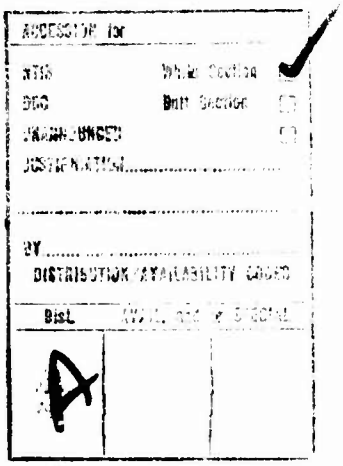
CHI

CULLER / HARRISON INC.

Computer Research and Development

CULLER/HARRISON, INC.
150-A Aero Camino
Goleta, California 93017
(805) 968-1064

March 1976



NETWORK VOICE CONFERENCING

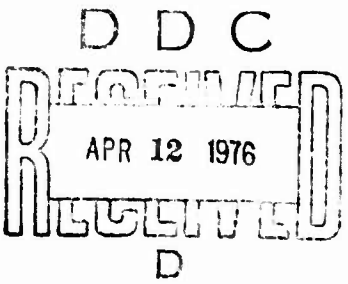
Quarterly Technical Report

Speech Compression Research at CHI

December 1975 - February 1976

PRINCIPAL INVESTIGATOR: Dr. Glen J. Culler

PROJECT SCIENTIST: Dr. Michael McCammon



This research was supported by the Defense Advanced Research Projects Agency under ARPA Order No. 2359/4 Contract No. DAHC15 73 C 0252

CONTENTS

I. INTRODUCTION AND SUMMARY

During the period covered by this report, December 1975 through February 1976, Culler/Harrison, Inc. has begun the development of an experimental real time voice conferencing system on the ARPANET. This conferencing system is a continuation of our previous work and uses linear predictive coding (LPC) to achieve effective data rates as low as 1000 bits per second for digital transmission of speech information. The development of this conference system is a joint effort with other ARPA contractors, principally MIT Lincoln Laboratories and USC Information Sciences Institute. The objectives of our participation in this development are to establish a voice conferencing facility on the ARPANET and to use this experimental facility to develop improved procedures for transmitting and receiving packet speech data in a conference environment.

The Culler/Harrison implementation of voice conferencing makes use of the CHI Signal Processing system developed under ARPA support. Within the structure of the protocols for communication and control between host sites developed for network voice conferencing [3], we are implementing a system supporting one LPC vocoder to translate between digitally sampled speech and the parameters used for digital transmission on the ARPANET. This system will support up to four local participants in a single conference sharing the vocoder with appropriate switching of audio signals between the participants. Current plans provide for up to six other sites and sixteen nonlocal participants. Conference conversation is from one speaker to all other participants, with speaker selection and conference participation determined by a conference chairman located at one site. The introduction to Chapter III gives further details of the conference organization.

The majority of our efforts up to this point have been in preliminary planning and implementation of the initial version of the network voice conferencing system. This system is an extension of the network voice system which provided continuous voice communication between two speakers using LPC for speech compression. The extension provides a set of control messages for selecting the speaker and data message transmission and receiving programs which can dynamically respond to this control. A separate chairman program has also been developed to provide the control for a network conference. Both chairman and local conference control program provide for monitoring of significant events and recording of the data gathered for subsequent evaluation. This

information provides a record of some aspects of system performance and serves to aid in the development of improved protocols. The network voice conference system is described in Chapter III. Chapter IV describes the monitor procedures used and gives some examples of the use of the trace record.

Several experiments with network voice conferencing have already been completed. These experiments have involved from one to four sites and up to six participants. A demonstration of this preliminary network voice conference system was given in late January at Lincoln Labs with CHI, ISI and Stanford Research Institute participating. These experiments have demonstrated that such a conference system can function. Much additional development and experimentation are needed before it becomes an operational capability, particularly if large numbers of sites are to be involved.

During the next quarter we expect to conduct several more experiments in an attempt to increase the reliability and ease of use of the conference system. Audio switching equipment will be integrated to facilitate multiple local participants. Revised protocols for coding and transmission of the LPC speech parameters have now been specified and should permit reduction in the effective bit rate well below 1000 bps. Implementation of these new protocols should commence during this next quarter.

The development of a network voice conferencing system appears to be proceeding quite well, and we believe that there should be no difficulty in providing a usable facility by the end of this contract period. We hope that it will be possible to extend the conference capability to additional sites on the ARPANET, and plan to explore procedures for dealing with data routing problems in large conferences.

II. THE CULLER/HARRISON SIGNAL PROCESSING SYSTEM

The CHI signal processing system used for real time linear predictive coding has been described in detail in an earlier technical report [1]. Since that report was prepared, one significant modification in the hardware configuration has taken place. The analog-to-digital and digital-to-analog conversions are now performed using the multi-channel audio signal system developed at CHI and described in the same technical report.

The new audio signal system provides better sampling time resolution through an automatic counter capable of 250 nanosecond resolution. The former system required software updating of the timer after each sample period and had a resolution and repeatability of one microsecond. The new system is interfaced directly to the MP-32A processor which is used for all control functions and data buffering during the processing. This shortens the times from data input to analysis and from synthesis to data output. It places an additional load on the MP-32A processor, however, as the processor must service the analog system for each point input and output.

Almost all processing in the system is now performed in two processors. The MP-32A is the master computer, and in addition to the analog conversion service, it provides all input/output functions, data and message buffering, parameter formatting for analysis and synthesis and scheduling of the other processor. In addition, all nonvocoder functions of the network voice conference system are performed by this machine. The AP-90 is a high-speed fixed and floating point arithmetic unit which performs the actual analysis and synthesis computation. It communicates only with the MP-32A. Network messages pass through an additional processor which provides the support for most IMP/HOST protocols, including the reliable transmission protocol for the very distant host interface to the ARPANET.

III. NETWORK VOICE CONFERENCING

Previous work has demonstrated successful low bit rate digital voice communication between two participants over the ARPANET. These experiments utilized a full duplex communication path, with the output of each participant's analyzer connected logically to the other's synthesizer inputs. For conferences with three or more participants, it is not desirable for everyone to talk at once, and each participant can only synthesize one speaker's parameters at a time. An extension of the protocols used for previous network voice communication [2] has been adopted for network voice conferencing. This protocol is known as the Network Voice Conference Protocol (NVCP) [3].

The conference environment is essentially half duplex. Only one person is transmitting. All others receive his messages and synthesize speech from them. Therefore, continuing control procedures are required to allow for switching from one speaker to the next. In addition, since it is now possible for participants to join and leave an ongoing conference, the set of hosts receiving data may change. Hence, an expanded set of control procedures, with control communication between hosts, is needed. These control procedures are implemented as a conference chairman (CHAIR), located at one host, together with local conference controllers (LCC) at each host participating in the conference. Both the CHAIR and LCC are programs which may accept input from human participants to help control their functions. The LCC may provide control for several participants as well as one or more vocoders. In the current conference system at CHI we have one vocoder, but up to four local participants are permitted by our LCC.

The flow of data between participant sites is directed by the CHAIR, who selects which participant is to speak. When the speaker is to change, three types of control messages must be sent by the CHAIR. The first requests the old speaker (and his LCC) to stop speaking and sending data messages. The second control message is sent to all participant LCCs to identify the new speaker whose data they should accept. The third is sent to the speaker and his LCC to inform him that he may speak and to provide the list of hosts who should be sent copies of the speech data.

With these three control messages, the chairman can dynamically reconfigure the data transmission paths to allow any participant to speak to all others in the conference. The utilization of a chairman providing the control

with distributed switching of the data provides for shorter data transfer times than having all data sent to a central location for distribution to the listeners. It does require more time to change speakers, however, as control messages must reach all sites before switching is complete.

The control messages just described provide dynamic switching of the data paths in an established conference. To establish a conference, and to allow participants to join or leave an ongoing conference, additional control communication is required. Also, each host system involved in the conference must agree on the vocoder and message transmission parameters to be used. Since more than one participant may be located at a single host system, participant negotiation is separated from system parameter negotiation.

The system parameter negotiation is identical to that used in the network voice protocol. It is started automatically when the first "request to join a conference" message is received by the CHAIR from a given LCC. This negotiation establishes the vocoding and data transmission parameters to be used during the conference. Since negotiation is carried out separately with each host's LCC by the conference chairman, while data transmission takes place directly between all pairs of hosts, the parameters are in fact dictated by the chairman.

A separate negotiation procedure is followed for each participant which wishes to join a conference. This negotiation is limited at present to a request to join message identifying the participant to the chairman and an acceptance or rejection message returned by the chairman. The same rejection message may be used at any time by the chairman to remove a participant from the conference. The LCC provides the actual access control for the participants under the direction of the CHAIR.

Participants can communicate directly with the CHAIR, and vice versa, through control messages containing function codes. The primary use of these is to permit a participant to request a turn speaking or to enable the speaker to inform the CHAIR that he is through. These permit an automatic CHAIR program to have the information needed to schedule speakers on demand.

The control messages described up to this point, together with protocols for when they must be used, define the interface between hosts participating in a network voice conference. A list of the control messages defined at this time for network voice conferencing is given in Appendix A. The rest of this chapter will describe the implementation of the network voice conferencing system at CHI, including the interface between the system and participant users.

This system has several components in addition to the LPC vocoder. The local conference controller (LCC) must interact with the conference chairman through control messages over the network, control the transmitter and receiver for speech data, and interact with the user through voice and nonvoice means. The conference chairman (CHAIR) is used only when the conference is being chaired locally and operates separately from the LCC. It must also interact with a local user if manual chairman functions are being used. Finally, the system monitoring facilities, when enabled, collect data about events in the system for subsequent analysis.

A. The Local Conference Controller

The network voice system, which serves as the base on which the voice conferencing system is being developed, already provides for preparation of data messages from the parameters output by the LPC analysis and for the selection and decoding of input messages to obtain the parameters for LPC synthesis. These procedures, as well as the LPC analysis and synthesis, remain essentially unchanged in the conferencing system. With the exception of the selection of input messages for processing, these parts of the system were described in a previous technical report [1]. The input message selection will be described here.

1. Input Message Selection

As input messages are received, they are ordered by their time stamp, which represents the time at the transmitter when the first data in the message was processed. Since the time which a message takes to travel from its source to the receiver varies, a delay is introduced after the expected time for a message's arrival before it is used for synthesis. This delay defines the amount by which a message can be late without creating a gap in the synthesized output. The delay is measured from the expected arrival time, rather than the actual arrival time of any particular message, such as the first message after a silence, to minimize raggedness in the speech due to variations in the arrival time of any particular message. The establishment of a preferred time for processing of a particular message based on its time stamp allows continuation of synthesis at the proper time even when intervening messages are lost or very late.

The expected arrival time for a message is calculated by adding the expected network transit time to the time stamp of the message. The expected network transit time is updated each time a message is selected for processing by the exponential averaging formula:

$$NT' = NT + \frac{1}{16}(OT-NT)$$

where OT is the observed network transit time,

$$OT = \text{Time Received} - \text{Time Sent}$$

and the time the message was sent is assumed to be given by the message time stamp plus the time represented by the speech parcels in the message. A message is selected for processing by the synthesizer when

$$\text{Time} = \text{Time Stamp} + NT + D$$

where D is the delay parameter chosen to account for variations in the message length and short term variations in the actual network time. The expected network transit time, NT, incorporates the difference between the clock used by the sender to generate the time stamp and the receiver's clock. This permits NT to adjust for any variations in the clock frequency between hosts.

For the conference system the expected network transit time from each other host in the conference will be different. This separation could be maintained by saving the current NT value for the old speaker's host and using the value for the new speaker's host each time speakers are switched. However, if much time has passed since any data was received from the new host, the value for NT may no longer be usable. Instead, we reset the NT computation to its initial state each time a speaker switch occurs and any time that the observed transit time OT is much less than NT. In this state, NT is set to OT for the first message received. Although we presently keep D fixed when speaker switching occurs, it is possible to maintain different values for each host, depending on factors such as the distance to the host and the amount of speech data represented by each message.

2. Transmitter and Receiver Switching

The voice conference system differs from the network voice system in requiring selective enabling, or switching of data at both sides of each half of the vocoder. On the user side of the vocoder, the participant who is to speak, if any, must have his microphone enabled for input to the LPC analysis. The speaker's loudspeaker or earphone must be disabled from output of his own delayed, synthetic speech. The earphones of other local participants are enabled only if they have been accepted into the conference. Actual implementation of microphone and earphone switching will be completed during the next quarter. On the network side, vocoder output can be sent to other hosts only when a local participant is speaker. The list of hosts to send the data to is set each time a new speaker is selected. Input messages from the network, on the other hand, are accepted only if received from the host of the current speaker.

The LCC provides these switching and selection functions as a layer of programs around the LPC vocoder. The LPC analysis program runs continuously, processing new input data every 9.6 milliseconds and preparing data messages for transmission. The conference system message transmission program then decides whether to transmit the message to its list of hosts or to discard it. The LPC synthesis program operates whenever messages are available on its input queue for selection. The data message receiver decides whether incoming messages are to be placed on the queue or discarded.

The conference system receiver must discriminate between messages arriving from several different hosts, and even from one host, but representing data from different speakers. To sort these out, all data messages except those from the current speaker are discarded immediately when they arrive. When a command is received from the chairman to listen to a speaker, any data messages from the previous speaker which are waiting for processing are immediately discarded. This approach assures that only one speaker's messages will be available for selection at any time.

In the current network voice conference protocol the speaker's transmitter must send copies of each data message to all other hosts. The list of hosts who are to be sent data messages and the LINK to be used at each is included in the "Speak To" control message. This list is subsequently used by the transmitter to generate the HOST/IMP leader for each copy. When no local

participant is speaking, there must be a way of inhibiting transmission of data messages. This is currently accomplished by use of a transmission enable flag. If this flag is not set, any data messages prepared are discarded. The transmit enable flag is set by the LCC approximately 154 milliseconds after a "Speak To" message is received and cleared immediately when a "Shut Up" message is received. If the flag is set and a data message is ready, the message leader is filled in for one host at a time and the message is sent to the network interface processor for transmission to the IMP. When the IMP has accepted this copy, the next copy is prepared and sent. When all copies of the message have been sent, the message is discarded.

3. Participant Interaction

Participant input to the CHAIR is needed to allow it to schedule speakers on demand. This control is provided through the LCC. In addition, a user must be able to indicate to the LCC that he wishes to join a conference or that he is ready to leave it. The LCC, in turn, needs to inform the user that he has been accepted into the conference, that he has the floor and may speak, or that he must stop speaking. This communication between the LCC and the local participants is provided by the use of two lights and a 15 key keyboard, which together with a microphone and earphone make up a conference console.

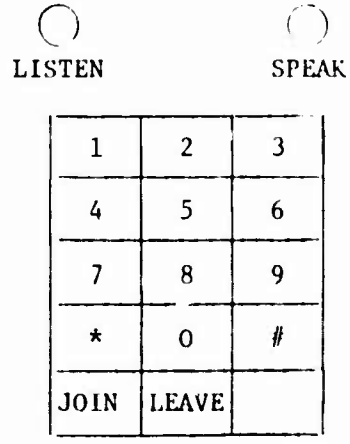


Figure 1

Conference Participant's Console

A user indicates that he wishes to participate in the conference by pushing the JOIN key. No user identification is used at this time, so the LCC sends a "Request to Join" message to the CHAIR, identifying the participant by the conference console (1-4) where he is located. When the participant is accepted by the CHAIR, the LISTEN light is lit and the user's earphone is enabled for vocoder output.

A participant wishing to leave the conference pushes the LEAVE key. The LCC then sends a "Lost a Participant" message to the CHAIR. When a "Remove a Participant" message is received, the LCC clears both lights for the participant named.

Participant communication to the CHAIR is supported through user function codes generated by the keys 1 through 9. Functions 1 and 2 are assigned meanings of "I want to talk" and "I've finished talking", respectively. These functions are for the CHAIR's information only, the LCC does not attempt to interpret them.

The primary communications between the LCC and the user are provided in response to the control commands "Speak To....." and "Shut Up" received by the LCC and naming a participant as the speaker. The SPEAK light is lit and the LISTEN light is turned off to indicate that the participant has the floor when the "Speak To....." message is received. The SPEAK light is turned off and the listen light lit when the "Shut Up" message is received. In addition, the CHAIR to participant function 1, which is a request to "Wrap Up" what is being said, is indicated to the participant, if he is speaking, by turning on his listen light.

Since the current conference consoles are actually general purpose interactive terminals, there is also a display screen (a direct view storage tube) available for additional information from the LCC. At present, the LCC prints the identification (host name and extension number) of the speaker each time a "Listen To....." message is received.

In an attempt to improve the responsiveness of the conference system to the user, the LCC also monitors the response of the CHAIR during the conference. Every five seconds an INQ control message, as defined by the network voice protocol, is sent to the host of the CHAIR. If a READY response is not received within ten seconds a message is displayed for local participants warning them that the CHAIR may be gone. Similarly, if a IMP/HOST message is received indicating that the host is dead, it is passed on to local participants. The aim is to lessen the feeling of isolation that can arise when there is no response to user requests from the system.

B. The Conference Chairman

The role of the CHAIR is primarily to schedule speakers for the conference and to communicate this information to the LCCs. To help it in its scheduling duties, individual participants can notify the CHAIR that they wish to speak or are through speaking.

It is possible for the CHAIR to be implemented entirely as an automatic program, much like the processor scheduler in a single processor operating system. On the other hand, all decisions could be made by a human chairperson, with the CHAIR program simply making the participant requests known and accepting manual control of each speaker change. We have chosen to use a mixed strategy, with a human chairperson informed of all requests and the actions taken by the CHAIR program. Both program and human are able to make a limited set of switching decisions.

1. The CHAIR Program

The speaker selection algorithm used by the CHAIR program is first-come, first-served, based on user "Request to Speak" functions. The speaker is changed when the current speaker sends a "Speaker Done" function. The CHAIR responds by sending a "Shut Up" control message to the speaker's LCC. If any requests to speak are outstanding, the oldest one is select. "Listen To" messages are sent to each active LCC. A "Speak To" message is prepared listing each active host and this message is sent to the LCC of the new speaker.

The human chairman can presently override the speaker selection procedures in only two ways. First, he can force the selection of the next speaker in line, which has the same effect as receiving a "Speak Done" function from the current speaker. Second, he can insert a request to speak from his own extension at the head of the queue of pending requests. These facilities permit the human chairman to reply to the current speaker, or to break off the current speaker if he is unable or unwilling to voluntarily finish what he has to say. They do not permit selection of an arbitrary participant as the next speaker, or skipping a particular participant's request to speak (although his turn can be made arbitrarily short). Such facilities are consistent with the current structure, and could be added if they appear to be necessary.

In addition to the speaker selection function, the CHAIR is also responsible for negotiating with LCCs about the vocoder parameters and accepting or rejecting requests from LCCs to add new participants to the conference. These functions are presently performed entirely by the CHAIR program, although at some future time it may be desirable for the human chairman to have some control over participation in the conference. Vocoder parameter negotiation is initiated when a "Request to Join the Conference" is received over the initial connection link. The CHAIR always serves as negotiation master. The CHAIR accepts any "Request to Join" from an LCC once the negotiation is completed, unless the system limit for participants (currently 20) is exceeded, in which case a "Remove a Participant" message is sent to reject the new participant. In the same spirit, a participant is removed from the conference only in response to a "Lost a Participant" message from an LCC. The CHAIR does not voluntarily terminate an individual's participation in the conference.

The CHAIR program uses two data structures to manage information about the conference. The first, UHOSTS, has one four-word entry for each host whose LCC has joined or is in the process of joining the conference. This entry contains the HOST ID, the link to be used for sending commands to the LCC (and, therefore, the data link, which is the command link + 1), the state of the connection (used primarily during parameter negotiation) and a save cell containing the extension of the user whose request to join the conference triggered the parameter negotiation. "Listen To" messages are sent to all LCCs in the UHOSTS list who have completed parameter negotiation each time a speaker selection takes place.

The second data structure, USERLS, contains a three-word entry for each participant in the conference. This entry contains the Host ID and extension of the participant and a link field used to maintain the queue of pending requests to speak. A participant is added to this list when a "Request to Join" message is received, if he is already present, the message is not acknowledged. A participant is removed when a "Lost a Participant" message is received. Request to speak functions cause the entry for the participant to be linked to the end of the request queue, unless already present, in which case the new request is ignored. When a participant is selected to be speaker, his entry is removed from the request queue. The participant's entry will also be removed from the request queue if a "Speaker Done" function is recieved before the participant is selected as the speaker.

In order to allow for hosts leaving and rejoining the conference, procedures are included for removing a host and all its users from the UHOSTS and USERLS lists. When a participant is removed from the USERLS list, he is also removed from the request to speak queue and, if he is speaking, the next speaker is selected. There are three ways in which a host may currently leave the conference. The only normal way is for the LCC is send a termination control message. A "destination host or IMP dead" IMP/HOST message is recognized as an indication that the host LCC is no longer accessible, and the host is removed. Finally, if a "Request to Join" control message is received on the initial connection link, it is assumed that the host's LCC has been reinitialized. In this last case, the host is removed and immediately reinserted as the vocoder parameter negotiation begins. The host entry is also removed if the vocoder negotiation fails.

2. Interaction with the Human Chairman

The human chairman communicates with the CHAIR program through a set of keys on his input keyboard. Output from the program to the chairman is on a direct view storage tube. Any of the four standard user consoles connected to the CHI system may be used as a chairman's station. The console used is the one where the Network Voice Conference program is initiated. The chairman control keys are disjoint from the user control keys, since a chairman is normally also a participant in the conference. There are currently five control input keys whose functions are defined, seven additional keys are available in the same cluster.

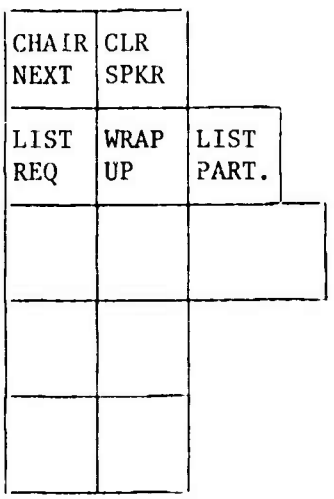


Figure 2

Chairman Console Input Keys

Pushing the CHAIR NEXT keys causes the chairman, if he is a participant in the conference, to be placed at the head of the queue of requests to speak. the CLR SPKR input causes a "Shut Up" message to be sent to the current speaker and schedules the next participant in the request queue. The WRAP UP input causes a CHAIR to USER control message with the Wrap Up function code to be sent to the current speaker, if any. The remaining input keys allow the chairman to have listed at his console the participants requesting to speak (LIST REQ) or all the participants (LIST PART). The listing is in the form hostname-extension number.

At present, most control message arrivals and speaker selections are displayed at the chairman's console. This provides a running record of the speaker requests and scheduling, as well as the arrival and departure of participants and hosts.

3. Relationship with the LCC

The CHAIR program is entirely independent of the LCC which may be running at the same time. The CHAIR does not require a vocoder, communicating with all LCCs and participants, including local participants, through control messages on the ARPANET. Our CHAIR program shares an input keyboard with the LCC if the chairman is a participant in the conference. Since distinct keys are used for the input to each, there is no difficulty in directing the inputs to the proper process. Control input from the ARPANET is separated between LCC and CHAIR through separate LINK assignment for each, just as data is separated from control by LINK.

IV. MONITORING OF NETWORK VOICE CONFERENCE EXPERIMENTS

During network conferencing experiments a record is made of events as they occur, together with relevant information about the state of the system at the time the event occurs. The purpose of this data gathering is to obtain more information to judge the behavior of the communications network and the computer systems which are connected through it. We are also interested in evaluating the effectiveness of the procedures adopted to deal with their behavior. Finally, we want to detect differences in implementation of the voice protocols which may affect the performance of the conference system.

A. Events Monitored

The events currently being monitored include the sending or processing of a control message, the transmission of a data message, the selection of an input message for processing and discarding of an out-of-order data message. For control messages, the time the message was processed or created is recorded along with the message itself. For data messages both the time the message was received or generated and the time it was processed, or the last copy sent, are recorded. In addition, the data message leader and header and the expected network transit time (NT) are saved. Approximately 400 to 800 events are traced each minute, with higher levels of activity occuring when the CHAIR is located at CHI.

B. Processing of Trace Data

The trace data gathered from a conferencing session is written on disk during the session using a file specified when the conference system was loaded. This file is normally processed using the Signal Interactive Mathematical System, an array oriented, programmable, interpretive system accessible from the CHI user consoles. This system permits computation and selection operations on the data files and can display graphical or numerical data.

Typically, we begin processing of the information from a session by selecting the control message entries and printing them. Table 1 illustrates a portion of such an output. This listing provides a summary of the control flow in the conference showing the speaker switching times. Response times over

the ARPANET can be determined by measuring the time from when a stimulus message such as a "Speaker Done" function is sent and the response, the "Shut Up" message from the chair. Similarly, speaker selection time can be measured by comparing the time a "Request to Speak" function is sent to the time the "Speak To....." message is received. The sample trace of Table 1 is from a network conference demonstration on January 23. This conference was chaired by extension 4 at Lincoln Labs (HOSTID = 202). SRI (HOSTID = 51), ISI (HOSTID = 22) and CHI (HOSTID = 182) also participated. The first line of this trace is a "Request to Speak" from extension 1 at CHI. The third line is the "Speak To" message from LL-4 telling CHI-1 that he may speak and directing the LCC to send data message to three hosts; SRI with link 241, ISI with link 241 and LL with link 225. The difference in the TIME column for these two events is 35 time units, about 670 milliseconds. The following line (INDEX 3306) is a message from CHI-1 to the CHAIR that he is done speaking. The response of the CHAIR on the next line is the control message "Shut Up." The time for this exchange is 43 time units (826 milliseconds). Five other cases of this combination on the same page of the trace record range from 450 to 730 milliseconds; the elapsed time is about 257 seconds. Three other cases of a "Request to Speak" from CHI being granted immediately (i.e., no one was speaking) range from 595 to 979 milliseconds delay to the "Speak To" message. These variations represent differences in both the network performance and the response of the systems at Lincoln and Culler/Harrison.

In addition to these timing measurements, the trace of control messages sent and received has been very valuable in identifying differences or errors in implementations of the protocols at the participating sites. We have already used this means to discover problems in our own implementation during preliminary testing with only ourselves in the conference. Other problems that occurred during initial experiments with other sites, including the conditions expected for successful termination of the vocoder negotiations and transmission of invalid host identifiers in the "Speak To" command, were found or verified using this information.

The data message entries in the trace record have been an important source of information for us in developing an implementation of the input message selection procedures described earlier. They provide evidence of the variations which occur in network transit times and make it possible to determine directly the amount of data transferred over the network during a conference. We expect

through a continuing series of experiments under varying conditions of network loading to obtain more data about the variations in throughput and in out-of-order arrivals of messages.

Table 2 is a part of the trace from the same conferencing demonstration illustrated earlier. This record contains entries created when an input message was selected for processing. The column headed HOST ID identifies the entries as coming first from Lincoln (202) then ISI (22) and finally Lincoln for a second time. The entry at index 21 illustrates a difficulty in the sender's transmission program. Its time stamp of -25582 identifies it as belonging immediately after the last message sent before ISI spoke (index 8). This message should have been sent earlier or discarded without being transmitted. Its arrival with the second set of messages caused an abnormality in the computed network transit time (NT). If this new value of NT was used to time the selection of the following messages, it would force a long delay and exhaust the message buffering capability of the receiver. Indeed, this was precisely what happened in an earlier experiment. From examination of the trace of that experiment it was evident what the problem was, and the input selection logic was modified to reset its expected time when the observed time was much shorter. This is illustrated in the trace by the value for NT in the subsequent entries being much less than the 8969 value computed from the first message.

In general, the amount of data gathered from the data entries is more easily interpreted through graphs of selected values. Figures 3 and 4 illustrate two graphs derived from the trace information. Figure 3 shows the value of NT as it varies from speaker to speaker. The differences in value between speakers represent primarily the different time frames used for the time stamps in their messages. The largest value of NT was obtained for ISI, the middle value is Lincoln, the lowest value is SRI. The relative flatness of each speaker's segment illustrates the small variation in network transit times during the two minutes period shown. The second graph, Figure 4, illustrates the short term variations in observed transit time (OT) from the expected time (NT) for the same set of messages. The variation rarely exceeded ten time units or 192 milliseconds, and except for the messages from Lincoln, was generally less than half that. This behavior appears to be better than often occurs, particularly at the time of day of this demonstration (1100-1200 PST).

The data gathered here suggests that a delay (D) of 10 to 12 time units plus the message length should be sufficient to insure that essentially all messages will arrive on time. In fact, a somewhat shorter delay of 15 time units, instead of the 20 called for, was used. There were 29 messages out of over 4500 which arrived out of order and so late that they were discarded. All but one of these came from Lincoln, which is over twice as far from CHI as either of the other sites involved.

Table 1: Sample Trace Record -- Control Message Entries

TRACE RECORD

TRACK	TIME	HOST ID	LINK	COMMAND	CHAIR	WHO		TYPE
2732	30470	202	234	40	20204	1	101	CMO
2733	30509	202	250	36	20204	18201		CMI
2739	30565	202	250	37	20204	18201	3	CMI
2739	30565	5141	2241	20225				CMI
3200	32096	202	234	40	20204	1	102	CMO
3215	32139	202	250	38	20204	18201		CMI
3214	32235	202	250	36	20204	20204		CMI
3179	-32199	202	234	40	20204	1	101	CMO
3175	-32170	202	250	36	20204	18201		CMI
3175	-32167	202	250	37	20204	18201	3	CMI
3175	-32167	5141	2241	20225				CMI
3173	-32030	202	250	39	20204	1	101	CMI
3141	-31677	202	234	40	20204	1	102	CMO
3137	-31640	202	250	38	20204	18201		CMI
3138	-31634	202	250	36	20204	20201		CMI
3703	-30677	202	250	36	20204	2200		CMI
3713	-30525	202	234	40	20204	1	101	CMO
3751	-30005	202	250	36	20204	20204		CMI
3189	-29192	202	250	36	20204	18201		CMI
3186	-29191	202	250	37	20204	18201	3	CMI
3186	-29191	5141	2241	20225				CMI
4139	-28560	202	234	40	20204	1	102	CMO
4146	-28322	202	250	38	20204	18201		CMI
4147	-28165	202	250	36	20204	20204		CMI
4200	-27703	202	250	36	20204	20201		CMI
4270	-27187	202	234	40	20204	1	101	CMO
4271	-27148	202	250	36	20204	18201		CMI
4272	-27136	202	250	37	20204	18201	2	CMI
4273	-27136	2241	20225					CMI
4430	-26359	202	234	40	20204	1	102	CMO
4435	-26320	202	250	38	20204	18201		CMI
4436	-26201	202	250	36	20204	20204		CMI
4525	-25663	202	250	36	20204	20202		CMI
4533	-25502	202	250	36	20204	20204		CMI
4590	-25444	202	234	40	20204	1	101	CMO
4599	-25419	202	250	36	20204	18201		CMI
4730	-25413	202	250	37	20204	18201	3	CMI
4701	-25413	5141	2241	20225				CMI
4999	-22547	202	234	40	20204	1	102	CMO
5005	-22522	202	250	38	20204	18201		CMI
5007	-22504	202	250	36	20204	5100		CMI
5071	-21531	202	250	36	20204	2200		CMI
5099	-21152	202	234	40	20204	1	101	CMO
5110	-20939	202	250	36	20204	20204		CMI
5206	-20302	202	250	36	20204	18201		CMI
5207	-20300	202	250	37	20204	18201	3	CMI
5200	-20300	5141	2241	20225				CMI
5272	-20071	202	234	40	20204	1	102	CMO
5273	-20038	202	250	38	20204	18201		CMI
5279	-19744	202	250	36	20204	20204		CMI

NOTES:
1. TYPE is CMI for control message input and CMO for control message output.
2. TIME is in 19.2 millisecond units.
3. CHAIR and WHO are given as HOSTID and EXTENSION (HHHXX).
4. The send list in SPEAK TO commands (37) is given on the following line as HOSTID and LINK-200 (HHHLL).
5. Function codes in chairman user commands (39 and 40)are given as user's extension and function code (XXCC).

Table 2: Sample Trace Record -- Data Messages Processed

BLOCK	RECEIVE TRACE RECORD TIME	HOST ID	ACONT	TSTAMP	PCNT	RCVD	NT	TYPE
0	-18254	3202	9	-25645	7	-18254	7307	DMI
1	-18236	3202	0	-25638	7	-18236	7309	DMI
2	-18234	3202	4	-25631	7	-18235	7303	DMI
3	-18223	3202	1	-25624	7	-18223	7309	DMI
4	-18221	3202	5	-25617	7	-18222	7309	DMI
5	-18219	3202	10	-25610	7	-18219	7308	DMI
6	-18209	3202	8	-25603	7	-18209	7303	DMI
7	-18199	3202	6	-25596	7	-18199	7308	DMI
8	-18197	3202	9	-25589	7	-18193	7307	DMI
9	-16850	3022	0	-31185	14	-16851	14320	DMI
10	-16837	3022	1	-31171	14	-16837	14320	DMI
11	-16817	3022	0	-31157	14	-16817	14321	DMI
12	-16809	3022	7	-31143	14	-16809	14321	DMI
13	-16797	3022	8	-31129	14	-16797	14321	DMI
14	-16783	3022	9	-31115	14	-16784	14320	DMI
15	-16770	3022	10	-31101	14	-16770	14320	DMI
16	-16751	3022	5	-31087	14	-16751	14320	DMI
17	-16737	3022	5	-31073	14	-16737	14320	DMI
18	-16726	3022	8	-31059	14	-16726	14320	DMI
19	-16710	3022	7	-31045	14	-16711	14320	DMI
20	-16701	3022	10	-31031	7	-16705	14320	DMI
21	-16598	3202	0	-25582	4	-16609	8969	DMI
22	-16591	3202	0	-23996	7	-16599	7300	DMI
23	-16579	3202	0	-23974	7	-16583	7300	DMI
24	-16577	3202	4	-23967	7	-16579	7300	DMI
25	-16569	3202	4	-23960	7	-16569	7390	DMI
26	-16567	3202	9	-23953	7	-16567	7300	DMI
27	-16554	3202	3	-23946	7	-16554	7381	DMI
28	-16537	3202	0	-23939	7	-16537	7383	DMI
29	-16535	3202	4	-23932	7	-16539	7382	DMI
30	-16533	3202	8	-23925	7	-16536	7382	DMI
31	-16525	3202	9	-23918	7	-16525	7302	DMI
32	-16520	3202	10	-23911	7	-16523	7382	DMI
33	-16511	3202	9	-23904	7	-16511	7382	DMI
34	-16506	3202	10	-23897	7	-16515	7381	DMI
35	-16491	3202	3	-23890	7	-16491	7382	DMI
36	-16489	3202	8	-23883	7	-16493	7382	DMI
37	-16483	3202	9	-23876	7	-16483	7382	DMI
38	-16478	3202	10	-23869	7	-16481	7382	DMI
39	-16470	3202	10	-23862	7	-16472	7382	DMI
40	-16463	3202	10	-23855	7	-16468	7382	DMI
41	-16456	3202	10	-23848	7	-16457	7382	DMI
42	-16445	3202	7	-23841	7	-16445	7383	DMI
43	-16442	3202	10	-23834	7	-16440	7382	DMI
44	-16435	3202	10	-23827	7	-16437	7382	DMI
45	-16421	3202	4	-23820	7	-16421	7383	DMI
46	-16418	3202	7	-23813	7	-16427	7382	DMI
47	-16414	3202	11	-23806	7	-16414	7382	DMI
48	-16407	3202	10	-23799	7	-16412	7382	DMI
49	-16400	3202	10	-23792	7	-16405	7382	DMI
50	-16393	3202	10	-23785	7	-16395	7382	DMI

NOTES:
1. TIME is the time the message was selected for processing.
2. AØCNT is the number of frames of data ready for D/A output.
3. RCVD is the time the message was received from the net.

EXPECTED NETWORK TIME (NT)

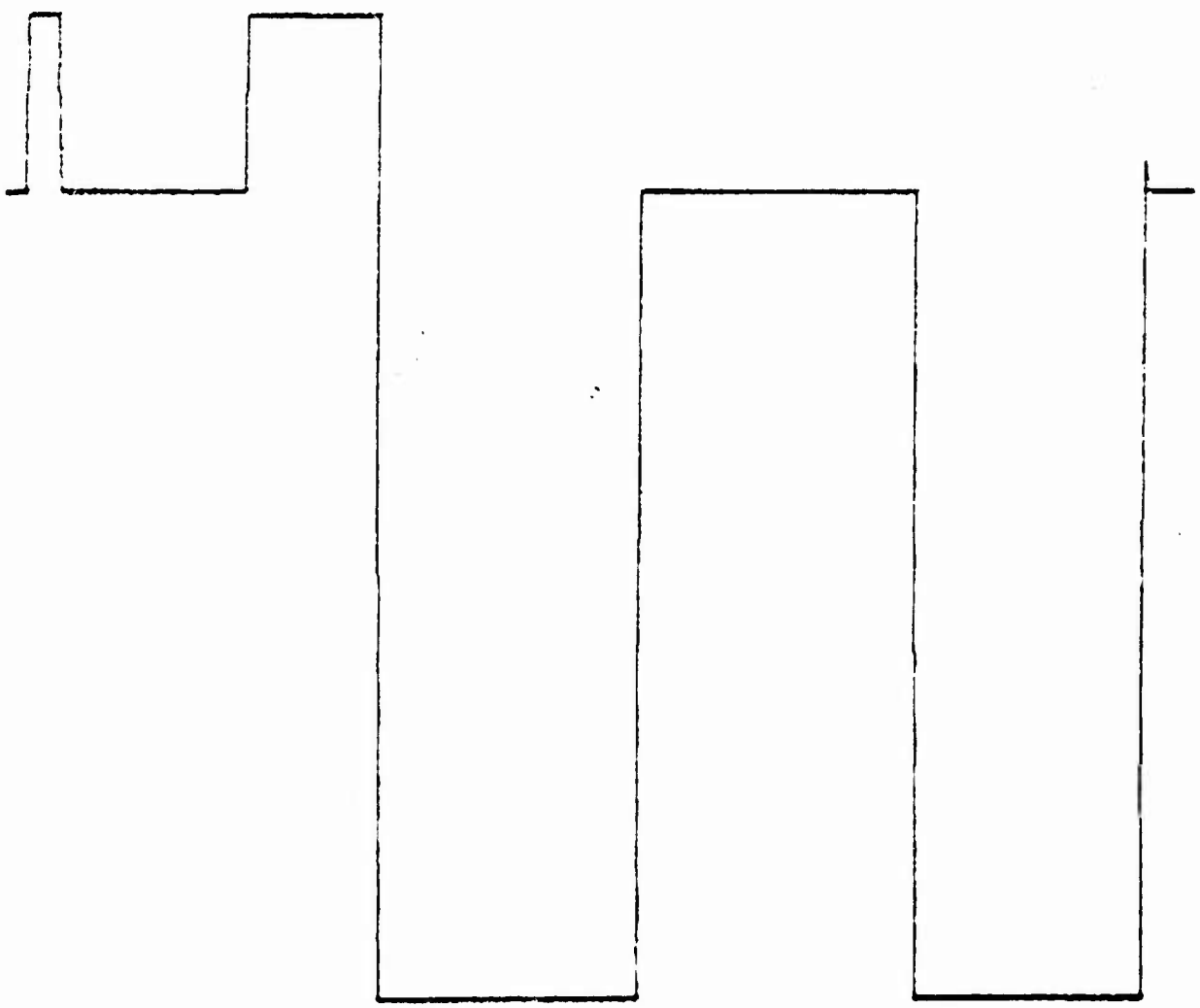


Figure 3

ØT-NT for LINCØLN,ISI,SRI
NETWØRK CØNFERENCING DEMØNSTRATIØN
JANUARY 23, 1976

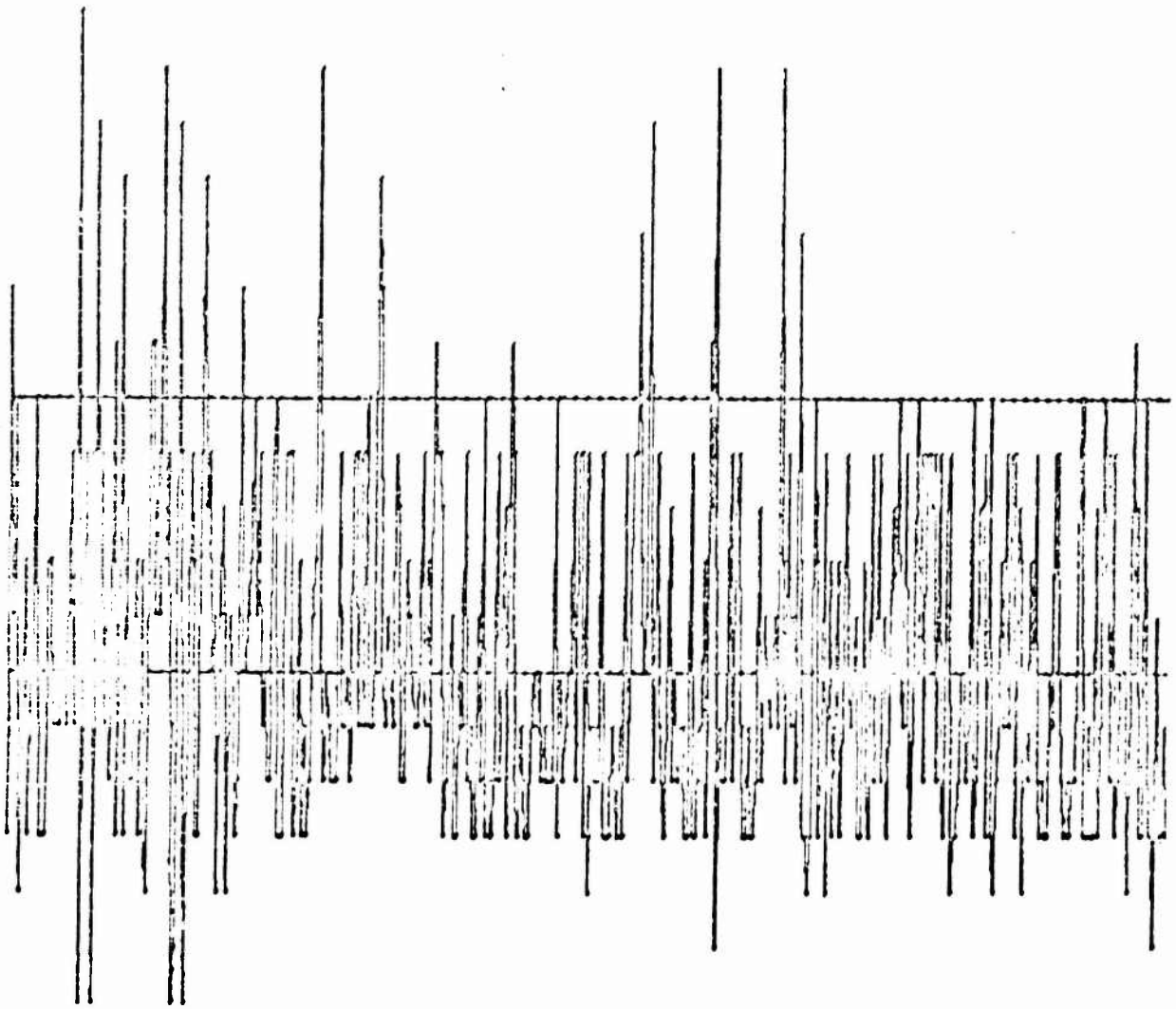


Figure 4

Appendix A: NVCP Control Messages

Only conferencing protocol control messages are included here. A complete listing of these messages appears in [3]. Reference [2] describes the messages of the network voice protocol.

The following terms are used:

<CHAIR>	The 8-bit HOSTID and 8-bit EXTENSION of the chairman. This appears as the second word in all NVCP control messages. Shown in trace listings as 3-digit HOSTID and 2-digit EXTENSION.
<USER> <PARTICIPANT> <SPEAKER>	The HOSTID and EXTENSION of an individual participant or potential participant in the conference. Shown in trace listings as 3-digit HOSTID and 2-digit EXTENSION.
<FUNCTION>	The 8-bit EXTENSION of the participant and an 8-bit function code. Shown in trace listings as 2-digit EXTENSION and 2-digit function code.
<WHERE>	The HOSTID and LINK of an LCC data receiver. Shown in trace listings as 3-digit EXTENSION and 2 low-order digits of LINK. Third digit of link is 2.

1. "Request to Join"

   33, <CHAIR>, <USER>, K

   This message is sent by the LCC to the chair on link 255 to establish a connection. It initiates the standard NVP vocoder negotiation with the calling message replaced by this one. K is the Link on which the LCC expects control messages. K+1 is the link for data messages. Once a connection is established, this message is sent on the CHAIR's control link to introduce additional participants.

2. "Add your Participant"

   34, <CHAIR>, <PARTICIPANT>

   Sent by CHAIR to LCC to accept a participant.

3. "Remove a Participant"

   35, <CHAIR>, <PARTICIPANT>

   Sent by CHAIR to LCC to reject a participant or acknowledge that he has left the conference.

4. "Listen To"

36, <CHAIR>, <SPEAKER>

Sent by CHAIR to LCC to switch to a new speaker.

5. "Speak To....."

37, <CHAIR>, <SPEAKER>, N, <$WHERE_1$>, ···, <$WHERE_N$>

Sent by CHAIR to LCC to allow it to transmit data messages from the named speaker and provides the set of LCC which are to be sent data.

6. "Shut Up"

38, <CHAIR>, <SPEAKER>

Sent by CHAIR to terminate a particular speaker and cause the LCC to stop sending data messages.

7. "Control functions for Users"

39, <CHAIR>, N, <$FUNCTION_1$>, ···, <$FUNCTION_N$>

Sent by CHAIR to provide control functions for specific participants. Function code 1 means "Wrap Up" when sent to the speaker.

8. "Control functions for Chairman"

40, <CHAIR>, N, <$FUNCTION_1$>, ···, <$FUNCTION_N$>

Sent by LCC to provide control functions from particular participants to the chairman. Function code 1 is a "Request to Speak" function. Function code 2 is a "Speaker Done" function.

9. "Lost a Participant"

41, <CHAIR>, <PARTICIPANT>

Sent by an LCC to notify CHAIR that a participant has left the conference. The CHAIR responds with a 35 message.

REFERENCES

1. Culler, G. J., McCammon, M., McGill, J. F., Real Time Implementation of an LPC Algorithm, Quarterly Technical Report, Speech Signal Processing Research at CHI, Culler/Harrison, Inc., Goleta, California, May 1975.

2. Cohen, D., "Specifications for the Network Voice Protocol (NVP)," NSC Note 68, January 1976.

3. Cohen, D., "The Network Voice Conference Protocol (NVCP)", USC/Information Sciences Institute, November 1975.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: CHI-QTR-201
2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle): NETWORK VOICE CONFERENCING. Quarter Technical Report Speech Compression Research at CHI.

5. TYPE OF REPORT & PERIOD COVERED: Quarterly Technical Report. December 1975 - February 1976

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s): Michael McCammon

8. CONTRACT OR GRANT NUMBER(s): DAHC15-73-C-0252, ARPA Order - 2359

9. PERFORMING ORGANIZATION NAME AND ADDRESS: Culler/Harrison, Inc. 150-A Aero Camino Goleta, California

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: Program Code: P5P10

11. CONTROLLING OFFICE NAME AND ADDRESS: Defense Supply Service -- Washington Room 1D 245, The Pentagon Washington, D.C. 20310

12. REPORT DATE: March 1976

13. NUMBER OF PAGES: 25

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): Defense Contract Administration Services District, Van Nuys 14450 Erwin Street Van Nuys, California 91408

15. SECURITY CLASS. (of this report): Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Distribution of this document is unlimited. It may be released to the Clearinghouse, Department of Commerce for sale to the general public.

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

18. SUPPLEMENTARY NOTES

This research was supported by the Defense Advanced Research Projects Agency under ARPA Order No. 2359/4.

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

Speech Compression
Network Voice Conferencing
Linear Predictive Coding
ARPANET
Digital Voice Transmission

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

This report covers the activities of CHI in development of a network voice conferencing system using linear predictive coding for speech data compression from December 1975 through February 1976. The voice conferencing system is an extension of an existing network speech compression system incorporating control messages for speaker selection and data routing, together with transmitting and receiving programs which respond to them. The current implementation of this system at CHI is described. The system monitoring and recording facilities implemented are discussed with examples of their use for analysis of system performance.

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

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)