations, and the complete network simulation developed for this study are described next. The variables that may be set at run time to produce a particular rate of anomaly generation, delay characteristic, etc., are indicated.

A. SPEECH ENCODING ALGORITHMS

Our simulations make use of three encoding techniques: LPC at 3.5 kbps, CVSD at 16 kbps, and APC at 15 kbps. We had planned initially to include PCM at 64 kbps, but preliminary calculations indicated that we would have difficulty achieving a full-duplex network simulation at the higher rate without the addition of new input/output hardware to the PDP-11/45. The required hardware is now being installed for another project and higher data rate simulations will soon be possible.

All three encoding algorithms use frame times on the order of 20 milliseconds, but exact frame times differ as do the techniques used for speech activity detection and the filling of "silent" gaps. The following sections discuss these aspects of the algorithm in more detail. We assume that the reader is familiar with encoding techniques per se and do not describe them here.

1. LPC Encoding

The LPC vocoder algorithm used was proposed originally by J. Markel for the ARPANET packet speech experiments.[11] It uses a sampling rate of 150 microseconds produces a 67-bit parcel every 19.2 milliseconds resulting in a 3489.58-bps bit rate. LPC is the lowest bit rate encoding technique used in this study.

Speech activity detection for LPC encoding is obtained by examining the 5-bit coded energy parameter in the PDP-11. The activity threshold as well as anticipatory and hangover delays are run-time parameters.

Gaps are filled in LPC by replaying the last received frame after switching the excitation to unvoiced (noise) and dividing the coded energy parameter by two with each repetition. The result is a signal that decays to silence within a few frame times without changing the spectral character of the output signal. This technique is quite satisfactory for dealing with gaps, both real and anomalous.

2. CVSD Encoding

The CVSD modulation technique involves transmitting a one-bit-per-sample encoding of the difference between the input waveform and an estimate formed by integrating the product of the transmitted difference function and a slope factor that is adjusted continuously on the basis of the three most recently generated difference values. The slope factor is obtained by smoothing a step function that has one of two values, V_{MAX} or V_{MIN}, at each sample time. When the last three difference values are either all ones or all zeros, indicating that the estimator is failing to keep up with the input, V_{MAX} is used as input to the smoother. Otherwise, V_{MIN} is used. For our 16-kbps encoder, the smoother uses a time constant of 5.6 milliseconds. At a sampling rate of 62.5 microseconds, our CVSD encoder produces a parcel of 320 bits every 20 milliseconds.

Speech activity detection for CVSD is based on the same 3-bit sequence used to adjust the slope factor. True silence at the input results in a sequence of alternating ones and zeros. A small amount of background noise causes some other pattern sequences to occur occasionally, but alternating ones and zeros should predominate. The speech activity detector maintains a running score of occurrences of "010" and "101" patterns in the three most recently generated output bits. For each occurrence it adds 2 to the score. When "000" or " 111" patterns are seen, it subtracts 12 from the score. Other patterns have no effect. The score value is clamped at zero and 600. Whenever the score is 470 or less the detector indicates the presence of speech. Both computation and thresholding take place in the LDVT. Anticipatory and hangover delays are run-time parameters handled by the PDP-11.

In a normal CVSD communication situation, the transmitter and receiver follow the same algorithm to compute the slope factor and estimate the waveform being encoded. In a packet speech system during natural pauses or anomaly gaps, the receiver cannot follow the transmitter exactly because the transmitted bit stream is not available. In our experiments, the decoder is given an alternating sequence of ones and zeros in the absence of a received bit stream. This causes the output waveform to go to zero at a rate determined by the time constant of the estimating integrator. The result is an abrupt transition to silence during anomalies and a quieter background during natural pauses than would be heard if speech activity detection were not in use.

We feel now that a better technique for filling silence gaps for CVSD would be to use a sequence that results in a random noise for output rather than silence. Ideally, the slope factor

adjustment algorithm would be set to produce a noise of average amplitude a little bit less than a value that would cause the speech activity detector to indicate speech. The contrast between the quiet of silence intervals and this relatively noisy character of CVSD speech would then be reduced.

3. APC Encoding

Adaptive Predictive Coding[12] is an algorithm that combines a pitch prediction with a low-order linear prediction and corrects the result by encoding and transmitting the residual error signal. In our implementation, the sampling time is 150 microseconds. A pitch prediction plus a fourth-order linear prediction is corrected with an error signal coded with two-bits per sample. A parcel of 368 bits is generated every 24.6 microseconds. The resultant bit stream requires 14959.35 bits per second, which is a little less than the 16 kbps used by the CVSD encoder, but quality of the output speech is superior, appoaching that of 50-kbps PCM. APC is the highest quality encoding technique used in our experiments.

Unlike LPC, APC encoding does not include an explicit energy parameter, and while the energy can be derived from the error signal, we chose to base speech activity detection on the amplitude of the speech signal at the input to the encoder. Measurement of the signal is handled by the LDVT with thresholding in the PDP-11. As in the LPC case, the activity threshold, and anticipatory and hangover delays are run-time parameters.

Handling gaps and natural pauses in APC, the PDP-11 continues to send the last real frame of data to the LDVT, but it sets the value of an extra word to indicate that a gap has occurred. The LDVT, on seeing that a gap is indicated, sets the excitation during synthesis to zero, thereby causing the output to become zero after any energy left over from the last real frame has died away. The result is a more rapid decay to silence than occurs via LPC encoding. Because APC is a much smoother-sounding encoding technique than CVSD, the transitions in and out of silence are less disturbing.

B. DELAY ANOMALY SIMULATIONS

Common to all our anomaly simulations is the algorithm to determine when the next anomaly event should occur. This algorithm uses the random-number generator with its table set to produce a Poisson distribution with a mean value of one. A run-time parameter sets the desired mean value for the experiment in number of packet times. Thus, if the experimenter enters the value 100, the program produces anomalies whose separations in time have a Poisson distribution with an average value of 100 packets times. The mean time between events then depends on the packet length in parcels (another run-time parameter) and the parcel time (a quantity fixed by the encoding technique). For example, with the anomaly interval set at 100 packet times and the packet length set at 2 parcels and using APC encoding, the mean time between events would be $100 \times 2 \times 24.6$ microseconds = 4.92 seconds.

In these anomaly simulations, packets are generated only when speech is present (as detected by the speech activity detector in effect for the simulation). Thus, the real time between events is longer than the nominal time by the duration of any ensuing pauses.

Another common feature of all the anomaly simulations is the use of a gated sine wave signal to mark the time location of each anomaly. By recording the resulting tone bursts on a second tape track, the experimenter can obtain a confirmation of the existence of an intentional delay anomaly as opposed to one that might have resulted from the action of a speech activity

detector. In addition, the tone bursts often serve to point out anomalies that might not be heard on first listening to the speech output of the simulation. For lost packet and underflow anomalies the tone bursts are set equal in duration to the gap in the output speech. For overflow anomalies, that cause no gap in the speech, the burst duration is arbitrarily set at 12 frame times.

Programs that generate anomaly events at controlled rates all use the general network simulation program described above. They differ, however, in the technique used to produce the particular kind of anomaly required in each case. The differences are:

1. Lost Packets

To produce lost packet anomalies, the anomaly generator is placed between the packetization buffer and network buffer (see Fig. IV-1). At program initialization, an integer, n, is obtained from the random-number generator. As packets are generated they are counted, and when the count reaches n, the n^{th} packet is discarded. A new value for n is then obtained from the random-number generator and the process repeated.

For this case, the simulated network delay is kept constant and the reconstitution delay is set to zero. The result is that packets are played out as they arrive, and a gap occurs in the output speech of a duration equal to the length of the discarded packet. To control gap size, the experimenter sets the packet length to the number of parcel times that approximate the desired length.

2. Underflow

To produce underflow anomalies the anomaly generator is placed between the packetization buffer and network buffer. As in the lost packet case, the decision to generate an anomaly is based on the packet count equaling a random variable. In this case, however, the anomaly is created by adding an increment to the network delay. The increment is obtained from a run-time parameter and is equal to the desired duration of the underflow anomaly. As in the lost packet case, the reconstitution delay is kept equal to zero so that packets are played out when received.

Since each underflow anomaly increases the delay in the simulated network, the quantity of speech held in the network buffer would eventually exceed the capacity of the buffer. To avoid this difficulty the network delay is reduced during natural pauses, taking care to prevent the delay becoming negative as a result of attempting to recover a delay increase greater than the length of the pause. Any unrecovered delay is held for recovery in future pauses. Even though this process might reduce a natural pause to zero length, there would still be a perceptible gap in the output speech because of the action of the hangover and anticipatory delays in the speech activity detectors.

If the decision to create an underflow anomaly occurs on the first packet of a talkspurt, the underflow is not created since the resulting increase in length in the natural pause is unlikely to be detected by a listener.

3. Overflow

Overflows are simulated by operations that take place on speech that has accumulated in the reconstitution buffer. During natural pauses the reconstitution delay is set to a large value to cause packets to accumulate in the buffer prior to playout. As playout proceeds, a packet count decides when to cause an anomaly in a fashion similar to that used for lost packet simulations. There are two types of overflow anomalies that can be generated depending upon the settings of

run-time parameters. The first calls for throwing out the next m contiguous vocoder frames. The second calls for throwing out m out of the next n frames spreading m over n as smoothly as possible. In either case, it may not be possible to generate the anomaly at the indicated decision point because there may not be the required m or n frames still to be played out in the current talkspurt. Rather than wait until enough packets are available, the program aborts the decision and requests a new random integer. This procedure avoids creating a disproportionate number of overflow events at the start of talkspurts.

When m frames are thrown out to create an underflow event, the reconstitution delay is reduced by m to prevent the introduction of gaps to replace the thrown out frames. The reconstitution delay is restored to its initial large value in an ensuing natural pause.

4. Combined Underflow Plus Overflow

Because the underflow and overflow mechanisms are realized in different parts of the general simulation program their combination is straightforward. For this simulation, underflow and overflow events alternate. Independent run-time parameters set the mean times to the occurrence of the next event of each type. The packet count to decide the first underflow starts at the beginning of the experimental run. Subsequent underflow counts start at the completion of each overflow event, and vice versa.

If underflow or overflow fail due to the constraints described for these individual events, they will try again with a new random number and persist until the anomaly in question is generated successfully. At that point, the count starts for the other type of anomaly.

V. FULL-DUPLEX NETWORK SIMULATION PROGRAM

Our network simulations make use of the general simulation program structure (Fig. IV-1) with a complete structure for each speech path in a full-duplex conversation. Here are detailed explanations on the network delay simulations available, the adaptive reconstitution algorithm used in our experiments, and the instrumentation available to measure the behavior of the simulated systems.

A. NETWORK DELAY SIMULATIONS

We experimented with two different delay simulations. One uses an overall delay distribution taken from our experiments with the ARPANET. The second was suggested by DCEC as a model for a three-node path in a hypothetical network that uses fixed-size packets and has Poisson packet-arrival statistics for each link. The ARPANET simulation is used principally as a check on simulator adequacy. Its characteristics are not discussed.

For the hypothetical network, we assume that each link is an independent M/D/1 queuing system with an arbitrary line speed and traffic intensity. This model corresponds to a link in a network where all packets are of equal length and arrive at the node with Poisson statistics. Prabhu[13] shows an expression for the waiting time distribution function for such a system that involves a summation of exponentials. Evaluation of the expression for different values of traffic intensity, ρ, yields plots that are essentially straight lines on semi-log paper (Fig. V-1). They can thus be approximated by simple exponential distribution functions. For purposes of the simulation program, it is convenient to use a single normalized exponential as a shaping function for the random-number generator and multiply the output of the table lookup by a run-time parameter representing the traffic intensity. We call this parameter the loading factor (LF). The relationship between LF and ρ obtained from Fig. V-1 is plotted in Fig. V-2.

In addition to the LF, the experimenter must provide another run-time parameter so that the service time for each link in the simulated network can be computed. The service time is defined as the total packet length in bits divided by the data rate of the link. The packet length is determined by the type of speech encoding in use, the number of parcels in the packet, and an overhead quantity we arbitrarily set at 150 bits. The link data rate is a run-time parameter that can be set independently for each of the four links in the simulated network. The LF can similarly be set independently for the four links.

B. THE RECONSTITUTION ALGORITHM

An ideal reconstitution algorithm delays the process of playing out the received speech packets just enough to make it highly probable that each packet arrives when needed. If the time stamps on arriving packets are based on a global time base known to both transmitter and receiver, and if the delay characteristics of the network were known a priori, the task of the reconstitution algorithm would be trivial. On the other hand, if there is no global time base, and network delay characteristics can vary, a more complex algorithm is needed to produce smooth output speech without unnecessary delay.

The reconstitution algorithm used in our simulations is designed to operate in a network where transmitter and receiver clocks run at almost the same rates, but differ by some arbitrary and unknown offset value. By "almost the same rate" we mean that drift between the

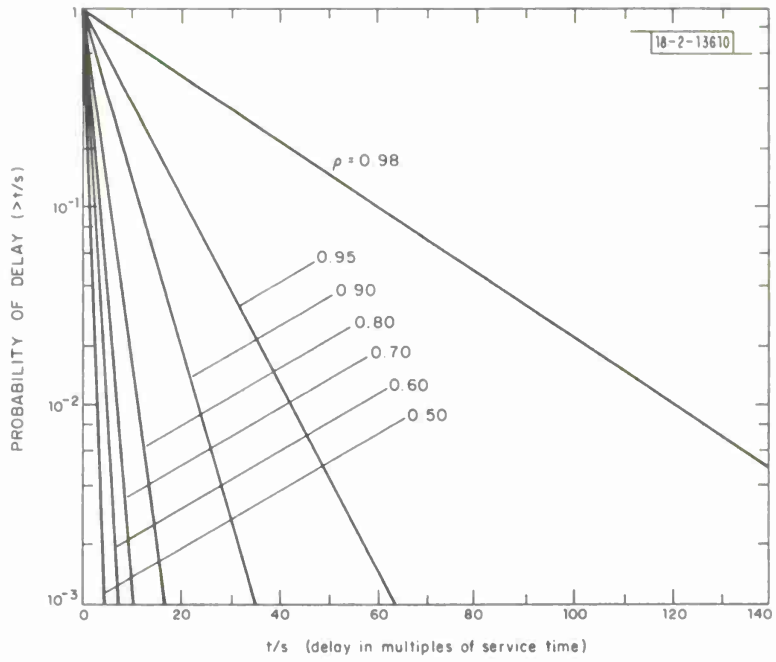


Fig. V-1. Waiting time distributions for a single link in the full-duplex network simulation plotted for various values of the traffic intensity, ρ.

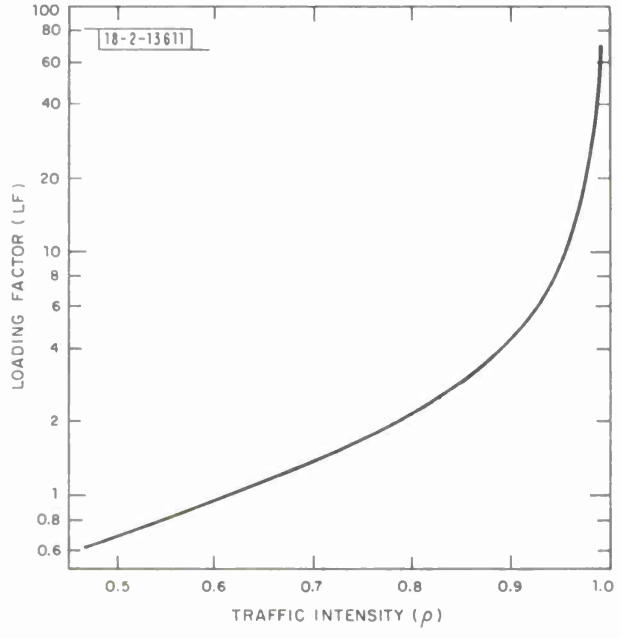


Fig. V-2. Relationship between the traffic intensity, ρ, and LF used to set the effective value of ρ in a simulation run.

clocks should not produce anomalies in the speech more often than once an hour or so. The unknown clock offset means that the reconstitution algorithm has no knowledge of the actual network transit time.

At the start of a conversation, the reconstitution algorithm can do little better than wait an arbitrary time (the reconstitution delay) after receipt of the first speech packet before starting to play it out. If it has some a priori knowledge of average network delays, it can make a reasonable guess at the required delay value. If the first packet experienced average or greater delay in passing through the net, and the reconstitution delay is right for the average case, there should be high probability that the first talkspurt is played out without anomalies or excessive delay. However, if the first packet experienced a delay close to the minimum, anomalies are likely unless the reconstitution delay is long enough to span the distance between minimum and maximum (almost) delays. The use of this more conservative reconstitution delay results in unnecessarily long delays for most talkspurts.

If the technique of delaying a fixed time from the arrival of the first packet is used on each talkspurt, unnecessary talkspurt-to-talkspurt jitter occurs in output speech. When the silences between talkspurts are long this jitter is of no consequence, but when silences are short (one second or less) it can result in unnatural sounding shifts in apparent speech timing.

In an attempt to minimize both overall delay and talkspurt-to-talkspurt jitter, our algorithm estimates the median packet arrival time and applies the reconstitution delay to this time rather than the actual arrival time of the first packet of each talkspurt after the first. When the first packet of the first talkspurt arrives, its time stamp value sets a clock called the ARRIVAL CLOCK. The clock value is incremented by one each time the speech encoder produces a new frame of data. As succeeding packets arrive, their time stamps are compared with the ARRIVAL CLOCK. If the time stamp is greater than the clock, the clock is advanced by one count. If the time stamp is less than the clock, the clock is retarded by one count. If the time stamp equals the clock, no change is made. The effect of this procedure is to cause the ARRIVAL CLOCK to follow the median packet arrival time. We use the median rather than mean for two reasons; (1) it is simpler to compute, and (2) the median is less sensitive to extreme values that are expected to occur occasionally, but to which we do not wish to respond in a linear fashion, because we do not want to make a change in speech timing as a result of receiving an isolated very late packet.

Actually, the time stamp of the arriving packet is not used directly to control the ARRIVAL CLOCK. If all packets are of uniform length the time stamp can be used directly, but since the final packet in a talkspurt may be shorter than the nominal length, the algorithm must take packet length into account. In setting and making comparisons with the ARRIVAL CLOCK, the length of the packet (in parcels) is added to the time stamp before any action is taken. This action results in a time value that corresponds to the clock value at the time the packet is prepared for delivery to the network. If the network has characteristics like the ARPANET, which deliver shorter packets with less delay than longer ones (as much as 80-msec difference between minimum and maximum), a further correction to the time stamp value is desirable. For example, in ARPANET experiments between Lincoln Laboratory and the Information Sciences Institute in California, we used a table that had values between ±2 to correct the arriving time stamps for this effect. A simpler solution to this problem would be to ignore the relatively few short packets insofar as they affect the ARRIVAL CLOCK. For our M/D/1 hypothetical network, no correction for length was made because the network simulator did not cause packets to be delayed in a length-dependent fashion.

The playout of speech packets is controlled by another clock called the PLAYOUT CLOCK. This clock is set using the value of the ARRIVAL CLOCK less the value of reconstitution delay. Setting of the PLAYOUT CLOCK occurs whenever a packet arrives that has a time stamp greater than that of the last packet played out and when there is no speech in the buffer that has still to be played out. This situation corresponds to the arrival of the first packet of a talkspurt following a silence that is long enough for the playout process to catch up with the input packet stream. The PLAYOUT CLOCK is incremented once for each encoder frame time. When the clock value equals the time stamp corresponding to a parcel, that parcel is played out. In the process of locating a parcel with a matching time stamp, any parcels with time stamps less than the PLAYOUT CLOCK value are discarded.

The use of the ARRIVAL CLOCK to estimate median packet arrival times provides some smoothing of talkspurt-to-talkspurt jitter and permits use of a reconstitution delay value somewhat smaller than would be possible if playout was based on the arrival of the first parcel only. In addition, the median estimator provides some capability for adapting to changing delay characteristics. However, because the dispersion of network delay is likely to vary over a wider range than the median delay, it is desirable to provide a facility to adjust the reconstitution delay. In our reconstitution algorithm the delay is adjusted dynamically by noting the occurrence of late packet arrivals. Whenever all or part of a packet arrives that is too late to be used in the playout process, the reconstitution delay is increased by one encoder frame time. Whenever a sequence of 100 packets is played out without any parcels being discarded, the reconstitution delay is decreased by one encoder frame time. The result is an algorithm that adapts rather rapidly to increasing network delays, but is slow to respond to decreasing delays. When network delays are stable, the algorithm causes a low rate of lost packet anomalies. These are not disturbing, but they could be reduced further by a slightly more complex algorithm that would halt the decrease in reconstitution delay at a point where further decrease would cause anomalies at some specified rate.

The buffer space available to the reconstitution algorithm can be set by a run-time parameter to explore the relationship between network delay characteristics and buffer size. If the buffer size is set too low, overflow anomalies occur. If the buffer size is not large enough to accommodate the needs of the reconstitution delay process, the output-speech quality is unacceptable with a very high rate of overflow anomalies. If buffer size is just slightly larger than the minimum required for reconstitution, very few underflow anomalies occur with stable network delay characteristics. We have not attempted in our reconstitution algorithm to adaptively adjust the reconstitution delay for some compromise between lost packet and underflow anomalies forced by an inadequate buffer capacity.

The adaptive features of our reconstitution algorithm have value only when the dispersion of network delays is several times larger than the speech encoder frame time (≈20 msec). For smaller dispersions, a fixed reconstitution delay of one or two frame times is satisfactory. Minimization of overall delay does not result in any significant advantage since the improvement expected is not noticed by human users of the system.

C. INSTRUMENTATION

As the network simulation runs, data are accumulated in PDP-11 memory on the delay characteristics of the simulated network paths and the behavior of speech activity detectors and

reconstitution algorithms. The statistics are gathered separately for each channel of the full-duplex simulation. On completion of a run, the data can be saved for analysis later. A program is available that produces a printout of the data in a mixed tabular and graphical form. Included in the accumulated statistics are:

1. Complete histogram of network delays.
2. Histograms showing behavior of the ARRIVAL and PLAYOUT clocks in the reconstitution algorithm.
3. Histogram showing the quantity of speech in the reconstitution buffer after arrival of each packet.
4. Histograms of talkspurt and silence durations.
5. Histogram of input signal amplitude during periods considered to be silent by the speech activity detector.
6. Counts of total packets received, packets out-of-order, packets too late to be used, total parcels received, number of times parcels are discarded during reconstitution, and total parcels discarded.
7. Run-time parameters for the simulation.

VI. RESULTS AND CONCLUSIONS

A goal of this study has been to conduct formal listening tests on material containing controlled rates and types of packet speech anomalies. Because of the intermittent nature of these anomalies conventional intelligibility tests were not considered appropriate. Our earlier experience indicated that the main effect of anomalies at modest rates was a reduction in speech quality. We anticipated that as rates increased, quality effects would lead to packet speech systems being judged unacceptable before intelligibility was reduced to a point where communicability suffered seriously. Accordingly, we planned to use the Diagnostic Acceptability Measure[14] (DAM) procedures developed by Dynastat, Inc. that are aimed at assessing overall acceptability of speech communication systems. Since Dynastat personnel had had no experience with temporal distortions such as are caused by packet anomalies, they requested that we make a preliminary test tape with which they could work prior to the beginning of formal tests. They concluded that their procedures had to be extended in order to handle packet speech anomalies. Appropriate new DAM tests were developed, but the several months required to extend the procedures has delayed this report and limited reportable results to that obtained from the preliminary test tape. That tape had eleven different anomaly conditions all making use of CVSD speech encoding. We plan to continue formal testing, explore other encoding techniques and anomaly conditions, and report the results in a journal article.

In the following sections we report the results from the DAM evaluation of our CVSD test tape and summarize our conclusion and suggestions for further work.

A. DAM EVALUATION RESULTS

The Diagnostic Acceptability Measure has grown out of experience at Dynastat with acceptability evaluation work undertaken for the Defense Communication Agency. The results of the effort included the Paired Acceptability Rating Method (PARM) and the Quality Acceptance Rating Test (QUART). The DAM procedure combines direct and indirect approaches to acceptability evaluation by a 20-item system rating form. The rating data are analyzed to yield system diagnoses with respect to selected perceptual qualities (Table 1). Results for each test include diagnostic scores for the basic signal and background qualities plus direct ratings of intelligibility, pleasantness, and acceptability. In addition, indirect estimates of acceptability are derived from listener ratings of various other characteristics. Each diagnostic score for signal and background qualities represents the acceptability rating a system would have if it were deficient only with respect to the corresponding perceptual quality.

The overall scores for intelligibility, pleasantness, and acceptability are difficult to interpret as absolute numbers. They are meaningful, however, when compared across systems. To aid in the interpretation of overall results, Dynastat now includes a Predicted Percent User Acceptance value that attempts to effect a transformation from overall test scores to experience with real systems. This overall index of acceptability seems to be well suited for our purposes in comparing different anomaly effects. We have chosen to present it as well as the basic signal and background quality scores in this report and avoid discussion of the other overall measures that vary over a narrower range.

The test material for the DAM evaluation consists of twelve phonemically controlled sentences spoken by each of the desired number of speakers. Approximately one minute total running time is required for each speaker. In our tests, we used two speakers for each of eleven

TABLE 1

SYSTEM CHARACTERISTICS EVALUATED BY DAM

Diagnostic Scale	Typical Descriptor	Exemplar	Intrinsic Effect On Acceptability
SIGNAL QUALITIES			
SF	Fluttering	Interrupted or amplitude modulated speech	Moderate
SH	Thin	High-pass speech	Mild
SD	Rasping	Peak clipped speech	Severe
SL	Muffled	Low-pass speech	Mild
SI	Interrupted	Packetized speech with "glitches"	Moderate
SN	Nasal	2.4-kbps systems	Moderate
BACKGROUND QUALITIES			
BN	Hissing	Noise masked speech	Moderate
BB	Buzzing	Tandemmed digital systems	Moderate
BF	Babbling	Narrowband systems with errors	Severe
BR	Rumbling	Low frequency noise-masked speech	Moderate
TOTAL EFFECT			
	Scale		
	Intelligibility		
	Pleasantness		
	Acceptability		

different anomaly conditions; 16-kbps CVSD speech encoding was used throughout. The conditions included six different lost packet cases, two underflow cases, one overflow, and two combined underflow/overflow cases. The test results for the lost packet cases are shown in Table 2; results for the other conditions are presented in Table 3. The factors (SF, SH, etc.) correspond to the diagnostic scales of Table 1. A higher score indicates a more acceptable system. The scores shown are the mean values plus or minus the standard errors. No error estimates are available for the Predicted Percent User Acceptance values, but discussion with Dr. Voiers of Dynastat indicates that the values should be viewed as approximate.

Many of the diagnostic factors show no significant correlation with the parameters affecting the anomaly events. This behavior is expected since those factors are intended to reflect different aspects of the speech encoding technique in use and/or other system aspects intentionally held constant in these experiments. As expected, the factor SI (Interrupted) shows the largest correlation with anomaly conditions with SF (Fluttering) second. A small reverse correlation is observed with high rates (5%) of lost packet anomalies scoring slightly better on factor SL (Muffled) than lower rates.

Overall ratings are lower than expected for 16-kbps CVSD speech, principally because of low scores for factor SD (Rasping). We do not know whether these low scores result from some misadjustment of our CVSD encoder, from some problem in preparing the test tape, or from the use of speech activity detection that accentuates the difference in quality between silent intervals and speech intervals that tend to sound harsh with CVSD encoding. Further tests are needed to determine whether or not there is an effect due to speech activity detection. The effect of the SD factor scores in lowering overall scores is uniform across the different packet anomaly conditions and does not distort the effect of the anomalies on relative acceptability.

Fig. VI-1 shows the effects of packet size and anomaly rate for lost packet anomalies. Scores for the factors SI (Interrupted) and SF (Fluttering) both fall as packet loss increases [Fig. VI-1(a)]. The rate of fall for SI is greater for long packets (140-msec) while SF falls more rapidly for short packets (20-msec). The slight rise in SF (between 0.2 and 1 percent for both cases) and the rise in SI for short packets is probably due to noise in the evaluation process, which in this case, depends on a very small sample. There was only one lost packet in the 0.2-percent, 140-msec, test recording. It should be noted that anomalies occur seven times more frequently for 20-msec packets than for 140-msec packets at the same percentage loss. The average times between events for each case are shown in Tables 2 and 3. These time values count only periods in which speech was indicated by the speech activity detector. The total elapsed time between events was much greater because the test material had long gaps between sentences.

Figure VI-1(b) shows the effects of lost packet anomalies on the Predicted Percent User Acceptance. Again, the higher values for 1-percent packet loss are not significant. For long and short packets, there is a substantial drop in acceptability when packet loss rates reach 5 percent. The results for the short packet case indicate a little more acceptability at high rates. We feel from our own listening that the difference could be even greater if longer samples were used in the testing, because it is a matter of chance whether or not a long gap damages the meaning of a sentence. For this test situation, the listeners are familiar with the sentences and may not respond to an anomaly that would have caused an unknown sentence to be incorrectly understood.

The test results presented in Table 3 all correspond to an anomaly rate of 1 percent, i.e., an event occurs, on average, once in every 100 packets. For underflow cases, as expected, a larger anomaly produces a greater effect (see columns 1 and 2 of Table 3), but at the 1 percent

TABLE 2

DAM RESULTS FOR LOST PACKET ANOMALIES

Anomaly Type		Lost Packet					
Average Size (msec)		140			20		
Average Speech Lost (%)		0.2	1.0	5.0	0.2	1.0	5.0
Average Speech Time Between Events (sec)		70	14	2.8	10	2	0.4
Signal Quality Factors	SF	87.3 ± 0.7	87.2 ± 0.8	82.7 ± 1.5	85.1 ± 1.4	87.7 ± 0.9	78.9 ± 2.1
	SH	88.9 ± 0.6	87.8 ± 1.2	88.9 ± 0.3	88.9 ± 0.6	89.0 ± 0.3	87.7 ± 0.8
	SD	67.2 ± 1.5	67.3 ± 1.6	69.6 ± 1.8	66.0 ± 1.5	66.4 ± 1.6	70.2 ± 1.3
	SL	78.8 ± 1.3	79.4 ± 1.5	82.7 ± 0.8	80.8 ± 1.1	79.4 ± 1.6	82.4 ± 1.6
	SI	86.5 ± 2.9	78.6 ± 2.1	43.7 ± 4.2	83.0 ± 2.8	84.2 ± 2.3	75.8 ± 3.2
	SN	81.2 ± 2.1	84.2 ± 1.4	84.3 ± 2.0	83.8 ± 1.9	82.3 ± 1.9	81.5 ± 2.4
Background Quality Factors	BN	85.5 ± 0.9	84.4 ± 1.2	82.0 ± 0.7	85.7 ± 1.2	85.7 ± 1.0	83.9 ± 1.0
	BB	72.8 ± 2.7	79.1 ± 3.2	71.8 ± 2.3	71.7 ± 3.5	72.1 ± 1.5	75.9 ± 2.0
	BF	85.3 ± 0.9	85.3 ± 0.7	85.1 ± 0.6	85.1 ± 0.8	85.3 ± 0.7	82.8 ± 1.1
	BR	87.2 ± 0.2	87.6 ± 0.4	85.8 ± 1.1	86.7 ± 0.3	86.9 ± 0.2	85.0 ± 1.2
Predicted Percent User Acceptance		81.9	87.0	49.1	82.3	83.2	57.0

TABLE 3

DAM RESULTS FOR UNDERFLOW AND OVERFLOW ANOMALIES

Anomaly Type		Underflow		Overflow	Underflow Plus Overflow	
Anomaly Sizes (msec)		260	140	140	140	20
Average Speech Time Between Events (sec)		14	14	14	14	2
Signal Quality Factors	SF	86.1 ± 0.7	87.8 ± 0.7	85.9 ± 0.7	83.8 ± 1.3	85.0 ± 1.1
	SH	88.8 ± 0.4	89.2 ± 0.4	88.5 ± 0.6	88.4 ± 0.6	88.6 ± 0.6
	SD	69.6 ± 1.0	70.2 ± 1.2	67.6 ± 1.5	68.7 ± 1.8	68.0 ± 1.4
	SL	80.7 ± 1.1	83.4 ± 1.2	82.6 ± 1.0	81.4 ± 1.3	81.0 ± 1.2
	SI	50.5 ± 3.5	74.8 ± 2.2	57.1 ± 3.1	40.6 ± 2.7	83.2 ± 2.4
	SN	83.8 ± 1.5	86.3 ± 1.3	82.1 ± 2.0	84.2 ± 1.1	84.6 ± 1.4
Background Quality Factors	BN	81.9 ± 1.0	81.4 ± 1.0	83.2 ± 0.8	82.1 ± 1.0	84.6 ± 1.1
	BB	69.2 ± 2.1	70.8 ± 2.0	74.2 ± 2.1	71.0 ± 2.3	72.5 ± 2.7
	BF	84.9 ± 0.7	85.9 ± 0.4	84.7 ± 0.8	85.5 ± 0.4	84.5 ± 1.1
	BR	87.9 ± 0.4	87.9 ± 0.4	85.9 ± 0.9	86.8 ± 0.6	87.3 ± 0.2
Predicted Percent User Acceptance		75.0	81.8	53.0	39.2	81.5

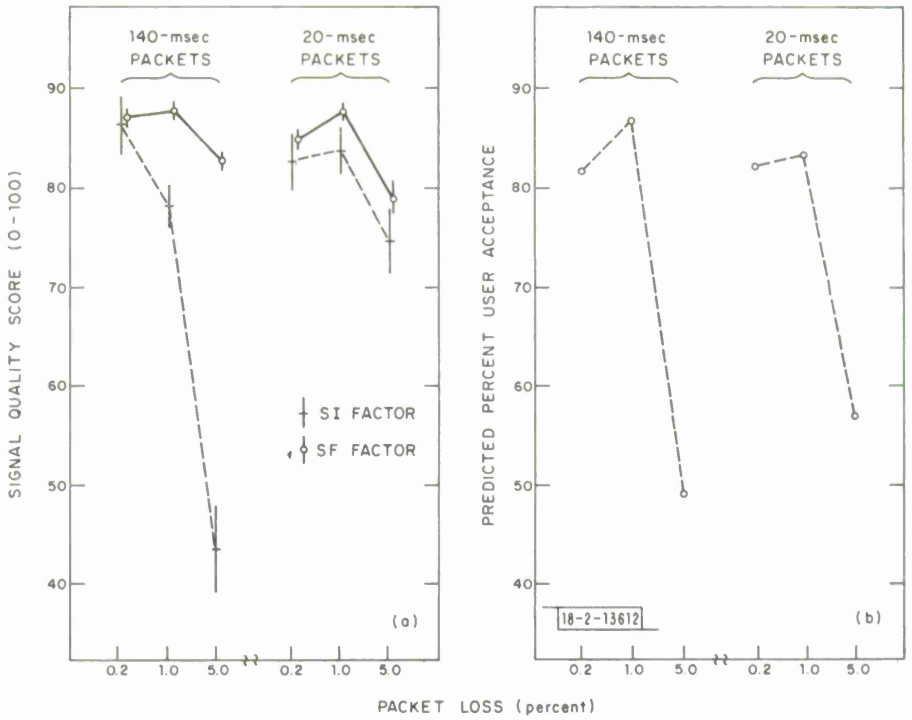


Fig. VI-1. DAM test results for six different lost packet anomaly cases; (a) the behavior of diagnostic factors SI and SF for two packet lengths and three rates of packet loss and (b) predicted percent acceptance for the same situations.

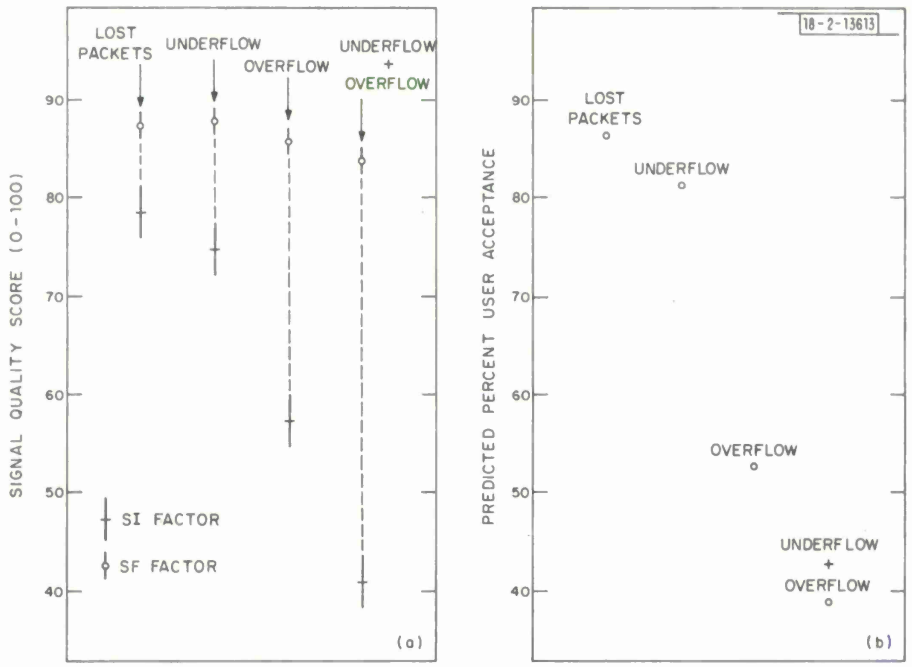


Fig. VI-2. DAM test results for four different types of anomalies occurring at the same rate (1%) and having the same magnitude (140 msec).

rate neither value reduces Predicted User Acceptance by a large amount. Overflow that loses 140 milliseconds of speech has a large effect, and combined underflow and overflow has an even greater effect when the speech loss is large. However, for the 20-msec case, combined underflow and overflow is not serious.

When the four different anomaly types occur at the same rate (1%) and have the same duration (140 msec), the lost packet case is the least damaging of the anomaly situation, and lost packets and underflows at 1 percent have an almost negligible effect on acceptability (Fig. VI-2). However, cases involving overflows cause a serious loss of acceptability.

B. SUMMARY OF CONCLUSIONS

(1) To be useful for speech communications a packet network must embody flow control procedures or priority mechanisms that can guarantee each speech subscriber an average data rate adequate to handle the encoding technique in use. If the available rate falls below that requirement, other than momentarily and infrequently, communicability suffers and the system is judged as unacceptable.

(2) Packet sizes should be kept small (50 msec or less of speech), if possible. The use of larger packets can result in occasional loss of whole syllables or words that can alter the meaning of the subscriber's speech in the event that a packet is lost.

(3) Listening tests indicate that lost packet and underflow anomalies can occur at rates on the order of 1 percent without significant loss of predicted user acceptability. This means that a designer can safely set parameters in the speech reconstitution algorithm to compromise values that result in anomalies at this rate or less.

(4) Overflow and combined underflow and overflow anomalies of duration greater than the order of 50 milliseconds should be avoided insofar as possible. A 1-percent rate of occurrence of such anomalies leads to significant loss of predicted user acceptability. This means that a system designer should use buffer sizes that prevent the occurrence of overflow anomalies under normal steady-state conditions. Occasional overflows may be unavoidable under an abnormal situation such as the loss of network node links. These should not be a problem if they do not occur too frequently.

(5) Simple energy thresholding with anticipatory and hangover delays is adequate to serve the needs for speech activity detection in packet networks so long as the talkers are in normally quiet environments such as offices. For speakers in noisy environments, more packets are given to the network for delivery than for the same speech spoken in quiet environments. As a result, network loads are higher. Further work is needed to determine if the switching activity of a speech detector in a noisy environment makes the background noise more or less objectionable to a listener in a quiet environment.

(6) Care should be used in filling gaps that occur when a successor packet fails to arrive when needed by the speech reconstitution algorithm at the receiver. The proper action to be taken depends on the speech encoding technique in use. The impact of anomalies can be reduced significantly by the choice of a procedure that is well matched to the encoding technique.

(7) Informal listening tests indicate no important interaction between the effect of anomalies and the speech encoding technique in use provided that gaps are filled properly. Formal testing has yet to be carried out to confirm this conclusion.

REFERENCES

1. Semiannual Technical Summary, Information Processing Techniques Program, Lincoln Laboratory, M.I.T. I, 3-7 (31 December 1974), DDC AD-A024999/5 and 2 (30 June 1975), DDC AD-A025000/1.
2. J. M. McQuillan, "Adaptive Routing Algorithm for Distributed Computer Networks," Report 2831, Bolt Beranek and Newman, Inc., Cambridge, Mass., 90 (May 1974).
3. G. J. Coviello, O. L. Lake, and G. R. Redinbo, "System Design Implications of Packetized Voice," ICC'77 Conf. Record, III, 49-53 (June 1977).
4. J. W. Forgie, "Speech Transmission in Packet-Switched Store and Forward Networks," National Computer Conf. 137-142 (1975).
5. J. W. Forgie and A. G. Nemeth, "An Efficient Packetized Voice/Data Network using Statistical Flow Control," ICC'77 Conf. III, 44-48 (June 1977).
6. K. Bullington and J. M. Fraser, "Engineering Aspects of TASI," Bell Syst. Tech. J. 38, 353-364 (1959).
7. R. J. McAulay, "A Robust Silence Detector for Increasing Network Channel Capacity," ICC'77 Conf. Record, III, 54-56 (June 1977).
8. J. A. Jankowski, Jr., "A New Digital Voice-Actuated Switch," COMSAT Tech. Rev. 6, 1, 159-178 (Spring 1976).
9. G. H. Leopold, "A System for Restoration and Expansion of Overseas Circuits," Bell Lab Record, 299-306 (November 1970).
10. D. E. Knuth, The Art of Computer Programming, 2, Chap. 3 (Addison-Wesley, Reading, Mass. 1969).
11. D. Cohen, "Specifications for the Network Voice Protocol," Report RR-75-39, Information Sciences Inst., Marina Del Rey, California (March 1976).
12. B. S. Atal and M. R. Schroeder, "Adaptive Predictive Coding of Speech Signals," Bell Syst. Tech. J. 49, 8, 1973-1986 (October 1970).
13. N. V. Prabhu, Queues and Inventories, 35 (Wiley, New York, 1965).
14. W. D. Voiers, "Diagnostic Acceptability Measure for Speech Communication Systems," 1977 IEEE Int. Conf. on Acoustics, Speech and Signal Processing, 204-207 (May 1977).

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER ESD-TR-77-178	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE *(and Subtitle)* Network Speech System Implications of Packetized Speech		5. TYPE OF REPORT & PERIOD COVERED Annual Report 1 January – 30 September 1976 6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) James W. Forgie		8. CONTRACT OR GRANT NUMBER(s) F19628-76-C-0002
9. PERFORMING ORGANIZATION NAME AND ADDRESS Lincoln Laboratory, M.I.T. P.O. Box 73 Lexington, MA 02173		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS Program Element No. 33126K
11. CONTROLLING OFFICE NAME AND ADDRESS Defense Communications Agency 8th Street & So. Courthouse Road Arlington, VA 22204		12. REPORT DATE 30 September 1976 13. NUMBER OF PAGES 38
14. MONITORING AGENCY NAME & ADDRESS *(if different from Controlling Office)* Electronic Systems Division Hanscom AFB Bedford, MA 01731		15. SECURITY CLASS. *(of this report)* Unclassified 15a. DECLASSIFICATION DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT *(of this Report)*

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT *(of the abstract entered in Block 20, if different from Report)*

18. SUPPLEMENTARY NOTES

None

19. KEY WORDS *(Continue on reverse side if necessary and identify by block number)*

packetized speech	packet-switched network	speech activity detection
packet network delays	packet speech systems	reconstruction algorithms
anomaly simulation program	full duplex network simulation	ARPANET

20. ABSTRACT *(Continue on reverse side if necessary and identify by block number)*

The effects, examined in parametric fashion, on the overall voice quality, acceptability, and communicability of speech packetization and its transmission through a packet-switched network. Speech processed through a number of real-time simulation programs developed to create anticipated anomalies (glitches) in packet speech systems were evaluated by informal acceptability testing. Depending on system design parameters, test results indicate that packet-system speech quality varies from essentially perfect (no packet-related anomalies) to unusable. Guidelines are provided for an acceptable packetized speech communication system.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE *(When Data Entered)*