AD A030925

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

September 1976

See 1473

A VARIABLE FRAME RATE NETWORK VOICE SYSTEM

Quarterly Technical Report

Speech Compression Research at CHI

June 1976 - August 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

Variable frame rate transmission algorithms are an attractive approach for voice transmission on a packet switched network such as the ARPANET. They lower the total number of bits transmitted in a nonuniform manner, which a packet network can take advantage of by sending other messages during time when few bits are needed for speech. However, since VFR algorithms require information from many frames to properly determine the parameters to be used for synthesis of one frame, they introduce new problems with transmission and selection rules for the transmitter and receiver. This report details the problems which we have detected and describes the methods incorporated in the CHI implementation of the VFR LPC-II network speech system to deal with them.

Good quality speech reproduction in a network speech compression system depends on more than the inherent qualities of the vocoder. It is also necessary that the receiver be able to supply the synthesizer with parameters at a fairly constant rate, without gaps exceeding the backlog of data already prepared for output. Providing a steady supply of parameters is complicated by the requirement for breaking the continuous stream of parcels of parameters into discrete messages for transmission over the ARPANET. These separate messages may take different amounts of time to reach their destination, may be lost or arrive out of order and may contain different numbers of parcels. Further complicating the problem of providing continuous speech output is the need to minimize the total delay in the system in order to permit interactive conversations. Chapter II presents a more detailed discussion of the issues involved in maintaining speech quality on the ARPANET and describes the protocols used for network voice communication as a background for the implementation discussion.

Much of the responsibility for quality network speech communication rests with the transmitter. By appropriately selecting the threshold for parameter transmission in the LPC-II system it can increase the speech quality or lower the transmission rate to adjust for varying network performance. By decreasing the number of parcels per message, the transmitter can cut the delays due to message loading time and transmission bit rates at the expense of additional overhead.

The transmitter must also take into account the relationship between variable frame coding and message boundaries. In particular, any time parcels are not transmitted, whether due to silence detection or speaker switching, the transmitter must send all parameters for the first parcel after the gap and should attempt to send all parameters in the last parcel before a gap. A more detailed explanation of the transmitter section of the Culler/Harrison network voice system is provided in Chapter III.

The receiver's function is to make use of the information received over the ARPANET to provide as smooth an output as possible, preserving the audio fidelity, without unnecessarily increasing the delay. One problem faced is how to estimate the minimum delay needed before starting output of the start of a speech segment in order to be sure all parcels of parameters will arrive in time to permit synthesis of speech with no gaps. An approach developed for this estimation is to maintain a continually updated estimate of the time a message will need to traverse the net, then adding an additional delay to allow for variations in network performance. This approach does not tie the output time of a speech segment to the arrival time of any particular message.

A second area of concern for the receiver is proper interpretation of the variable frame rate parameter information to produce valid parameters for each frame. In particular, care must be taken to avoid interpolation between parameters from different speakers or across gaps if high fidelity is to be maintained. The final chapter describes the CHI receiver implementation, providing more detail about the processing of messages in a variable frame rate system.

II. NETWORK SPEECH COMMUNICATION

Variable Frame Rate Speech Compression

The variable frame rate transmission algorithm is a method of further reducing the average transmission rate required for digital voice communication. It does this by not transmitting selected parameters describing a frame of speech when these parameters do not differ significantly from the last set transmitted. This converts a fixed frame rate system, where new parameters are computed every 9.6 milliseconds, to a variable rate system in which parameters are updated only when they have changed significantly. Actually, separate decisions are made for transmission of pitch, gain and reflection coefficient parameters, so the VFR system varies the number of bits per frame while maintaining a fixed frame rate. Three bits are transmitted with each frame to indicate what parameters are included. The variable frame rate algorithm decreases the number of bits transmitted during continuous speech from 47 to an average of less than 20 per frame.

Transmission rates are further reduced in a packet network by recognizing periods of silence when the gain parameter stays below a selected threshold. No parameters or frame information are transmitted during these silence periods, leaving the entire network bandwidth available for other traffic. In a typical two-way conversation, each speaker will be transmitting less than half the time because of silence detection. The Network Voice Conference system provides for additional reduction in network traffic by restricting speech transmission to the current speaker.

Maintaining Speech Quality

The quality of speech obtained in a network speech compression system depends on a number of factors. Cohen has identified three factors: the acoustic quality or fidelity of the reproduction, its continuity or smoothness, and the delay between the original speech and its reproduction at the destination [1]. The problem for a system is to maintain high fidelity and continuity of the reproduced speech while minimizing the end-to-end delay and not exceeding the desired transmission rate.

The question of fidelity of the reproduced speech has been examined in some length in the previous technical report [2]. To obtain continuity or smoothness in the reproduced speech, we must be able to synthesize the frames of data at a rate sufficient to provide a steady audio signal output. The problem here is not the computation time required, which is very little in modern signal processors. It is assuring the availability of a steady stream of parameters to control the synthesis. When the effects of the network are removed, as when the transmitter and receiver are located at the same site, the LPC-II system vocoder provides satisfactory acoustic fidelity while considerably reducing the average transmission rate. In the ARPANET, several parcels of parameters for a frame are transmitted as one network message which arrives as a unit. Essentially all messages eventually reach their destination. There is a considerable variation, however, in the time a message may take to reach its destination. If the receiver has output all the data generated from the parcels in one message before the following message has arrived, a discontinuity will occur in the speech. To avoid this problem, the receiver can wait before starting to output data generated after a silence or when a new speaker begins. The delay must be equal to the maximum deviation of any message's travel time from that of the first message. This time can only be estimated, of course. If the estimate is too low, a break will occur in the output. The delay is also subject to considerable variation as network performance changes.

When uncontrolled network messages are used to gain increased throughput the variations can cause additional problems. Messages may arrive out of order or even be lost completely. If enough delay is not included to allow for reordering at the receiver, glitches are created because of missing parameters from the late message. With variable frame rate algorithms, frames at the beginning of the following message may be unusable because they need parameters from frames in the late message. Hence the loss or excessive delay of one message may make it impossible to correctly synthesize more time than that represented by the message.

The third dimension of the quality of a speech compression system is the amount of delay which accumulates in the end-to-end transmission from speaker to listener. This delay is apparent to a person using the system as the time he must wait to receive a response to anything he says. It is most disruptive in uncontrolled, full duplex conversations when two people may begin speaking and neither will realize that the other is talking for this delay time. If the delay is more than 1/2 to 1 second, considerable adjustment is required, including careful alternation of speakers and explicit indication by each of when they are done talking. In controlled conference situations, where only one speaker at a time is permitted by the protocols, the problems are less, but delays of over a second are annoying.

The total delay from source to destination includes several parts. The vocoder analysis requires several frames of data to compute the parameters for each frame, introducing a delay of 40 to 50 milliseconds. The packing of many frames of data into each message for network transmission adds a delay equal to the total amount of speech represented by the message. As the efficiency of the vocoder improves in minimizing the average number of bits per frame the number of frames which can be packed into one fixed length message increases. For example, a single packet uncontrolled speech data message can have up to 938 data bits. This could represent from 180 milliseconds to over 3 seconds worth of speech.

The time a message spends in the network, traveling from source to destination, is an additional delay. It is assumed this delay increases for longer messages, but probably not proportionately. A lower bound on the network time can be obtained by counting the number of IMP's through which a message must pass along its shortest possible path and adding the times for the message to be serially transferred into each. For 1000-bit messages and 50 Kbps lines this time is 20 milliseconds/IMP [3]. For a relatively short path from CHI to ISI involving five intermediate IMPs, this adds at least 100 milliseconds to the delay. Observed times for short control messages indicate that about 80 milliseconds must be added to this for minimum delays due to IMP queuing and processing and transfer to local HOST processors.

The total minimum delay for a message is the sum of these individual delays and totals 220 milliseconds plus the amount of speech in the message. However, as observed in the discussion of maintaining continuity of speech, an additional delay must be added to account for the variation in arrival time for messages. By examining the factors that make up this variation, we see that it includes the variation in the amount of speech packed into each message and variations in the network performance. The first factor can be controlled by limiting the number of parcels per message to produce a fairly fixed delay due to message packing. If exactly 18 parcels are included in each message, we obtain a minimum delay of about 400 milliseconds. We must still add sufficient delay for the greatest expected deviation in network performance. If this addition is under 100 milliseconds we will have a total delay which is probably acceptable. Longer delays which result from lower network performance or longer transmission paths (CHI to LINCOLN is 10 IMPs) make uncomfortably long delays.

Network Voice Protocols

The efforts to provide good quality digital voice communication on the ARPANET have resulted in the development of protocols to provide standardization of the form of this communication and to insure that enough information is provided to the receiver to enable him to accurately reconstruct the speech waveform. These protocols provide for control messages which permit establishment of network voice communication, agreement on the form of vocoding and control of speakers when necessary. We will not review these commands in this report since they are not the area of concern here. We will concentrate on the rules and conventions for data transfer.

Each data message, in addition to a number of parcels of frame parameters, carries the standard HOST/IMP leader and a 32-bit network voice protocol header. The leader contains the HOST-ID of the destination or source and the LINK number specified by the receiving host for data messages. A separate LINK is used for control messages. The HOST-ID may be used by the receiver to discriminate between messages arriving from different speakers in a voice conference. The NVP header contains five fields:

- TIME - a 16-bit time stamp, giving the parcel number of the first parcel in the message.
- SP - a 1-bit skipped-parcels flag. Set when parcels immediately preceding the message were not transmitted.
- PC - a 7-bit parcel count of the number of parcels in the message.
- ST - a 1-bit indicator of whether this data is part of the primary or secondary data stream during a conference.
- EXT - a 7-bit extension of the individual speaker whose speech is being transmitted.

The extension and stream information are combined with the HOST-ID to provide complete speaker identification.and permit up to two simultaneous speakers talking to subsets of the total group under proposed extensions to the conference protocols. The time stamp and parcel count enable the receiver to properly order the messages as received and detect delayed or missing messages. They are also valuable in determining the variations in network performance in order to establish the appropriate delay before synthesis. The skipped-parcels flag provides an explicit indicator of the beginning of a speech burst and may be used for initiation of the synthesis delay.

The parcels field in a data message contains a variable number of parcels, each representing one speech frame. With variable frame rate transmission the size of each parcel may vary. Each parcel begins with three "presence-bits" which are one if the corresponding parameter type is present for that parcel and zero if it is not. If all three bits are zero the parcel is only three bits long. If all bits are on, the parcel is 50 bits long.

The variable frame rate transmission algorithm requires only a limited amount of agreement between the transmitter and receiver. Gain or reflection coefficients are transmitted whenever their distance from the previous ones transmitted exceeds a threshold. The receiver uses linear interpolation to fill in missing values if there is less than 100 milliseconds to the next occurrence of the parameter. Otherwise, or if a voicing change intervenes, the old values are used. Interpolation is used to avoid sharp transitions to the next transmitted values. Pitch is transmitted each time it changes, and no interpolation is used at the receiver.

Explicit silence detection is provided for by defining a silence threshold, measured in the same units as gain, and a time before silence interval. When the transmitter detects that the calculated gain has been below the threshold value for the specified amount of time, it ceases transmitting parcels, discarding all but the most recent ones. When the gain once again exceeds the threshold, transmission is resumed beginning with the parcels saved. The skipped-parcels bit is set in the NVP header of the first message after silence.

The receiver has no explicit indicator when the transmitter has declared silence. It is necessary to infer silence from the absence of further messages. The skipped-parcels bit in a message assure the receiver that this message is the first after silence, rather than being out of order, and can be used to start the delay after silence.

III. LPC-II TRANSMITTER SYSTEM

The transmitter's function is to take an input speech signal and produce output data messages which are sent over the ARPANET. It must provide the information needed by the receiver to reproduce the speech while attempting to minimize both the bit rate and the delay in delivering the message. The transmitter performs its function in a number of separate steps, each of which is implemented as a separate process within the total system; figure 1 illustrates the data flow through these processes.

Data Input

The input signal may come from one of four voice terminals which can be connected to the CHI system. The selection of which terminal to use is under control of the Local Conference Controller (LCC) program, which acts on control messages from the conference chairman (CHAIR). The signal is low pass filtered and sampled at 6.67kHz by an analog-to-digital converter. The data is stored in an input data buffer until one frame of data (64 points or 9.6 milliseconds) has been collected. This buffer is then queued for processing.

LPC Processing

The computational parts of the linear predictive coding of the speech data are performed in a separate processor, the AP90. The analysis calculation is performed at a fixed frame rate of about 104 frames/second. The AP computes the reflection coefficients (Ks) gain, and pitch and voicing estimates. It also performs the likelihood ratio test to determine if the reflection coefficients should be transmitted. If the coefficients are to be transmitted, it computes the autocorrelation of the predictor coefficients to use in future tests. The result of this test is reported to the MP along with the Ks, gain, pitch and voicing.

Parcel Processing

The output of the AP analysis is processed by ANPOST. The pitch and voicing estimates are refined to produce the pitch parameter. The pitch, gain and Ks

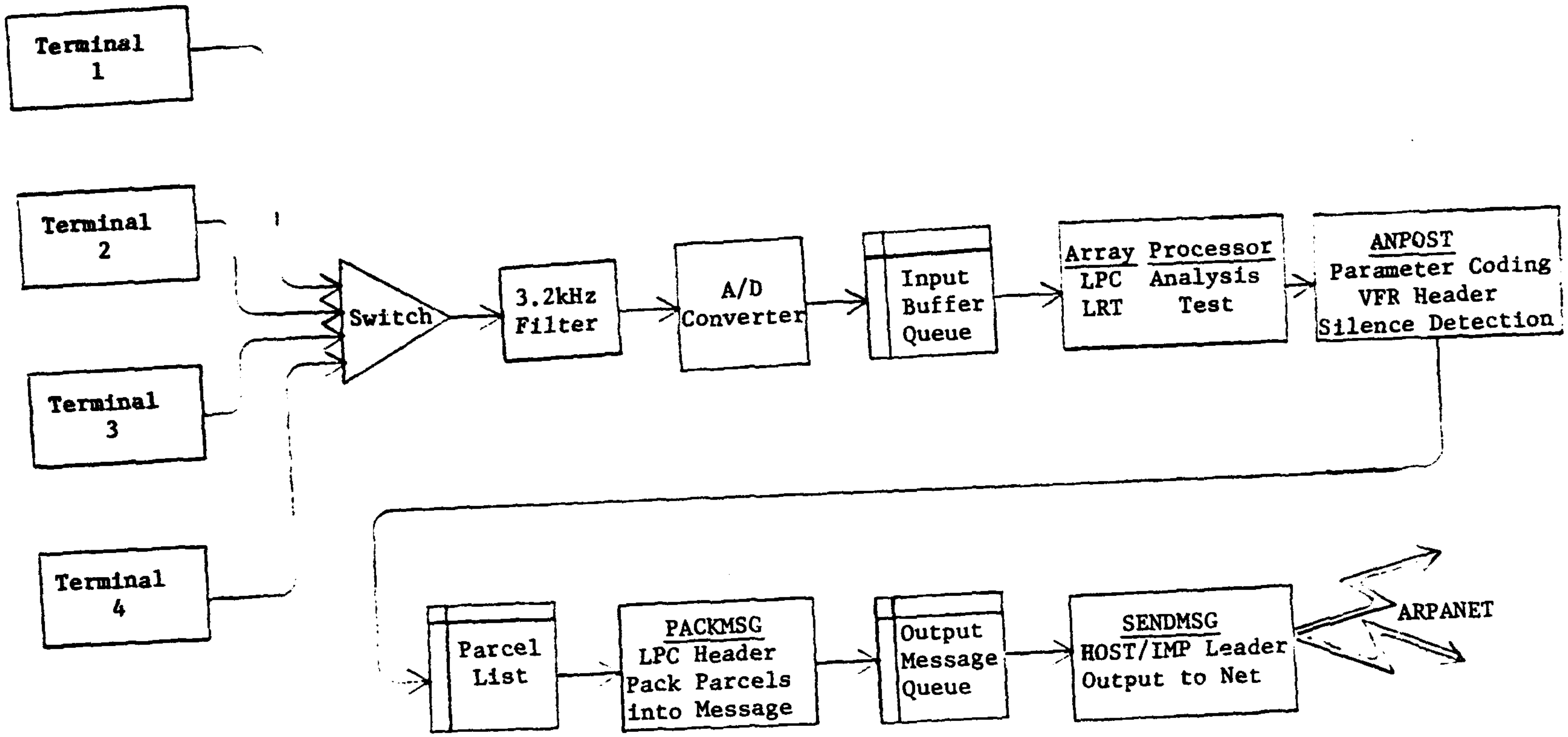

Figure 1: Transmitter Data Flow

are then encoded in separate words and the VFR distance tests are performed to determine which parameters will be transmitted. The 3-bit frame header is constructed at this time and this header is stored with the encoded parameters in the output list. The header code is used to index a table of bit lengths for the frame and the entry is added to the bit count for the output list.

A silence test is performed by comparing the gain of the frame to a threshold value; if the gain is below the threshold for a specified number of consecutive frames, silence is declared. If silence has not been declared, the bit count for the list is tested to see if there will be room for at least one more parcel. If not, or if the number of parcels has reached a prescribed upper limit, the message packing process is initiated. The two tests are needed because of the wide variation in the number of parcels that will fit in one message. The maximum number of parcels per frame is limited to 41. These limits are absolute upper bounds. Normally, a lower limit on the number of parcels is tested when network performance is such that there is no backlog of messages to send. This lower value is usually from 15 to 25 parcels per message to give message loading times of 145 to 240 milliseconds.

If silence is declared, the last parcel's header is forced to seven, causing all parameters to be transmitted. The message packing process is initiated to send whatever parcels are in the output list.

Once silence has been declared, gain of each parcel is tested to see if it exceeds the silence threshold. As long as the gain remains below the threshold, silence is continued. The output list is allowed to build up to eight parcels. Once it has reached eight parcels, the oldest parcel is discarded and the bit count is adjusted each time a new parcel is stored. When the silence threshold is exceeded, silence is terminated. If the output list had reached eight parcels, the skipped-parcels flag is set for the message packing process. The output list is allowed to continue growing to normal message sizes.

Message Preparation

Message packing is a separate process which normally is scheduled only when all copies of previously prepared messages have been delivered to the network. As noted above, if enough data is present to completely fill a message, the packing processes will be entered directly. This allows longer messages when network performance is very poor; thereby making more efficient use of the network.

If there is no local speaker, the output list is cleared without preparing a message. Otherwise, the frame number of the first parcel, the parcel count, skipped-parcels flag and speaker's extension are merged to form the NVP header for a new data message. If the skipped parcels flag is set the header code for the first parcel in the message is set to seven, ensuring that all parameters are present. The coded parameters and code for each parcel are then packed into the new message using the code to determine which parameters are to be included. The new message is placed on an output message queue to await transmission.

Message Transmission

The message transmission process is scheduled when the output message queue is not empty. It uses a list of HOST-IDs and LINKs provided by the conference CHAIR to generate the HOST-IMP leaders. One copy of the message is sent to the network through a separate input/output processor each time this process is run. When the last HOST entry is used the message is removed from the output queue and discarded.

Since present protocols require transmission of one copy of each data message to each HOST in a network voice conference, the time to deliver these copies to the network becomes a factor in the total delay. This is particularly true at HOSTs such as CHI which have VDH connections to their IMP. The serial transfer time over our VDH connection adds about 20 milliseconds to the delay for each HOST the message is sent to. As a small measure in reducing this delay, we short-circuit the normal transfer of messages to the IMP if their destination is our own HOST. Messages for our HOST are delivered directly to our input message processor, eliminating both the network delay for the local copy and the addition of 20 milliseconds of delay to copies to other HOSTs.

Conference Control

The flow of messages from our transmitter is gated by the local conference controller, which acts under orders from the CHAIR to allow message preparation and determine which voice terminal is connected to the vocoder. It is also necessary for the LCC to initialize the output list when a local speaker is selected. The skipped parcels flag is set. This provides information for the receiver to adjust its delay for the new speaker and guarantee that all parameters will be provided for the first parcel in the initial message.

When a speaker's turn is ended by a message from the CHAIR, no further data messages can be sent to the network. Hence it is not possible to insure that the last parcel sent will have all parameters present, which would guarantee that parameter interpolation could be performed by the receiver where needed. This situation must be handled by the receiver.

Control of the Transmitter

There are several parameters which are available to adjust the performance of the transmitter. Proper adjustment of these parameters can help compensate for poor network response and noisy environments or take advantage of good performance to get better speech quality. Controls are provided in the CHI network voice system to permit control of these parameters dynamically from a system keyboard:

a. The silence threshold and the number of frames below this threshold before silence is declared can be changed. Raising the threshold allows operation in a noisier environment without background noise causing transmission when the speaker is silent.

b. The test value used for the likelihood ratio test for transmission of reflection coefficients can be raised to decrease the number of bits transmitted or lowered to obtain better quality.

c. The minimum number of parcels to be packed into each message can be lowered to shorten delays due to message loading or raised to make more efficient use of the network.

In addition, the bit count for each message is accumulated and read each second, then smoothed to provide a transmission rate indicator. This rate is displayed as a plot against time continuously and can be printed at the terminal on request.

IV. LPC-II RECEIVER

The receiver's function is to accept data messages arriving from the transmitter over the ARPANET and generate a synthetic audio signal which is played out to local listeners at their voice terminals. The receiver must attempt to produce an output signal despite the partitioning of parcels into messages and the variations in the time these messages take to reach it from the transmitter. It must properly interpret the variable frame rate data to provide the parameters actually used for synthesis. It must recognize silent intervals when no messages are received. In meeting these requirements, the receiver attempts to add as little delay as possible to the end to end delays in the network voice communication.

The receiver is part of the same system as the transmitter, and must share processor and memory resources with it. It is also made up of a number of separate, cooperating processes, as illustrated in figure 2.

Data Message Input

Each message as it arrives from the network is classified as data only if it arrives on the specified data LINK as specified by its IMP/HOST leader. Only those data messages from the current speaker, as shown by their HOST-ID, STREAM and EXTENSION fields, are retained. All other data messages are discarded. Messages from the current speaker are inserted in an input queue for the LPC synthesizer. The messages are kept in time stamp order, even though they are usually sent as uncontrolled network messages and may arrive out of order.

Message Selection

An attempt is made to select a new message for synthesis whenever there is not enough information available in the previous message to prepare a set of parameters for the synthesizer. Since not all parameters are transmitted in each parcel, and linear interpolation is the preferred method for filling in missing parameters, there may be up to ten parcels of parameters left in the current message when a new message is needed. (If the next ten parcels do not have a given parameter, the previous value is used rather than interpolating.)

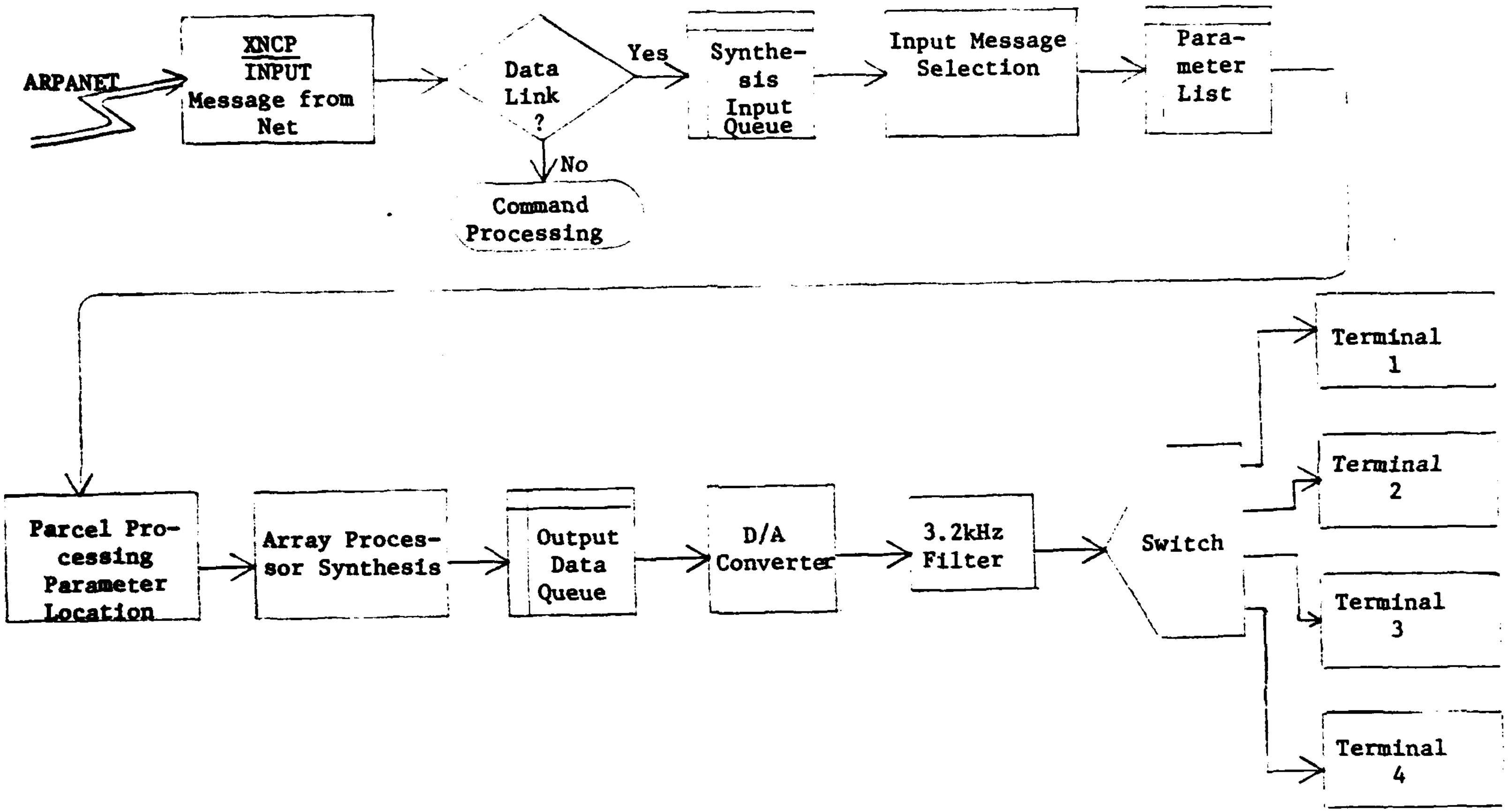

Figure 2: Receiver Data Flow

In order to allow for variations in network performance or the number of parcels in a message, both of which can affect the time between messages, it is necessary to establish a buffering interval, or delay, before processing the first message after a break in transmission because of silence or a change in speaker. The length of the delay should be sufficient to assure that a message will be available when needed.

It is possible to choose a delay equal to the sum of the expected variation in the network performance, the maximum difference in parcels per message and an allowance for the need to interpolate using values in the next message to determine parameters for parcels in the previous message. This delay can be added to the arrival time of the first mesage to give the time when that message is to be used. The disadvantage of this approach is that the maximum expected variation must be used for the network performance factor since it is not known whether the first message arrived quickly or slowly. Also, since the variation in arrival time of the first message is reflected in the time when the message is played out, the length of silence periods is not well preserved.

An alternative method for determining the delay before playing out a message has been developed which permits some reduction in the delay and in the variation in silence intervals. An expected network travel time (NT) is computed by smoothing the difference between the arrival time of a message and the time it was sent (Time Stamp + Parcel Count). This time actually includes the difference in time frames of the transmitter and receiver. NT is added to the time stamp of each message to give the expected arrival time of a message with no parcels. To this time we add a delay (D) which is the sum of the maximum expected number of parcels/message (25 - 40), the number of parcels from one message which may depend on the next (10) and the expected variation about the network time. This gives the time when the message should be processed:

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

The value of NT is adjusted each time a message is selected for synthesis using the observed time for that message. The delay time D could be adjusted to reflect network performance variations. At present, the CHI system permits setting the value for D dynamically but otherwise holds it fixed.

If the first message on the queue contains parcels immediately following those in the previous message, it is selected for synthesis without delay. If there are no messages on the queue, then either the speaker's transmitter has stopped sending messages or the message is delayed or lost. As long as there is a backlog of frames of synthesized data waiting to be played out, more parameters are not needed and no special processing is needed. When the backlog of frames for output is exhausted (and the last frame is being played out), any remaining parcels are used to prepare parameters for synthesis of frames, with missing parameters filled in by using the most recent values that have been received. When all parcels have been used, and all frames are output, it is assumed that the transmitter is silent and all zero frames are played out.

If the first message on the input queue has a time stamp greater than expected, it may have arrived out of order, messages may have been lost or it may be the first message after silence. In any case, the desired playout time is computed from its time stamp, NT and D. If the time has not arrived, the message is not processed. While waiting for the playout time, any remaining parcels from the previous message are used or all zero frames are played out as described above.

If the variations in network performance exceed that expected, a message may arrive out of order and delayed enough that the following message has already been processed. In this case, the time stamp of the new message will be less than that of the expected message. The new message must be discarded. A diagnostic is printed at the terminal in this case.

Once a message is selected, its parcels are unpacked into individual parameters and parcel header codes. Each parcel occupies a fixed length buffer within a parcel list with any residual parcels from the previous message at the top of the list.

Parcel Processing and Parameter Preparation

Parameters are prepared for synthesis processing one frame at a time. The array processor performs the parameter interpolation and synthesis filtering. It maintains the beginning of frame parameters from the previous frame. The main processor provides a set of interpolation weights and transmitted parameters. The array processor then interpolates between these parameters and the beginning of frame parameters to obtain the end of frame parameters.

The location of the next set of transmitted parameters for each of pitch, gain and reflection coefficients (Ks) is performed by the MP. The location of the parameters and determination of the proper interpolation coefficients is a three-step process. First, the header codes of all parcels in the buffer are

OR'ed. If the result is seven, or if there are ten or more parcels left, there is enough information to determine a set of parameters. If not, an attempt is made to get more parcels from the next message as described earlier. Next, the list of parcels is searched for the first occurrence of each type of parameter. The number of parcels passed by is used to look up the interpolation coefficient. If this distance is greater than nine, the coefficient is 0, causing the old value to be used. If the distance is 0, the coefficient is 1, and the new value will be used. The third step prevents interpolation between voiced and unvoiced parcels. The list of parcels is again searched, but this time starting with the parcel containing the parameter and working backwards to the first occurrence of a pitch parameter. If this pitch does not have the same voicing as the beginning of frame parameters, the interpolation value is set to 0 unless it is 1. The pitch parameter is never interpolated since it is transmitted whenever it changes.

Effect of Missing Parcels

When there are missing parcels, as occurs during silence or when speakers change the parameters remaining in the AP will not be related to the first parcel parameters of the next message. The old parameters are cleared by sending an extra set of parameters to the AP for synthesis when silence is recognized. These parameters are all 0 with interpolation values of 1. The data generated in this case gives the final transition into silence. When the new parameters arrive, the first frame synthesized will provide a proper transition out of silence.

If the first parcel in the first message does not contain all parameters, the receiver will not be able to process it. These parcels can be replaced by additional silence until all parameters have been found. This problem will not normally arise if all transmitters force the first parcel to be complete whenever the skipped parcels bit is set in the message header. It can still occur if messages are lost or very late.

REFERENCES

1. Cohen, D., Issues in Transnet Packetized Voice Communication, NSC Note 95, USC Information Sciences Institute, Marina del Rey, California, July 1976.

2. Culler, G. J., McCammon, M., Network Voice Transmission, Quarterly Technical Report, Speech Compression Research at CHI, Culler/Harrison, Inc., Goleta, California, July 1976.

3. Kimbleton, S. R., Schneider, G. M., "Computer Communications Networks: Approaches, Objectives, and Performance Considerations," ACM Computing Surveys, 7, 3 (September 1975), 129-173.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

14

1. REPORT NUMBER: CHI-QTR-203

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

6

4. TITLE (and Subtitle): A Variable Frame Rate Network Voice System, [redacted] Speech Compression Research at CHI.

9

5. TYPE OF REPORT & PERIOD COVERED: Quarterly Technical Report. June [redacted] — Aug[redacted]76,

6. PERFORMING ORG. REPORT NUMBER

10

7. AUTHOR(s): Michael/McCammon, Glen J./Culler

15

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 93017

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

11

12. REPORT DATE: Sep[redacted]76

13. NUMBER OF PAGES: 18

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

12 21p.

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)

LPC Vocoding
Variable Frame Rate
Network Conferencing
Speech Compression
Network Voice Protocols

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

A working computer system for transmission and receiving of speech messages over the ARPANET using variable frame rate linear predictive coding is described. This system conforms to the LPC-II protocols. A detailed discussion is given of the known problems and pitfalls in speech transmission. A description is given of the methods used to deal with these problems.

2

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

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

409276 LB