AD/A-003 478

NATURAL COMMUNICATION WITH COMPUTERS. VOLUME II. SPEECH COMPRESSION RESEARCH AT BBN

John Makhoul, et al

Bolt Beranek and Newman, Incorporated

Prepared for:

Advanced Research Projects Agency

December 1974

BOLT BERANEK AND NEWMAN INC

CONSULTING · DEVELOPMENT · RESEARCH

BBN Report No. 2976 December 1974

NATURAL COMMUNICATION WITH COMPUTERS

Final Report - Volume II

Speech Compression Research at BBN

October 1970 - December 1974

Principal Investigator

Dr. William R. Sutherland
(617) 491-1850

Project Scientist

Dr. John I. Makhoul
(617) 491-1850

This research was supported by the Advanced Research Projects Agency under ARPA Order no. 1697; Contract no. DAHC15-71-C-0088.

This report is one of five volumes which compose the final report of work performed over a four year period by Bolt Beranek and Newman Inc. under contract DAHC15-71-C-0088, Natural Communications with Computers. This work was supported by the Defense Advanced Research Projects Agency under ARPA order number 1697. Because of the wide spectrum of research activites performed, the final report has been structured as follows:

Title	Volume
Speech Understanding Research at BBN	I
Speech Compression at BBN	II
Distributed Computation Research at BBN	III
ARPANET TENEX	IV
INTERLISP Development and Automatic Programming	V

BBN Report No. 2976 December 1974

NATURAL COMMUNICATION WITH COMPUTERS

Final Report - Volume II

Speech Compression Research at BBN

October 1970 - December 1974

Principal Investigator

Dr. William R. Sutherland
(617) 491-1850

Project Scientist

Dr. John I. Makhoul
(617) 491-1850

This research was supported by the Advanced Research Projects Agency under ARPA Order no. 1697; Contract no. DAHC15-71-C-0088.

Unclassified
Security Classification

AD/A-003478

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author)
Bolt Beranek and Newman Inc.
50 Moulton Street
Cambridge, Mass. 02138

2a. REPORT SECURITY CLASSIFICATION
Unclassified

2b. GROUP

3. REPORT TITLE

NATURAL COMMUNICATIONS WITH COMPUTERS; Final Report - Volume II
Speech Compression Research at BBN

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)
Final Report; October 1970 - December 1974

5. AUTHOR(S) (First name, middle initial, last name)

John Makhoul, R. Viswanathan, Lynn Cosell, and William Russell

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
December 1974	104	33

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
DAHC15-71-C-0088	BBN Report 2976
b. PROJECT NO. ARPA ON 1697	
c.	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
d.	

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

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
This research was sponsored by the Advanced Research Projects Agency under ARPA Order No. 1697	

13. ABSTRACT

This report describes our work in developing a linear predictive speech compression system that transmits high quality speech at low bit rates. We have developed several methods for reducing the redundancy in the speech signal without sacrificing speech quality. Included among these methods are preemphasis of the incoming speech signal, adaptive optimal selection of predictor order, optimal selection and quantization of transmission parameters, variable frame rate transmission, optimal encoding, and improved synthesis methodology. When we incorporated all of these in a floating point simulation of a pitch-excited linear predictive vocoder, we obtained synthesized speech with high quality at average transmission rates as low as 1500 bps.

DD FORM 1 NOV 68 1473 REPLACES DD FORM 1473, 1 JAN 68, WHICH IS OBSOLETE FOR ARMY USE.

Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Speech Compression						
Speech Analysis-Synthesis						
Vocoders						
Linear Predictive Vocoders						
Linear Prediction						
Preemphasis						
Optimal Linear Predictor Order						
Variable Order Linear Prediction						
Reflection Coefficients						
Partial Correlation						
Log Area Ratios						
Quantization						
Encoding						
Huffman Coding						
Variable Rate Transmission						
Spectral Sensitivity Analysis						

PROJECT PERSONNEL

John Makhoul	Project Leader
R. Viswanathan	Co-Project Leader
Lynn Cosell	Research Engineer
William Russell	Research Engineer
Connie Williams	Project Secretary

ABSTRACT

This report describes our work in developing a linear predictive speech compression system that transmits high quality speech at low bit rates. We have developed several methods for reducing the redundancy in the speech signal without sacrificing speech quality. Briefly, preemphasis of speech was used to reduce its spectral dynamic range and thereby improve the accuracy of parameter quantization. The optimal order of the linear predictor was adaptively determined for every speech frame as the lowest value that adequately represents the speech signal. Among several equivalent sets of predictor parameters that were investigated, the reflection coefficients were judged to be the best for use as transmission parameters. An optimal procedure for quantizing the reflection coefficients was developed by minimizing the maximum spectral error due to quantization. The latter criterion was found to yield synthesized speech with maximum quality for a given average transmission rate. A scheme was used to transmit speech parameters at variable rates in accordance with the changing characteristics of the incoming speech. An information theoretic method was used to encode the quantized transmission parameters at significantly lower bit rates, and with absolutely no effect on speech quality. Finally, with the time-synchronous method of analysis, improved speech quality was obtained when synthesis was also done time synchronously. In addition to these major results, numerous other minor results of relevance to the stated goal were also obtained. As a combined result of all these findings, we obtained high quality speech at average transmission rates as low as 1500 bits per second.

TABLE OF CONTENTS

	Page
I. INTRODUCTION	1
A. Summary of Major Results	1
1. The Minimax Principle	2
2. Quantization	3
3. Encoding	5
4. Synthesis	5
B. Outline of Report	6
II. SPEECH COMPRESSION USING LINEAR PREDICTION	9
A. Components of a Speech Compression System	9
B. Linear Prediction of the Speech Signal	12
III. ANALYSIS	19
A. Parameter Extraction	19
B. Timing of Parameter Extraction	21
IV. VARIABLE ORDER LINEAR PREDICTION	22
V. CHOICE OF PARAMETERS FOR QUANTIZATION	30
A. Preprocessing of Speech	30
1. Preemphasis	31
2. Bandwidth Expansion Method	37
B. Quantization of Pitch and Gain	38
C. Choice of Filter Parameters	38
VI. OPTIMAL QUANTIZATION OF REFLECTION COEFFICIENTS	46
A. Sensitivity Analysis	47
B. Optimal Quantization	49
C. Optimal Bit Allocation	55
D. Comments on Another Spectral Sensitivity Measure	58
VII. VARIABLE FRAME RATE TRANSMISSION	62
VIII. VARIABLE WORDLENGTH ENCODING	67
A. Huffman Coding	67
B. Delta Encoding	72
C. Statistics for Huffman Coding and Bit Savings	73

Table of Contents (continued)

	Page
IX. SYNTHESIS	79
A. Excitation	79
B. Transfer Function	81
C. Parameter Setting and Interpolation	81
1. Time-Synchronous Versus Pitch Synchronous Synthesis	82
2. Interpolation Study	84
X. SIMULATION OF SPEECH COMPRESSION SYSTEM	86
A. Software Simulation	86
B. Typical Transmission Rates	87
XI. REAL TIME IMPLEMENTATION	90
A. Signal Processing System	90
B. Variable Speed Speech	92
XII. MISCELLANEOUS TOPICS	94
A. Measures for Objective Evaluation of Speech Quality	94
B. Variable Sampling Frequencies	98
C. Formant Bandwidth Correction	98
D. Parameter Smoothing	99
XIII. CONCLUSIONS	101
REFERENCES	102

SPEECH COMPRESSION

I. INTRODUCTION

In order to use the ARPA Network for voice transmission, a speech compression system which achieves high quality speech at low transmission rates is needed. During the past two years our research in various aspects of linear predictive speech compression systems (also known as linear predictive vocoders) has yielded numerous analytical and experimental results which can be applied to increasing both the quality of the speech and the efficiency of parameter transmission.

We have developed a time-asynchronous digital vocoder in which the transmission rate varies according to the properties of the incoming speech signal. The variable transmission rate has a low upper bound as well as a low average, an important consideration for a real-time application such as transmission over the ARPA Network.

A. Summary of Major Results

The objective of a speech compression system is to reduce the redundancy present in the speech signal as much as possible while maintaining good quality in the synthesized speech. The use of the linear prediction method for modeling speech already provides a major step towards meeting this objective. Our project has been directed towards significant implementation

aspects of the linear predictive vocoder that can result in further compression with limited distortion in speech quality. We have collected statistics about the parameters of the linear prediction model by analyzing speech utterances from male and female speakers. These statistics were used in the development as well as in the implementation of several compression schemes. The major results in our project are summarized below under the headings of quantization, encoding and synthesis. First, however, we state the guiding principle that we have used in linking speech quality to transmission rate.

1. The Minimax Principle

We have developed a systematic and objective design criterion which, in our experience, leads to synthesized speech with high quality for a given average transmission rate. The criterion is to minimize the maximum spectral error in the synthesized speech. This minimax principle has been used consistently in our research and is basically responsible for the high quality output of our low bit rate systems.

2. Quantization

Quantization is the major source of bit rate reduction. We distinguish three types of quantization: parameter quantization, predictor order quantization, and time quantization. Below is a summary of the results in these areas.

(a) Parameter Quantization

(i) We found that reducing the spectral dynamic range of the input speech improves quantization accuracy, regardless of which set of parameters is chosen for quantization. We proposed methods for the reduction of the dynamic range by preprocessing of the speech signal.

(ii) From a comparative study of the quantization properties of a number of parameter sets, we concluded that the reflection coefficients are the best set for use as transmission parameters.

(iii) We determined an optimal method for the quantization of the reflection coefficients by making use of the minimax principle. The optimal procedure consists of first transforming the reflection coefficients to log area ratios and then linearly quantizing these transformed parameters.

(iv) An optimal solution was also derived for the problem of allocating a fixed number of bits among the parameters.

(b) Variable Order Predictor (Order Quantization)

We have found that different speech sounds can be represented adequately by different order linear predictors. Thus, rather than sending a maximum number of parameters for every frame, one can minimize the bit rate by sending the minimum number of parameters that adequately represent that frame. We have introduced an information theoretic criterion that allows us to determine the optimal order for each analysis frame.

(c) Variable Frame Rate (Time Quantization)

In deciding how often to transmit parameters, the application of the minimax principle leads to the obvious result that one should transmit more often when the speech spectrum is changing rapidly and less often when the spectrum is changing slowly. In using this transmission scheme, we have employed an effective criterion to measure spectral changes. We have found that, for a given average bit rate, variable frame rate transmission produces distinctly superior quality speech than fixed frame rate transmission.

3. Encoding

We have collected appropriate statistics on the distribution of quantized values of the transmission parameters. These statistics were used to develop variable length bit encoding techniques that result in considerable bit savings (on the order of 20%). These are information theoretic techniques which have absolutely no effect on speech quality.

4. Synthesis

Proper methodology in synthesis is crucial in producing high quality speech. We have found that time-synchronous updating of linear prediction parameters at the synthesizer yields better speech quality than pitch-synchronous updating if the analysis is performed time synchronously. This method has the additional advantage of simplifying the necessary computations.

Using the results summarized above and others discussed in this report, we were able to demonstrate good quality speech at average rates of 1500 bps (bits/sec) or less. We consider this a significant step towards our goal of developing a high quality, low bit-rate linear predictive vocoder.

B. Outline of Report

In Section II, the various components of a speech compression system are introduced. A brief introduction to linear prediction of speech is also given.

In Section III, we discuss the extraction of predictor parameters, pitch, and gain. In conjunction with this discussion two comparisons are made: the autocorrelation versus the covariance method, and time-synchronous versus pitch-synchronous analysis.

The variable order linear prediction method is outlined in Section IV. The reason for varying the predictor order is given first, followed by a discussion of an information theoretic criterion to determine the "optimal" order for any analysis frame.

In Section V, two methods are given for preprocessing of speech which reduce its short-time spectral dynamic range and hence improve parameter quantization accuracy. Logarithmic quantization of pitch and gain is dealt with next. The remainder of the section presents the results of a comparative study of the quantization properties of several alternate sets of parameters which uniquely characterize the linear predictor. Specifically, it is concluded that the reflection coefficients are to be preferred over all other sets of parameters for purposes of quantization.

The problem of optimal quantization of the reflection coefficients is considered in Section VI. It is argued first that for good speech quality it is necessary to minimize the maximum error in the spectrum of the linear predictor due to parameter quantization. This naturally leads to the investigation of the sensitivity of the spectrum to changes in the values of the reflection coefficients. Using the minimax error criterion and the results of the sensitivity analysis, it is shown that the optimal quantization method consists of transforming the reflection coefficients to log area ratios and linearly quantizing the latter. An optimal bit allocation strategy for the transmission parameters is also presented. Use of an alternate sensitivity analysis of the reflection coefficients is then investigated.

Variable frame rate transmission is discussed in Section VII as a means of significantly lowering the bit rate while maintaining high speech quality.

In Section VIII, we present an information theoretic procedure, known as Huffman coding, which uses the statistics of the quantized values of a parameter to transmit them more efficiently with a variable number of bits and, very importantly, without introducing any additional error. This encoding method offers considerable savings in bit rates.

Issues relating to the synthesizer are discussed in Section IX. An important result presented in this section is that time-synchronous synthesis produces better quality speech than pitch-synchronous synthesis.

Section X briefly narrates the software simulation of the entire speech compression system on our time-sharing computer facility. Also included in this section are the typical average transmission rates encountered with the use of one or many of the bit-saving techniques presented in the earlier sections.

Section XI summarizes our work completed thus far towards implementing the speech compression system in real time in cooperation with the other sites in the ARPA community.

A few other topics that we have also worked on during this project are considered in Section XII. These include status of our research in developing measures for objective speech quality evaluation, a new method of testing the performance of the vocoder at different sampling frequencies without actually sampling at all those rates, and our experience with the two techniques: formant bandwidth correction before synthesis and parameter smoothing.

II. SPEECH COMPRESSION USING LINEAR PREDICTION

A. Components of a Speech Compression System

Figure 1 shows the various components of a speech compression system. The first component analyzes the speech signal s(t) that has been low-pass filtered and time-sampled, and extracts a vector of unquantized parameters $\underline{x}(t)$. These parameters are then quantized and encoded in the encoder as $\underline{y}(t)$ and are transmitted through the transmission channel. In a noiseless channel $\underline{y}'(t)=\underline{y}(t)$. This is generally the case for the ARPA Network. The parameters $\underline{y}'(t)$ are decoded in the decoder to produce an estimate $\underline{x}'(t)$ of the analysis parameters $\underline{x}(t)$. The last component in Fig. 1 uses the parameters $\underline{x}(t)$ to synthesize the signal s'(t) which is an approximation to the original signal s(t). A compression system attempts to minimize the number of bits/second in $\underline{y}(t)$ while maintaining good quality in the synthesized speech. The nature of the synthesizer dictates the type of analysis to be performed. Fig. 2 depicts the two major components of the synthesizer: excitation and transfer function. In our project, we have done work on each of the different parts of the vocoder shown in Fig. 1.

Once a speech model is chosen (the linear prediction model in our case), any reduction in transmission rate is accomplished by the encoder. The encoder in Fig. 1 performs the two functions, quantization and encoding. The quantization process converts the extracted parameters into a set of integers

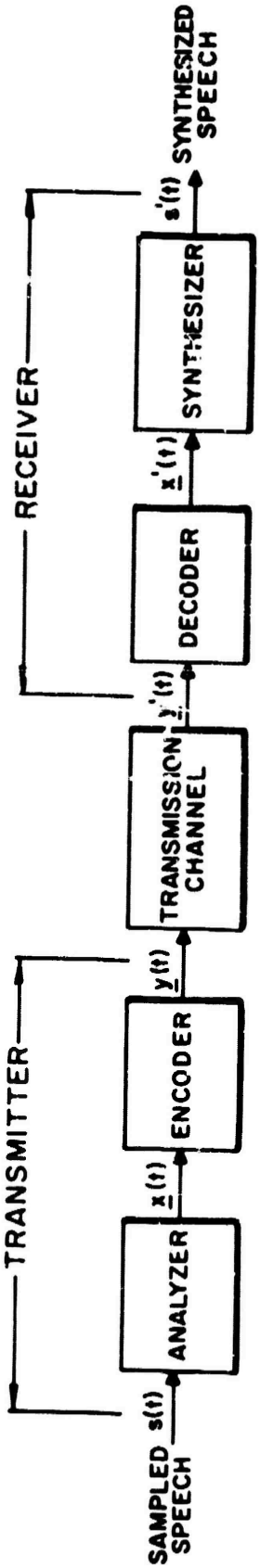


Fig. 1. Components of a Speech Compression System

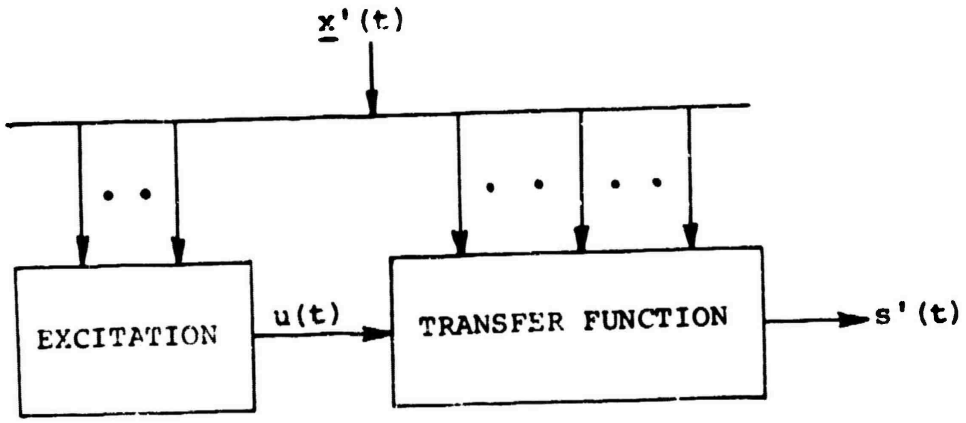


Fig. 2. Major components of a speech synthesizer.

using specified quantization schemes. The encoding process encodes these integers into a sequence of binary digits for transmission. The encoding can be as simple as direct binary encoding, or as complicated as desired for the minimization of the average transmission rate.

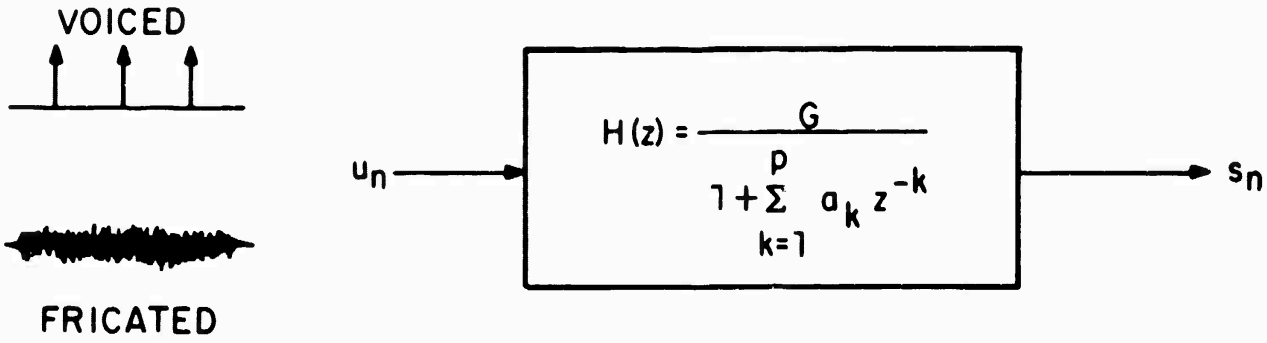
The linear prediction method models speech by a time varying all-pole filter. The filter parameters are assumed to vary slowly enough so that they can be considered constant over an analysis frame, usually 10-20 msec long. Next, we briefly review the linear prediction method and provide necessary background for later sections.

B. Linear Prediction of the Speech Signal

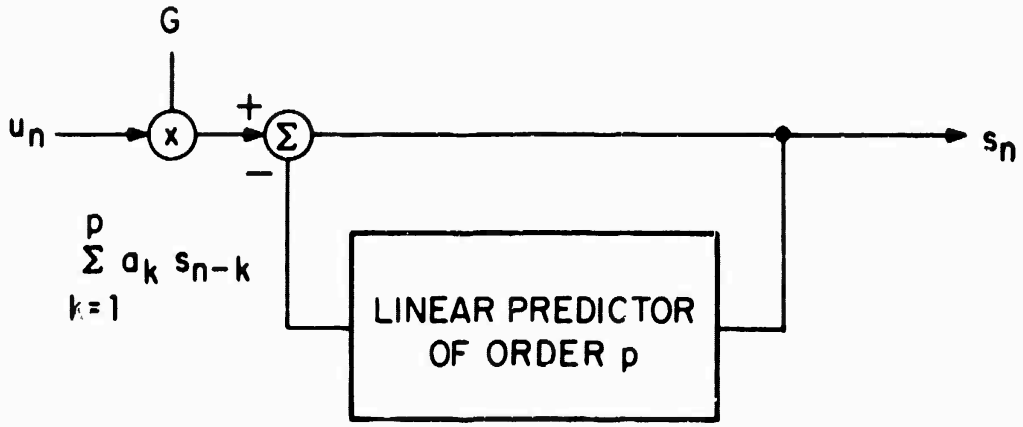
In linear prediction, speech is modeled by an all-pole filter

$$H(z) = \frac{G}{1 + \sum_{k=1}^{p} a_k z^{-k}} \qquad (1)$$

as shown in Fig. 3. The parameters a_k, $1 \leq k \leq p$, are known as the predictor coefficients, and G is the filter gain. The input to the filter is either a sequence of pulses separated by the pitch period for voiced sounds, or white noise for fricated (or unvoiced) sounds. For a particular speech segment the filter parameters are obtained by passing the sampled speech signal through the inverse filter



(a) FREQUENCY-DOMAIN MODEL



(b) TIME-DOMAIN MODEL

Fig. 3. Discrete model of speech production as employed in linear prediction.

$$A(z) = 1 + \sum_{k=1}^{p} a_k z^{-k} \tag{2}$$

as in Fig. 4, and then minimizing the total-squared predictor error

$$E = \sum_n e_n^2 = \sum_n (s_n + \sum_{k=1}^{p} a_k s_{n-k})^2 . \tag{3}$$

with respect to a_k. Depending on the range over which the summation in (3) applies and the definition of the signal s_n in that range, we have the two linear predictive methods of analysis: the covariance method and the autocorrelation method. For the covariance method, the signal is defined over a finite range, $-p \leq n \leq N-1$, and the minimization of E leads to the following normal equations [1,2]:

$$\sum_{k=1}^{p} a_k C_{ik} = -C_{i0} \quad , \quad 1 \leq i \leq p \quad , \tag{4}$$

where

$$C_{ik} = \sum_{n=0}^{N-1} s_{n-i} s_{n-k} \quad . \tag{5}$$

The pxp coefficient matrix $[C_{ij}]$ of the system of equations in (4) is symmetric and positive definite, and is called the covariance matrix. The covariance normal equations can be solved by an efficient triangularization method [3]. Using (4) and (3), the minimum prediction error is given by [2]

$$E_p = C_{00} + \sum_{k=1}^{p} a_k C_{0k} \quad . \tag{6}$$

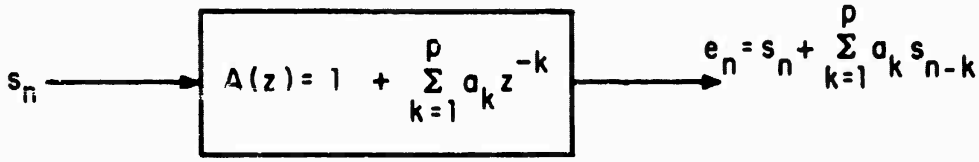


Fig. 4. The error sequence e_n as the output of an inverse filter A(z).

For the autocorrelation method, the signal s_n is assumed to be defined over the infinite interval, $-\infty<n<\infty$. Usually, the signal is multiplied by a finite window (e.g. Hamming) so that $s_n=0$ for $n<0$ and $n>N-1$. The normal equations for the autocorrelation method can be shown to be [2,4,6,7]

$$\sum_{k=1}^{p} a_k R_{|i-k|} = -R_i \quad , \quad 1\leq i\leq p \quad , \tag{7}$$

where

$$R_i = \sum_{n=0}^{N-|i|} s_n s_{n+|i|} \tag{8}$$

is the autocorrelation function of the windowed signal s_n. The autocorrelation matrix $[R_{i-j}]$ is symmetric, positive definite and Toeplitz (the values along any diagonal are equal). Levinson's method can be used to recursively solve the autocorrelation equations [2,4,5]. The minimum prediction error is obtained by substituting (7) in (3) as

$$E_p = R_o + \sum_{k=1}^{p} a_k R_k \quad . \tag{9}$$

When applying Levinson's recursive method to solve (7), we also obtain the auxiliary quantities, k_i, $1\leq i\leq p$, which are called the reflection coefficients [10,23] (or partial correlation coefficients [8,9]). Reflection coefficients occur naturally in the treatment of the vocal tract as a lossless acoustic tube with p sections, each with a different cross-sectional area [1,10]. The filter H(z) in (1) is stable (i.e., poles of H(z) lie inside the unit circle in the z plane)

if and only if

$$-1 < k_i < 1 \quad , \quad 1 \leq i \leq p \quad . \tag{10}$$

Of interest also is the normalized error V_p which is the ratio of the minimum error to the energy of the input speech signal, i.e.

$$V_p = E_p / R_0 \quad . \tag{11}$$

For the autocorrelation method, V_p can be expressed in terms of the reflection coefficients as [2]

$$V_p = \prod_{j=1}^{p} (1-k_j^2) \quad . \tag{12}$$

There exist two methods of synthesizing speech using the analysis parameters. First, the prediction error signal e_n (or a simple transformation thereof) is used as input to an all-pole filter $1/A(z)$ to produce the synthesized speech. A vocoder using this synthesis approach is called a residual excited vocoder [11,12]. As this vocoder needs the transmission of the error signal at the speech sampling rate, the transmission rate for acceptable speech quality is relatively high (on the order of 10,000 bps). As our goal was to develop a low bit-rate vocoder (less than 2000 bps), we did not consider the residual excited vocoder in our research.

The second method of synthesis uses the pulse/noise excitation as input to the all-pole filter $H(z)$ (see Fig. 3). A vocoder using this synthesis approach is called a pitch excited

vocoder. For each analysis frame, a decision has to be made as to whether that frame is voiced or unvoiced, and if voiced, the value of the pitch period has to be determined. In addition to the voicing information, the gain G of the filter H(z) has to be determined. By equating the energy of the synthesized speech to the energy of the original speech, G can be shown to be related to the minimum prediction error by [7,8,14]

$$G^2 = E_p = R_o V_p = R_o + \sum_{k=1}^{p} a_k R_k \quad . \tag{13}$$

For the pitch excited vocoder, the transmitter sends the voicing information and the gain at the same low rate as the filter parameters. As a result, transmission rates of about 2000 bps produce acceptable speech quality. In our research, we worked exclusively with the pitch excited vocoder. However, all the results stated in this report that pertain to the transmission of filter parameters also hold for the residual excited vocoder.

III. ANALYSIS

The analyzer in Fig. 1 extracts from the speech signal the excitation and the transfer function parameters that are used later for synthesis. We shall first discuss the extraction of these parameters and then the timing of such extraction.

A. Parameter Extraction

For the types of synthesizer implementation we have considered, the transfer function parameters can be computed directly from the linear prediction coefficients. A comparison of the two methods of linear prediction indicates that the predictor coefficients obtained in the autocorrelation method are guaranteed to produce a stable filter of the form given in (1) [15,16], while stability cannot be guaranteed in the covariance method [1]. Computationally, the autocorrelation equations (7) can be solved faster than the covariance equations (4). Also, the autocorrelation method offers many useful spectral interpretations [7]. The covariance method may, however, produce a better representation of the speech signal. In our experience, the possible improvement in speech quality produced by the covariance method was not commensurate with the extra computational cost in solving (4) and in coping with instability problems. Consequently in all our work reported below, we used only the autocorrelation method.

From a study of some of the available pitch extraction schemes, a modified version of the method of center-clipping [17] was chosen for use in our experimental speech compression system. The reliability of the basic scheme was improved by using several additional decision parameters such as the normalized error associated with the linear predictor, zero crossing rate of the clipped signal and frame-to-frame energy change in the speech signal. Furthermore, to yield accurate pitch estimation over a wide range of frequencies, the width of the time window chosen for pitch analysis was made variable; it was changed from 30 msec to 50 msec whenever pitch frequency fell below 100 Hz. Equipped with these features, the center-clipping algorithm was found to yield pitch data which compared quite favorably with those derived manually from the time signals.

For computing the gain of the linear prediction filter at the synthesizer, we also computed and transmitted energy per sample of the input speech signal. Clearly, the energy of the Hamming-windowed speech signal is less than the energy of the unwindowed signal. From both analytical and experimental considerations, we found that multiplying the energy of the windowed speech signal by a factor of 2.5 provided a loudness level for the synthesized speech that was about the same as the original input speech.

B. Timing of Parameter Extraction

There are two considerations in the timing of parameter extraction. The first deals with frame positioning with respect to the pitch period. Although it is generally agreed that pitch-synchron analysis is desirable in terms of the quality of the synthesized speech, it is also clear that such analysis can be quite involved in terms of complexity of computation and decision making. In our experiments with pitch-synchronous analysis we found that the resulting improvement in speech quality was only minimal and not commensurate with the added complexity. We have therefore used pitch-asynchronous analysis exclusively.

The second consideration deals with the rate of parameter extraction. In all our investigations, parameter extraction was done at a constant frame rate. However, we studied both constant frame rate and variable frame rate transmission of parameters. In Section VII, where we discuss variable frame rate transmission, we give criteria to decide when to transmit based on the parameter data extracted at a constant frame rate.

IV. VARIABLE ORDER LINEAR PREDICTION

If a fixed order linear prediction is used for speech that is sampled at a Nyquist rate of 10 kHz, then an order p=12 has been found satisfactory for modeling of all the speech sounds. However, we found that for some sounds (especially unvoiced sounds), good spectral representation was obtained using a considerably lower order linear predictor. In fact, it is possible to adaptively vary the order p of the predictor in accordance with the properties of the speech sounds being analyzed. The purpose of using variable order linear prediction is to lower the transmission rate by transmitting on the average a smaller number of coefficients, without causing any perceptible change in speech quality.

It is desirable to have a criterion to determine the "optimal" (minimum) predictor order that gives an adequate spectral representation of each speech sound. The criterion should strike a compromise between the number of coefficients used and the modeling accuracy obtained. We have found an information theoretic criterion due to Akaike to be particularly suitable for this purpose [18].

Akaike has stated the modeling problem as an estimation problem with an associated error measure. For the maximum likelihood estimation method, he has shown that an estimate of the mean log-likelihood reduces to an information theoretic measure. When the estimates of model parameters are close to

their true values, this measure can be simplified as [19]:

$$I(p) = -2 \log (\text{maximum likelihood function}) + 2p \quad . \qquad (14)$$

The value of p for which I(p) is a minimum is taken to be the optimal order. The first term in (14) relates to modeling or estimation error, while the second term represents the model complexity. Hence, the criterion in (14) reflects a mathematical formulation of the principle of parsimony in model building. In our problem of all-pole modeling, if we assume that the speech signal has a Gaussian probability distribution, then (14) reduces to (neglecting additive constants and dividing by N_e)

$$I(p) = \log V_p + \frac{2p}{N_e} \quad , \qquad (15)$$

where V_p is the normalized error given in (11) and N_e is the "effective" number of samples in the analysis frame. The word "effective" is used to indicate that one must compensate for possible windowing. The effective width of a window can be taken as the energy under the window relative to that of a rectangular window. For example, we use a Hamming window for which $N_e=0.4N$, where N is the number of samples in the analysis frame.

Note that the first term in (15) decreases as a function of p, and the second term increases. Therefore, a minimum can occur. In practice, there are usually several local minima;

then the value of p corresponding to the absolute minimum of I(p) is taken as the optimal value. Usually I(p) is computed up to the maximum value of interest, and the minimum of I(p) is found in that region. We used a maximum value of p=13 for speech preemphasized with a single-zero filter. (For a discussion on preemphasis, see Section V-A.)

A property of the reflection coefficients, k_i, $1 \leq i \leq p$, is that the values of k_i, $i<p$, do not change as p is varied. So, when applying Akaike's criterion, we need only to compute the reflection coefficients for the maximum order predictor. For any pth order predictor, where p is less than the maximum value used, I(p) is computed from (15) with V_p obtained using the first p reflection coefficients in (12).

Figure 5 shows an example of the application of Akaike's criterion to a voiced sound. The dashed curve is a plot of the normalized error which decreases monotonically with increasing p. The solid curve is a plot of I(p) in (15) multiplied by 10 $\log_{10}e$ to obtain the results in decibels. In Fig. 5 the optimal predictor order is p=10. Note that I(p) for p>10 slopes upward, but very gently. This indicates that the actual absolute minimum is quite sensitive to the linear term in (15). Application of Akaike's criterion to a fricative sound is illustrated in Fig. 6. The optimal order for this case is 3 as shown at the top of Fig. 6. The bottom plot in Fig. 6 shows that the spectrum of the third order predictor (smooth plot) is a reasonable approximation to the envelope of the power spectrum

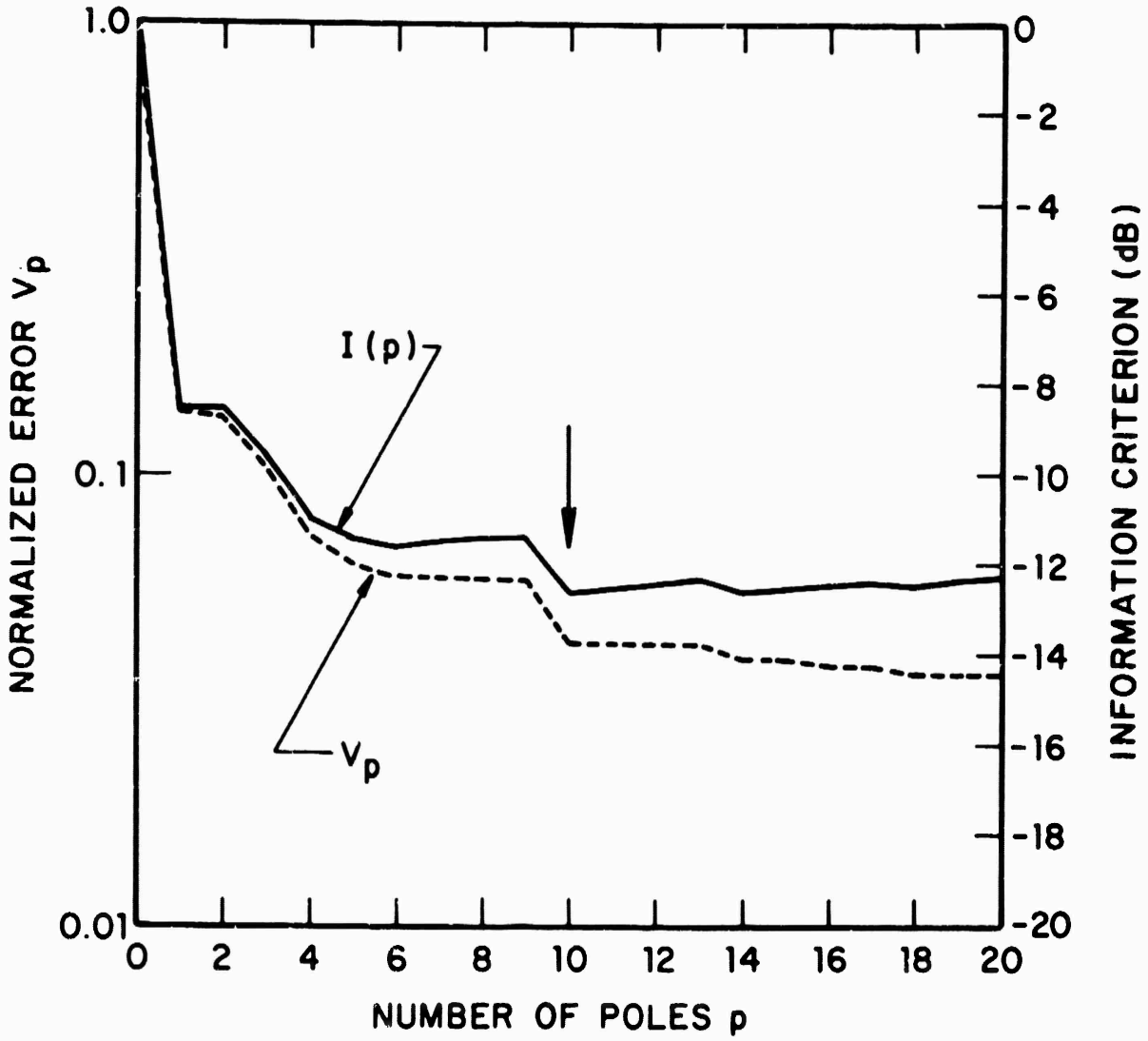
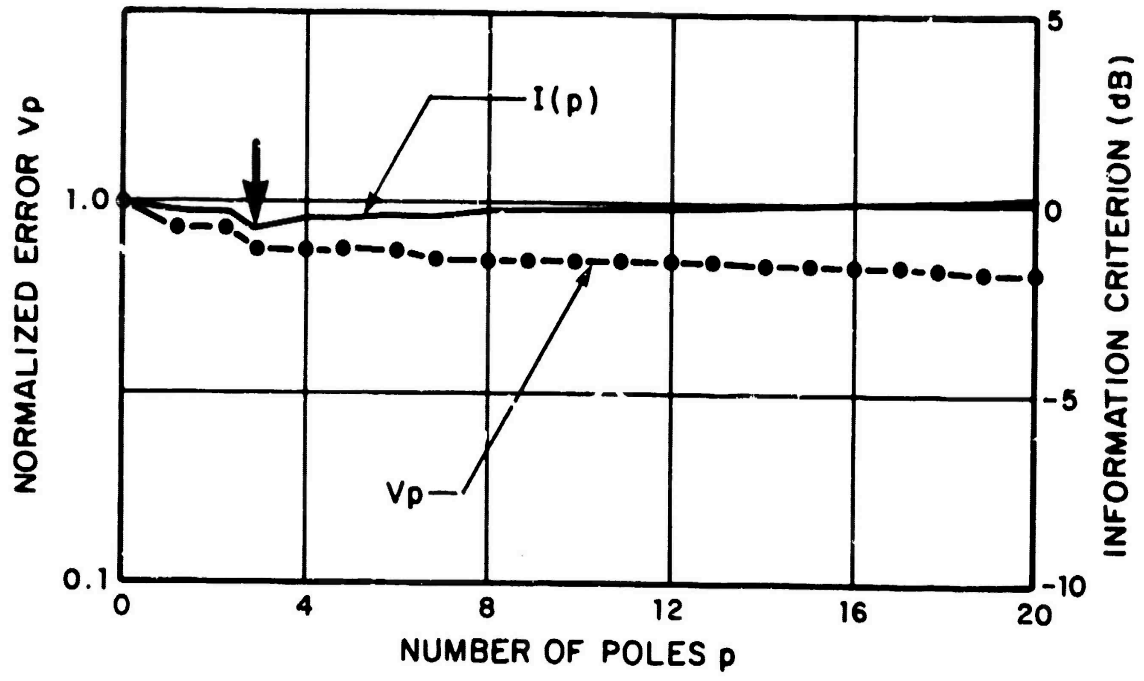


Fig. 5. A plot of Akaike's information criterion versus the order p of the predictor for a voiced sound. The optimal value of p occurs at the minimum of I(p), shown by the arrow at p=10.



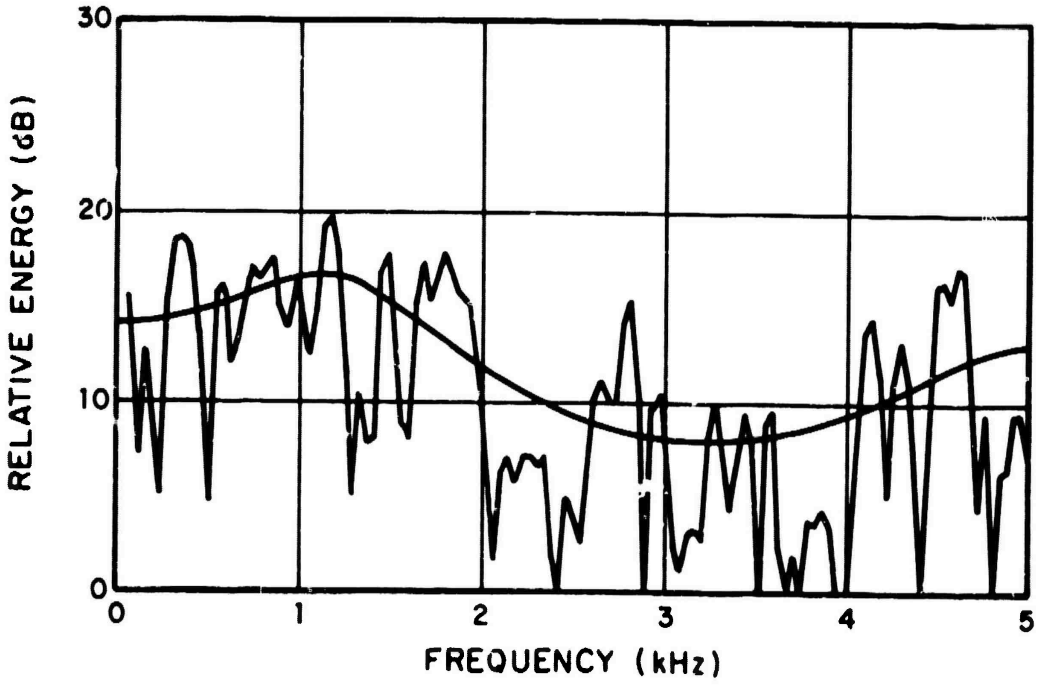


Fig. 6. Application of Akaike's information criterion to a fricative sound.

of the corresponding speech signal (ragged plot).

In practice, the criterion in (15) should not be regarded as an absolute, because it is based on several assumptions which might not apply for the signal of interest. For example, the Gaussian assumption might not hold. Therefore, the experimenter should feel free to adjust the criterion to suit one's application. One simple way of "tuning" the criterion is to multiply N_e by an appropriate factor. However, we have found that for speech compression applications the criterion (15) is by itself adequate without modification.

We collected statistics by applying the criterion (15) to preemphasized speech using a data base of several utterances from male and female speakers. The resulting histograms of the optimal order are given separately for voiced and unvoiced sounds in Fig. 7. In the plots, the ordinate at a given number of poles gives the probability of that number being chosen as the optimal order. From these probabilities, we found that the average value of the optimal order is 9.6 for voiced sounds and 5.2 for unvoiced sounds.

When applying variable order linear prediction to speech compression, we must also transmit a code indicating the number of poles used for every transmitted data frame. For the case when the maximum order is 13, and with the use of the Huffman coding procedure described in Section VIII, we can transmit this information with about 3.17 bits/frame on the average. In our

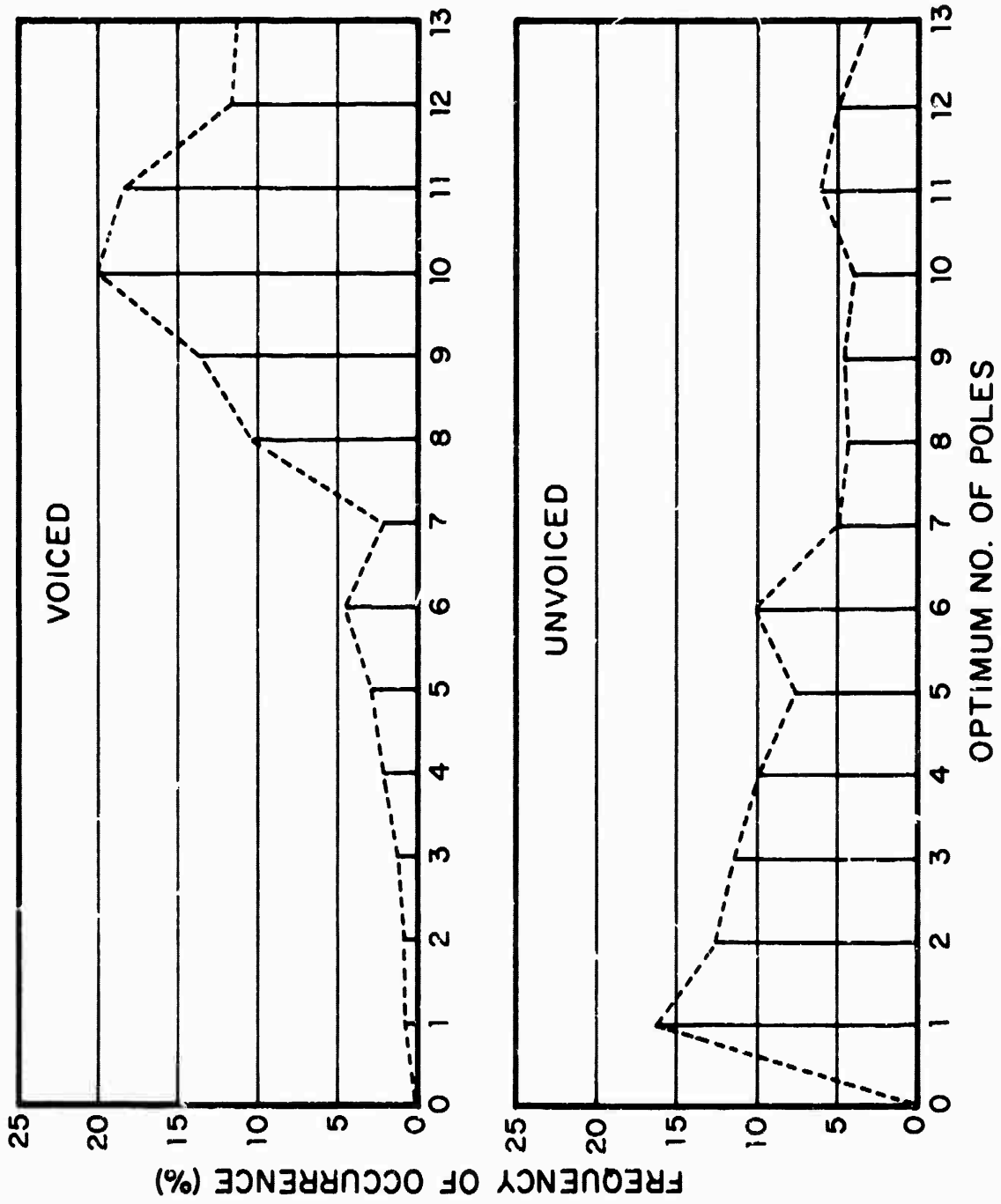


Fig. 7. Histograms of the optimal number of poles for voiced and unvoiced sounds.

experiments, we found that the average savings in transmission rates with the use of variable order linear prediction ranged between 10 to 15% . Informal listening tests on the synthesized speech did not indicate any perceptible difference between the fixed order and the variable order cases.

V. CHOICE OF PARAMETERS FOR QUANTIZATION

Proper choice of transmission parameters is important for reducing the bit rate while maintaining good quality speech at the synthesizer. First, we describe two methods of preprocessing of speech which we have used to improve the quantization properties of filter parameters. Next, as the quantization properties of pitch and gain are well understood, we discuss their quantization briefly. Finally, we report the results of a comparative study of several alternate sets of parameters representing the linear predictor. Specifically, we conclude that the reflection coefficients constitute the best set as transmission parameters.

A. Preprocessing of Speech

In our experiments, we observed that the short-time spectral dynamic range is the single most important factor that affects the quantization properties of transmission parameters. We define the spectral dynamic range to be the difference in decibels between the maximum and minimum spectral values within the frequency range of interest. The spectral dynamic range in turn is controlled by two somewhat related quantities, namely, the overall slope of the spectrum and the bandwidths of the poles and zeros. A large spectral slope or some narrow bandwidth poles result in a high dynamic range. We investigated two methods of preprocessing of the speech signal to reduce the spectral dynamic range and hence to improve the quantization

properties of transmission parameters. In the first method, called preemphasis, the speech signal is passed through an all-zero filter to alter its spectral slope. The second method, what we call the bandwidth expansion method, reduces the dynamic range by increasing pole bandwidths.

1. Preemphasis

To explain the concept, consider the first-order preemphasis illustrated in Fig. 8. The speech signal is passed through a filter with a simple real zero at z=b. From the amplitude responses shown, it is clear that when b is greater than zero, the high frequency components of the spectrum are emphasized, while when b is less than zero, the low frequency components are emphasized. The magnitude of b in both cases determines the extent of the emphasis. It is desirable to be able to find for a given speech segment a value for b that is optimal in some sense. Since the purpose here is to reduce the spectral slope as much as possible, it makes sense to estimate the overall spectral slope of the speech by a single pole filter and use its inverse for preemphasis. Since linear prediction gives an optimal all-pole approximation to the speech spectrum [2,7], it is clear that we can use linear prediction to determine the optimal value for b. For the first order case this optimal value can be expressed explicitly in terms of the autocorrelation of the speech signal: $b=R_1/R_0$.

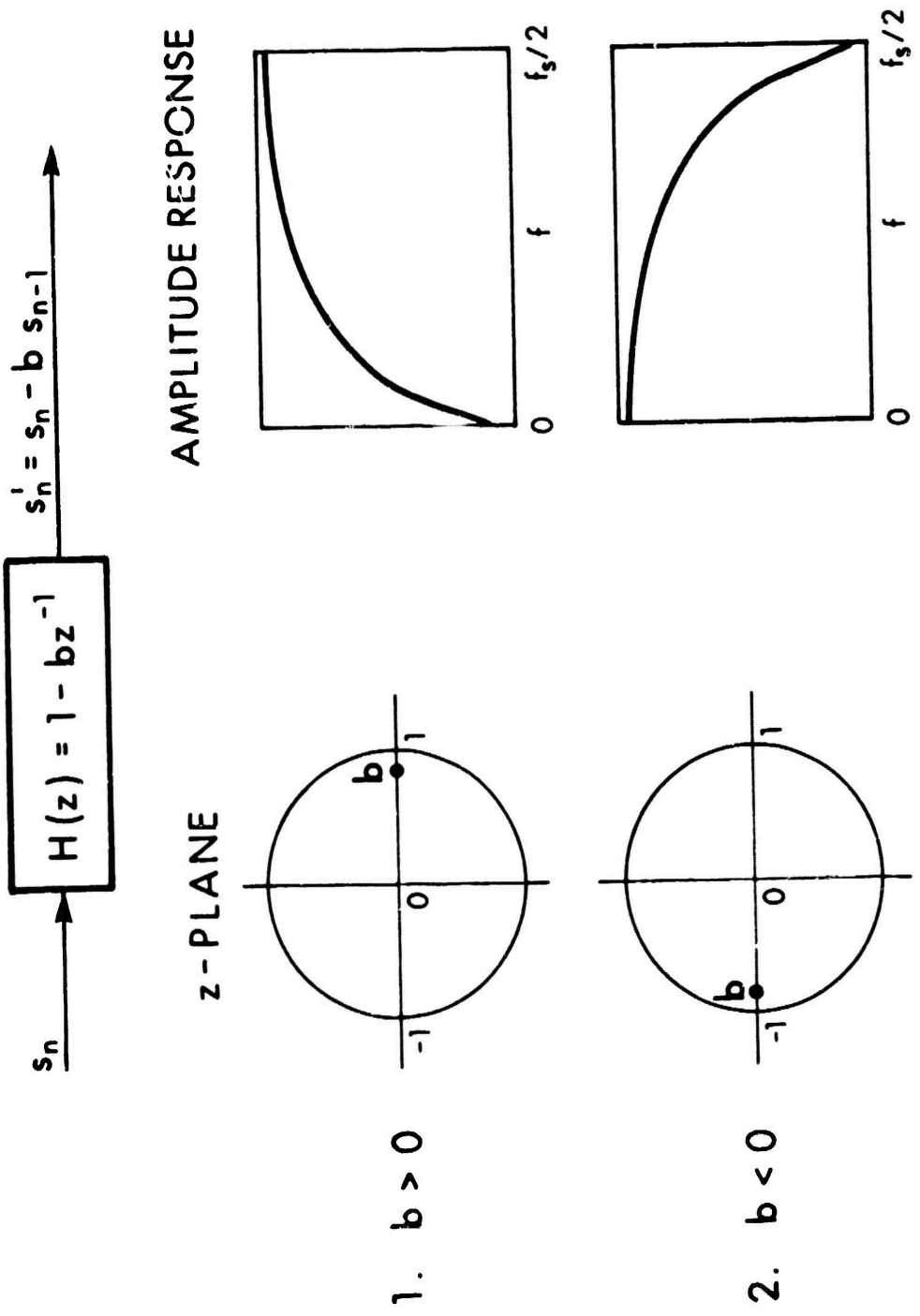


Fig. 8. First order preemphasis using a single real zero.

It is natural to ask whether higher order preemphasis would be more desirable. We have specifically investigated optimal second-order preemphasis. It clearly leads to a further reduction in dynamic range. However, when deemphasis (or postemphasis) is performed, distortions are introduced into the spectrum. This is illustrated in Fig. 9. At the top, we have the linear prediction spectrum after preemphasis. The solid curve represents the case where the parameters are unquantized, and the dashed curve represents the quantized case. (We used the reflection coefficients for quantization.) For the particular spectrum shown, the distortion due to quantization is small. The additional distortion attributable to second-order preemphasis is shown in the bottom plot, where the solid curve is the linear prediction spectrum with no preemphasis, and the dashed curve is the corresponding spectrum after preemphasis and deemphasis. In general, the first formant is affected most for voiced sounds; its frequency is often lowered, thereby producing a nasal-like quality or enhanced low-frequency buzz in the synthesized speech. It should be mentioned that such distortions occur even without any quantization. A primary reason for this phenomenon is that the second-order preemphasis often flattens the spectrum by eliminating the prominent formant in the speech. Upon postemphasis, this formant is not restored exactly. This was highlighted in our experiment where we observed that using an optimal second-order preemphasis filter in cascade with a suboptimal 10th order linear prediction filter (total order 12), produced a speech quality inferior to that of

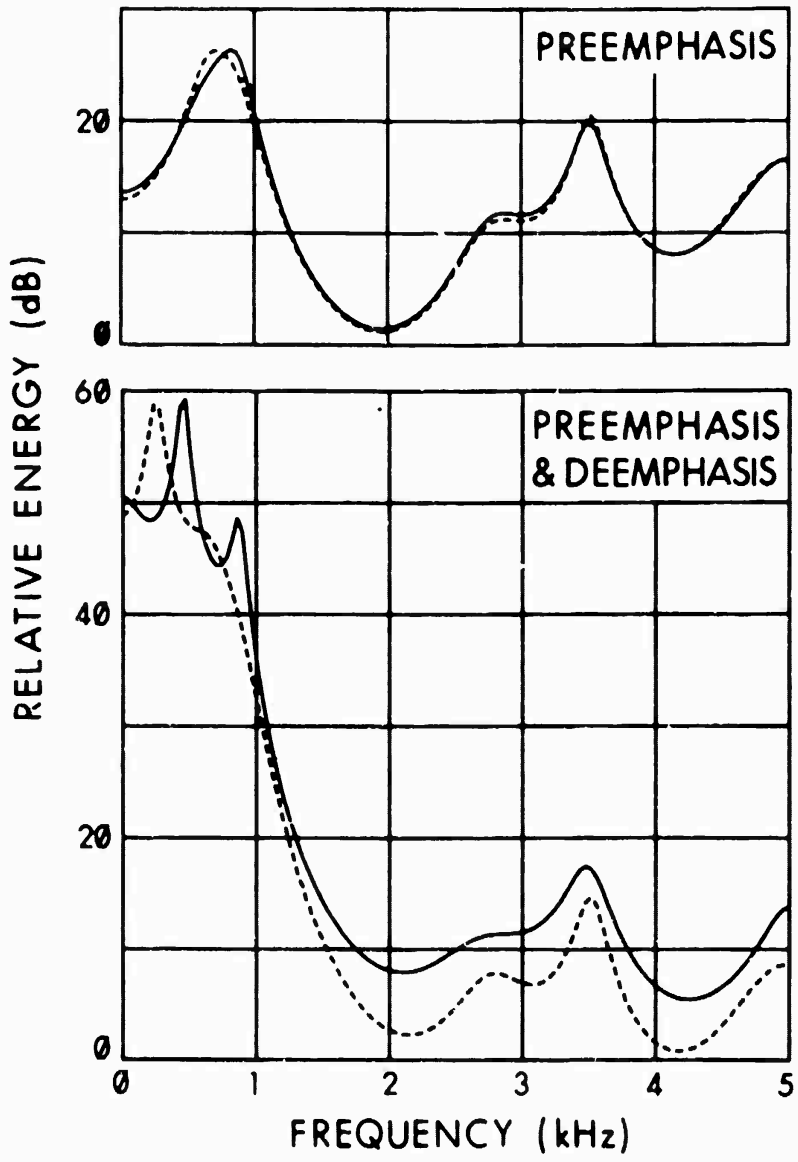


Fig. 9. An example illustrating the effects of using the optimal second-order preemphasis.

a 10th order optimal linear prediction filter with no preemphasis on the speech signal.

With optimal first-order preemphasis, only one real pole is removed from the input speech spectrum. This does not result in any perceivable distortion in the synthesized speech. In Fig. 10, the results with first-order preemphasis for the same example as above are shown. As can be seen, the spectra remain close even after deemphasis. Hence, we do not recommend the use of second-order optimal preemphasis in speech compression systems.

In continuous speech, the short-time spectrum changes with time, thus requiring different preemphasis filters, which must be encoded in some manner before transmission. We found that either 1 or 2 bit encoding of preemphasis data was sufficient. In one-bit encoding the signal was either not preemphasized or preemphasized using a fixed filter. This filter had its zero at 50 Hz (i.e. $b=e^{-100\pi T}$, T being the sampling period). With 2 bits, one is able to specify 4 preemphasis filters. By examining the quality of the synthesized speech, we concluded that one-bit adaptive preemphasis was adequate. However, for a real time system it might be sufficient and more practical to use simple fixed preemphasis.

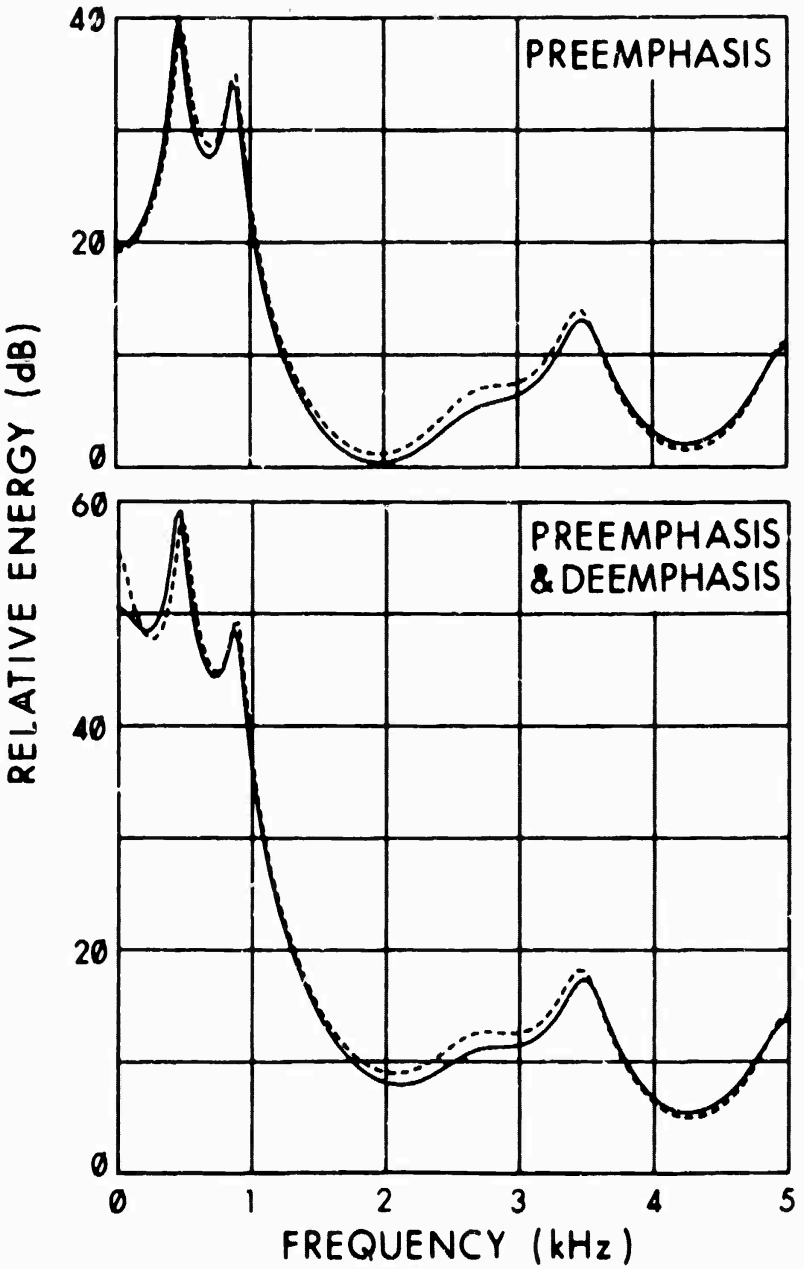


Fig. 10. Results obtained using the optimal first-order preemphasis for the same speech sound as in Fig. 9.

2. Bandwidth Expansion Method

This method reduces the spectral dynamic range by multiplying the impulse response of the inverse filter A(z) (i.e. the predictor coefficients) by a decaying exponential*. The new predictor coefficients are given by

$$a'_n = a_n e^{-\sigma n} \quad , \quad \sigma>0 \quad , \quad 1\leq n\leq p \quad . \qquad (16)$$

The result of this is to shift the poles of the linear predictor inwards with respect to the unit circle in the z plane, thus widening their bandwidths [20].

Preprocessing by either of these methods can be done after the linear prediction analysis, so that it can be viewed as part of the encoding process. In our experience we have found preemphasis to be a more effective preprocessing method than the bandwidth expansion method.

---

*If, however, an appropriate growing exponential is used, many of the pole bandwidths decrease thus enhancing the formant peaks in the spectrum and facilitating better formant tracking [2,7].

B. Quantization of Pitch and Gain

We quantize both pitch and gain logarithmically [21]. We use 6 bits to quantize pitch, with the quantization level of 0 indicating an unvoiced frame. The range of pitch frequency is taken to be 50-450 Hz. As gain parameter, we quantize the mean square value of the speech signal using 5 bits. We assume a range of 45 decibels.

C. Choice of Filter Parameters

For use as transmission parameters, we chose to investigate the following sets of parameters which uniquely characterize the linear prediction filter H(z):

(1) Impulse response of the inverse filter A(z), i.e. predictor coefficients a_n, $1 \le n \le p$.

(2) Impulse response of the all-pole model H(z), h_n, $0 \le n \le p$, which are easily obtained from (1) by long division. Note that the first p+1 coefficients uniquely specify the filter.

(3) Autocorrelation coefficients of $\{a_n/G\}$,

$$b_i = \frac{1}{G^2} \sum_{j=0}^{p-|i|} a_j \, a_{j+|i|} \quad , \quad a_0 = 1 \quad , \quad 0 \le i \le p. \qquad (17)$$

(4) Autocorrelation coefficients of $\{n_n\}$

$$r_i = \sum_{j=0}^{\infty} h_j h_{j+|i|} \quad , \quad 0 \le i \le p \quad . \qquad (18)$$

It can be shown that r_i is equal to R_i in (8) for $0 \le i \le p$ [3,7].

(5) Spectral coefficients of $A(z)/G$, P_i, $0 \le i \le p$, (or equivalently spectral coefficients of $H(z)$, $1/P_i$)

$$P_i = b_o + 2 \sum_{j=1}^{p} b_j \cos \frac{2\pi ij}{2p+1} \quad , \quad 0 \le i \le p \quad , \qquad (19)$$

where b_j are as defined in (17). In words, $\{P_i\}$ is obtained from $\{b_i\}$ through a discrete Fourier transform (DFT). Traditionally, vocoders that transmit the spectrum at selected frequencies have been known as channel vocoders. Thus, use of the spectral coefficients as transmission parameters leads to a linear prediction channel vocoder. While in the classical channel vocoder different channel signals are derived from contiguous band-pass filters, in the linear prediction channel vocoder a selected set of p+1 points from the all-pole spectrum constitute the "channel outputs." The main advantage of the linear prediction channel vocoder, however, is that we are able to regenerate exactly the all-pole spectrum from a knowledge of the p+1 spectral coefficients, unlike in the classical channel vocoder.

(6) Cepstral coefficients of $A(z)$, c_n, $1 \le n \le p$, (or equivalently cepstral coefficients of $H(z)/G$, $-c_n$)

$$c_n = \frac{1}{2\pi} \int_{-\pi}^{\pi} \log A(e^{j\omega}) \; e^{jn\omega} \, d\omega \quad .$$

Since A(z) is minimum phase, we obtain using the results given in [22, p. 246]

$$c_1 = a_1 \quad ,$$

$$c_n = a_n - \sum_{m=1}^{n-1} \frac{m}{n} c_m \, a_{n-m} \quad , \; 2 \leq n \leq p \quad . \tag{20}$$

(7) Poles of H(z) (or equivalently zeros of A(z)).

(8) Reflection coefficients k_i, $1 \leq i \leq p$, or simple transformations thereof, e.g. area coefficients [1,10]. The area coefficients are given by

$$A_i = A_{i+1} \frac{1+k_i}{1-k_i} \quad , \; A_{p+1} = 1 \quad , \; 1 \leq i \leq p \quad . \tag{21}$$

Some of the above sets of parameters have p+1 coefficients while others have only p coefficients. However, for the latter sets the signal energy (or gain G) needs to be transmitted as well, thus keeping the total number of parameters as p+1 for all the cases. Although the above sets provide equivalent information about the linear predictor, their properties under quantization are different. Certain aspects of the sets (1), (4), (7) and (8) have been studied in the past [1,9]. Our purpose was to investigate the relative quantization properties of all these parameters with a particular emphasis on the reflection coefficients.

It should be emphasized tha the predictor coefficients can be recovered from any of the various sets of parameters listed above. The required transformations for such a recovery are given below only for the sets (3) and (5) since they are either well-known or obvious for the others [23].

The sequence $\{b_i\}$ is transformed through an FFT after appending it with an appropriate number of zeros to achieve sufficient resolution in the resulting spectrum of the filter $A(z)/G$. The spectrum of the all-pole filter $H(z)$ is then obtained by simply inverting the amplitudes of the computed spectrum. Inverse Fourier transformation of the spectrum of $H(z)$ yields autocorrelation coefficients $\{r_i\}$ defined in (18). The first p+1 autocorrelation coefficients r_i, $0 \leq i \leq p$, are then used to compute the predictor coefficients via the normal equations (7) with $R_i = r_i$, $0 \leq i \leq p$.

The predictor coefficients are recovered from the spectral coefficients $\{P_i\}$ by first taking the inverse DFT of the sequence $\{P_i\}$ to get the autocorrelation sequence $\{b_i\}$. The process of getting the predictor coefficients from $\{b_i\}$ has been discussed above.

For the purpose of quantization, two desirable properties for a parameter set to have are: (a) filter stability upon quantization and (b) a natural ordering of the parameters. Property (a) means that the poles of $H(z)$ continue to be inside the unit circle even after parameter quantization. By (b) we

mean that the parameters exhibit an inherent ordering, e.g. the predictor coefficients are ordered as $a_1, a_2, \ldots, a_p$. If a_1 and a_2 are interchanged then H(z) is no longer the same in general, thus illustrating the existence of an ordering. The poles of H(z), on the other hand, are not naturally ordered since interchanging the order of any two poles does not change the filter. When such an ordering is present, a statistical study on the distribution of individual parameters can be used to develop better encoding schemes (e.g. Huffman coding, see Section VIII). Only the poles and the reflection coefficients ensure stability upon quantization, while all the sets of parameters except the poles possess a natural ordering. Thus, only the reflection coefficients possess both of these properties.

We investigated experimentally the quantization properties of the sets of parameters discussed above, with and without preprocessing of the speech signal. The absolute error between the log power spectra of the unquantized and the quantized linear predictors was used as a criterion in this study, since we believe that a good spectral match is necessary for synthesizing speech with good quality. A summary of the results is provided in the following.

The impulse responses $\{a_n\}$ and $\{h_n\}$ are highly susceptible to causing instability of the filter upon quantization. This is well-known from discrete filter analysis. Positive definiteness of autocorrelation coefficients $\{b_i\}$ and $\{r_i\}$ is not ensured

under quantization, which also leads to instabilities in the linear prediction filter. An attempt to synthesize speech with quantized autocorrelation coefficients $\{r_i\}$ resulted in distinctly perceivable "clicks" in the synthesized speech. Our conclusion is that the impulse responses and autocorrelation coefficients can be used only under minimal quantization, in which case the transmission rate would be excessive.

In the experimental investigation of the spectral and cepstral parameters, we found that the quantization properties of these parameters are generally superior to those of the impulse responses and autocorrelation coefficients. The spectral parameters often yield results comparable to those obtained by quantizing the reflection coefficients. However, for the cases when the spectrum consists of one or more very sharp peaks (narrow bandwidths), the effects of quantizing the spectral coefficients often result in the autocorrelation coefficients $\{b_i\}$ being non-positive definite and hence cause certain regions in the reconstructed spectrum to become negative. This in turn causes the autocorrelation coefficients $\{r_i\}$ to be non-positive definite, which leads to instability of the filter. Preprocessing the speech signal by the bandwidth expansion method (see Section A) remedies this situation, but the spectral deviation in these regions can be relatively large. Quantization of cepstral parameters can also lead to instabilities, where the predictor coefficients are computed from (20). As before, with proper preprocessing stability is

restored, but at the expense of increased spectral deviation.

As mentioned earlier, the stability of the filter H(z) is guaranteed under quantization of the poles. This makes the poles potentially a good set of parameters for transmission. Unfortunately, the poles do not possess a natural ordering: a property that is necessary if a low transmission rate is desired. Traditionally, poles have been ordered in terms of vocal tract resonances (formants). Since the ranges of frequencies for the various formants have been well established, their quantization can be done with improved accuracy. In addition, the formant bandwidths may be quantized less accurately than formant frequencies, which leads to further savings in transmission rate. However, experience has shown that the problem of identifying the poles as ordered formants is computationally complex and involves a fair amount of decision making which is not completely reliable. In addition, computing the poles requires finding the roots of a pth order polynomial ($p \sim 12$): not a straightforward task.

Based on the results of our experimental study of the spectral deviation due to quantization, on computational considerations, and on stability and natural ordering properties, we concluded that the reflection coefficients are the best set for use as transmission parameters. In addition to these advantages, the values of the reflection coefficients k_i, $i<p$, do not change as p is varied, unlike any of the other parameters. (This property of the reflection coefficients is

useful when applying the variable order linear predictive analysis discussed in Section IV.)

V OPTIMAL QUANTIZATION OF REFLECTION COEFFICIENTS

Having selected the reflection coefficients, we proceeded to develop an optimal quantization scheme which gives the best results in terms of the quality of the synthesized speech. First, we established a suitable criterion with respect to which we developed an optimal quantization scheme. It is known that an utterance that has been synthesized perfectly but for one or two "glitches" (segments involving large errors of some sort) would invariably be rated by a human subject as having a relatively poor quality. In other words, these glitches mask the perception giving an impression that the utterance has been poorly synthesized. Thus, the quality of the synthesized speech is a function of the "maximum perceptual error" between the synthesized and the original speech. Therefore, a reasonable criterion is to minimize the maximum perceptual error. We assumed that an accurate representation of the power spectrum is necessary for synthesizing good quality speech. Thus, the criterion we used for optimal quantization was to minimize the maximum spectral error due to quantization.

To use the minimax spectral error criterion in developing an optimal scheme for quantizing the reflection coefficients, it was necessary first to investigate the sensitivity of the all-pole model spectrum to small changes in the values of the reflection coefficients. Section A below describes this sensitivity analysis. The development of an optimal quantization scheme using the sensitivity properties is given in

Section B. Section C presents an optimal bit allocation strategy that we derived for the transmission parameters by minimizing the maximum spectral error due to quantization. Finally, we present in Section D the results of our investigation of a second sensitivity measure for the reflection coefficients. A detailed description of the material given in this section can be found in [23].

A. Sensitivity Analysis

We define the spectral sensitivity for the reflection coefficients by [23]

$$\frac{\partial S}{\partial k_i} = \lim_{\Delta k_i \to 0} \left| \frac{1}{\Delta k_i} \left[\frac{1}{2\pi} \int_{-\pi}^{\pi} \left| \log P(k_i,\omega) - \log P(k_i+\Delta k_i,\omega) \right| d\omega \right] \right|, \qquad (22)$$

where

$$P(\cdot,\omega) = \left| H(e^{j\omega}) \right|^2$$

is the spectrum of the all-pole model H(z). The quantity between brackets in (22) is the spectral deviation due to a perturbation in the ith reflection coefficient. Experimentally, $\frac{\partial S}{\partial k_i}$ was computed by replacing the integral by a summation, and by using a sufficiently small value for Δk_i. A sensitivity curve $\frac{\partial S}{\partial k_i}$ versus k_i was obtained by plotting sensitivity values $\frac{\partial S}{\partial k_i}$ for k_i in the interval (-1,1) while keeping the other reflection coefficients fixed. We performed this type of sensitivity analysis for a large number of speech sounds recorded from male and female speakers. The sensitivity curves have the following

properties in common:

(i) Each sensitivity curve $\frac{\partial S}{\partial k_i}$ versus k_i has the same general shape, irrespective of the index i and irrespective of the values of the other coefficients k_n, $n \neq i$, at which the sensitivity is computed.

(ii) Each sensitivity curve is U-shaped. It is even-symmetric about $k_i=0$, and has large values when the magnitude of k_i is close to 1 and small values when the magnitude of k_i is close to zero.

It must be emphasized that property (i) refers only to the shape of the sensitivity curve. The actual value of the sensitivity for a particular reflection coefficient does in general depend on the values of the other reflection coefficients.

Although the above sensitivity properties were derived experimentally by perturbing, one at a time, the magnitudes of the reflection coefficients that corresponded to different speech sounds, these properties should be viewed as inherent to the reflection coefficients themselves and not to the particular speech sounds. Thus, voiced sounds generally have a higher spectral sensitivity than unvoiced sounds because some of the reflection coefficients for voiced sounds have magnitudes close to 1. Also, in general, preemphasis reduces the spectral sensitivity of voiced sounds by reducing the magnitudes of the reflection coefficients which are close to 1.

The sensitivity properties given above strongly suggested the existence of a prototype sensitivity function which would apply approximately to every reflection coefficient and for different speech sounds. Such a prototype function could then be used in developing an optimal quantization scheme that would apply to all reflection coefficients all the time. In view of the above sensitivity properties, we computed this prototype sensitivity function by simply averaging the sensitivity curves over different reflection coefficients and for a large number of different speech sounds. Such an averaged sensitivity function is shown plotted as the solid curve in Fig. 11. In this plot the sensitivity values are given in decibels relative to the sensitivity at k=0. In the following, we present an optimal quantization scheme for the reflection coefficients which we developed using the averaged sensitivity function in Fig. 11.

B. Optimal Quantization

From the sensitivity properties of the reflection coefficients discussed in the previous section and depicted in Fig. 11, it is clear that linear quantization of the reflection coefficients is not satisfactory, especially when some of them take values close to 1 in magnitude. What is needed is a nonlinear quantization scheme that is much more sensitive (has more steps) near 1 than near 0. A nonlinear quantization of a reflection coefficient is equivalent to a linear quantization of

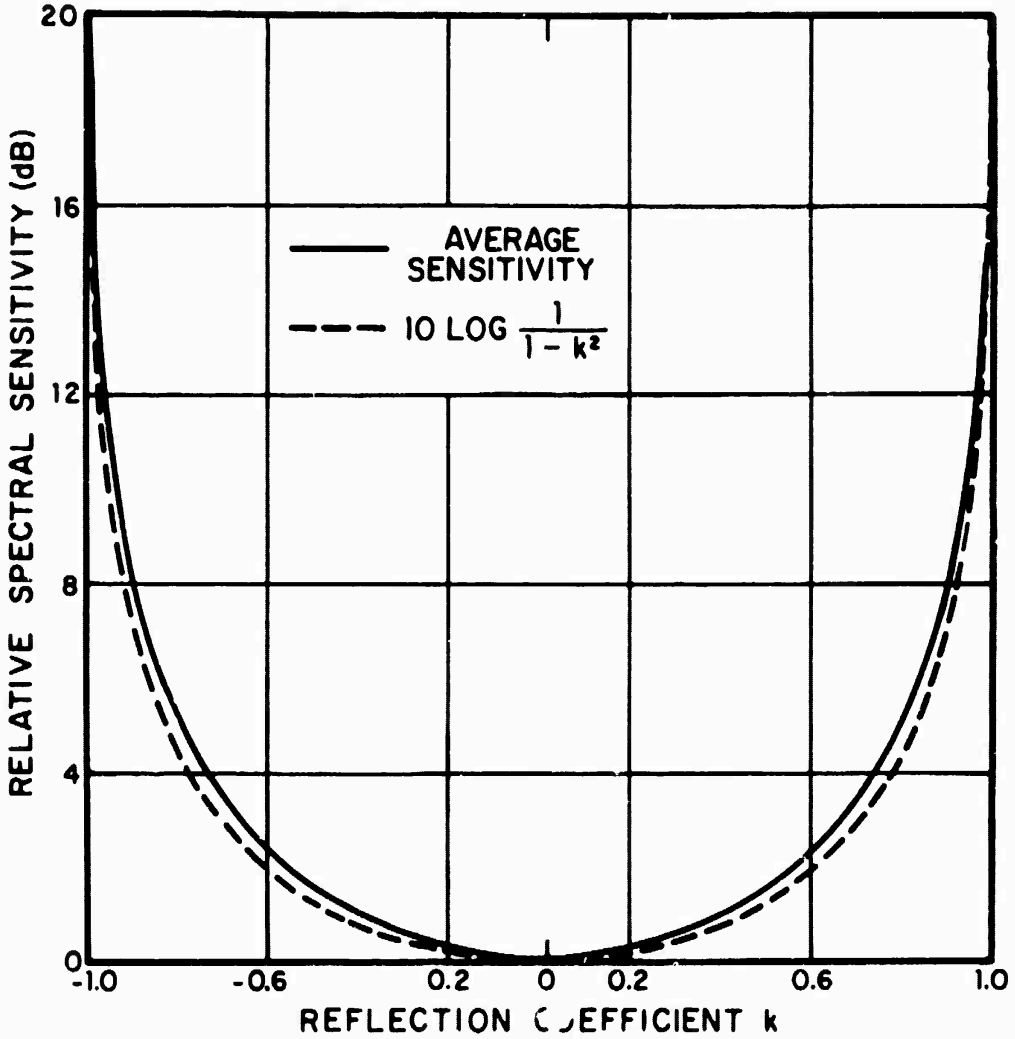


Fig. 11. Averaged spectral sensitivity curve for the reflection coefficients (solid line) and an analytical function that approximates it (dashed line).

a different parameter that is related to the reflection coefficient by a nonlinear transformation. It is not difficult to show that linear quantization of the transformed parameter is optimal (in the sense of minimizing the maximum spectral error due to quantization) if and only if the transformed parameter has a flat or constant spectral sensitivity behavior. The sufficiency of the condition is evident from the fact that with a flat sensitivity behavior and linear quantization, the maximum spectral error is constant over the entire range of variation of the parameter, which trivially leads to a minimum equal to that constant value. The necessity of the condition can be established by using the proof by contradiction method as follows. If the transformed parameter does not have a flat sensitivity behavior, then a suitable nonlinear quantization leading to a smaller maximum spectral error can be found by assigning smaller quantization steps in regions where the parameter has high sensitivity and vice versa. This is clearly a contradiction to the fact that linear quantization is optimal. Thus, the search for the optimal quantization scheme for the reflection coefficients reduces to the search for a nonlinear transformation that results in a flat spectral sensitivity behavior for the transformed parameters.

If the transformed parameter is denoted by $g=f(k)$, we have shown in [23] that the optimal nonlinear transformation $f(k)$ is given by

$$\frac{df(k)}{dk} = L \frac{\partial S}{\partial k} \quad , \tag{23}$$

where L is some constant. To derive the mapping that is optimal on the average for all the reflection coefficients, we used the averaged sensitivity function in Fig. 11. Although it is possible to obtain the optimal transformation by integrating the solid curve in Fig. 11 directly, we found it simpler and ultimately more useful to approximate the averaged sensitivity curve by a well specified mathematical function which could then be integrated to obtain an approximately optimal f(k). An experimental fitting of the averaged sensitivity curve in Fig. 11 has revealed that the function $1/(1-k^2)$ approximates the sensitivity function reasonably well (to within a multiplicative constant), as shown by the dashed curve in Fig. 11 (note that the plot is given in decibels). Using this approximation in (23) and integrating with L=2, we get the optimal mapping as

$$f(k) = \log \frac{1+k}{1-k} \quad . \tag{24}$$

The optimally transformed parameters are therefore given by

$$g_i = \log \frac{1+k_i}{1-k_i} \quad , \quad 1 \le i \le p \quad . \tag{25}$$

Using (21), the transformation in (24) is simply the logarithm of the ratio of the consecutive area coefficients. Thus, we have shown that the logarithms of the area ratios (henceforth called log area ratios) provide an approximately optimal set of

coefficients for quantization.

Figure 12 shows a plot of the log area ratio as a function of the reflection coefficient. We have also plotted in Fig. 12 a linear characteristic that passes through the intersection of a vertical line at k=0.7 and the log area ratio curve. For values of k less than 0.7 in magnitude, the log area ratio curve is almost linear. Thus, if a certain reflection coefficient takes values always less than 0.7 in magnitude, one could quantize it linearly to obtain approximately flat sensitivity characteristics. In practice it is found that the reflection coefficients k_i, i>3, have in general magnitudes less than 0.7. However, use of the log area ratios automatically leads to the desired quantization irrespective of the reflection coefficient and the range of values it spans.

We note from (10) and (25) that, for a stable filter, the log area ratios take on values in the region $-\infty<g_i<\infty$, for all i. The filter becomes unstable if any of the log area ratios becomes unbounded. The potential unboundedness of the log area ratios means that the range over which they need to be quantized can be very large, which can lead to an excessive number of quantization bits or else to very coarse quantization. However, in practice, the range is often limited by the types of signals that are processed. For example, we have not found the range to be very large for speech signals, especially when preemphasis is used. The problem could still arise, as a result of computations with a small wordlength. In that case, the range

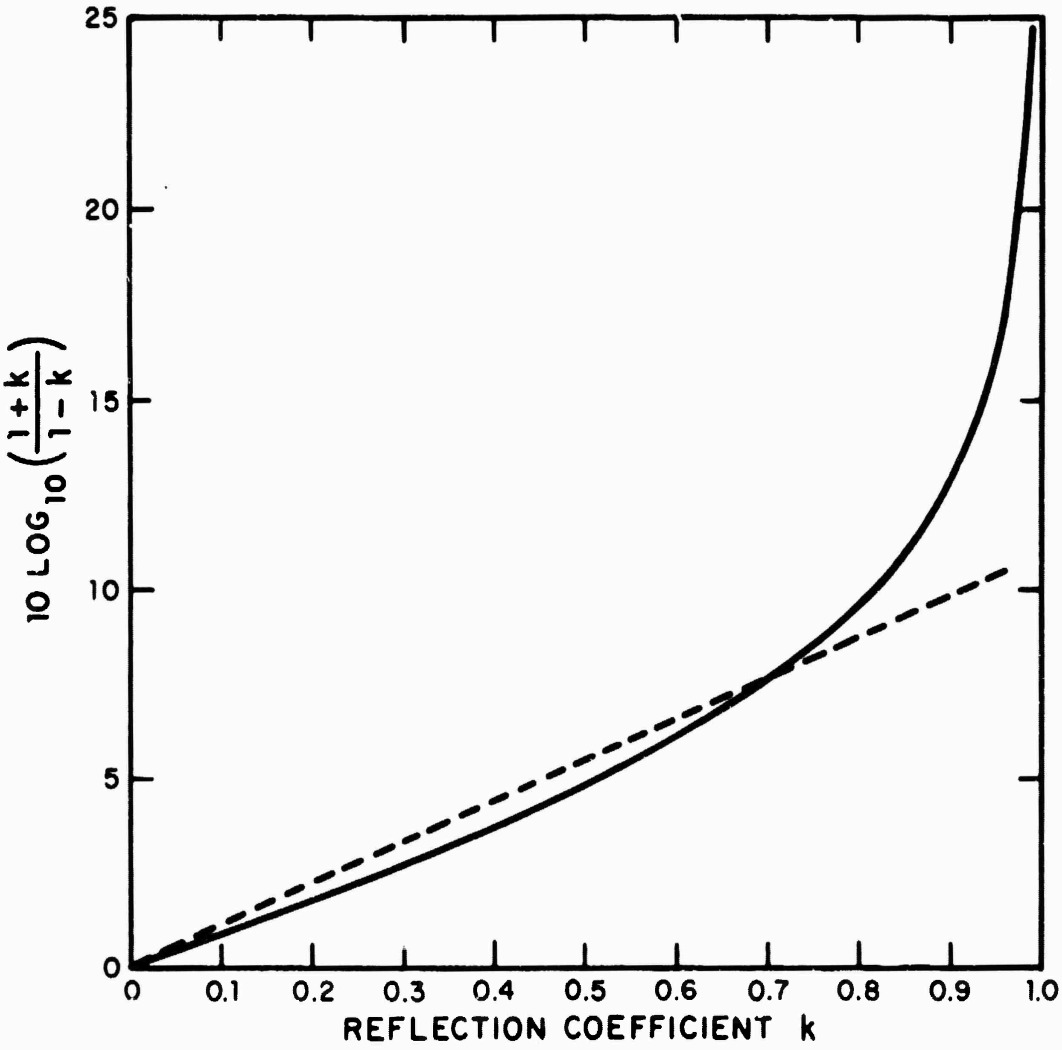


Fig. 12. Log area ratio plotted as a function of the reflection coefficient (solid line) and a linear characteristic that intersects it at k=0.7 (dashed line).

could be limited artificially. This is good practice because otherwise very narrow bandwidth filters would result, which in general is not a good thing in speech synthesis.

C. Optimal Bit Allocation

For the log area ratios, we have derived an optimal bit allocation strategy by minimizing the maximum spectral deviation due to quantization [23]. If g_i is the ith log area ratio with its lower and upper bounds $(g_i)_{max}$ and $(g_i)_{min}$, respectively, and N_i is the number of levels used for its quantization, the step size for g_i is given for linear quantization by

$$\delta_i = \frac{(g_i)_{max} - (g_i)_{min}}{N_i} \quad . \tag{26}$$

The optimal bit allocation is obtained if δ_i is the same for all the log area ratios. This result is also intuitively clear since the spectral sensitivity is approximately constant and is approximately the same for all the log area ratios. The total number of bits required to quantize the p log area ratios is

$$M = \log_2 \left[\prod_{i=1}^{p} N_i \right] \quad .$$

We found it convenient and useful to begin with a particular quantization step size. That automatically determines the total number of bits needed, as well as the maximum spectral deviation, which in turn determines the resulting speech quality. One can then study the change in speech quality as a function of only one variable, namely the quantization step

size.

For the quantization of the log area ratios as well as for determining the optimal bit allocation strategy discussed above, we need the knowledge of the ranges of the different log area ratios g_i, $1 \leq i \leq p$. For a set of 12 speech utterances that were sampled at 10 kHz and preemphasized using a fixed filter, we extracted, at a rate of 100 frames/sec, the log area ratios through the linear predictive analysis using p=11 and an analysis interval of 20 msec. The maximum and minimum values were found for each log area ratio, and the corresponding range was then determined by allowing some margin on both of these values. In this study we used $10 \log_{10}$ instead of the natural logarithm in computing log area ratios from (25). So, the computed log area ratios were in "decibels". In collecting the range statistics for log area ratios, we treated voiced and unvoiced sounds separately. In our experience we found that using a step size of 1 dB for quantizing log area ratios provides a good compromise between speech quality and transmission rate. In Table I we have given the bounds of log area ratios along with their optimal bit (level) allocations for a step size of 1 dB [21].

TABLE I

i	VOICED			UNVOICED		
	$(g_i)_{max}$(dB)	$(g_i)_{min}$(dB)	N_i	$(g_i)_{max}$(dB)	$(g_i)_{min}$(dB)	N_i
1	10.5	-17.5	28	15.5	-13.5	29
2	14.5	- 7.5	22	14.5	- 6.5	21
3	7.5	-11.5	19	7.5	- 6.5	14
4	9.5	- 5.5	15	8.5	- 4.5	13
5	6.5	- 7.5	14	4.5	- 5.5	10
6	8.5	- 4.5	13	5.5	- 3.5	9
7	7.5	- 5.5	13	6.5	- 5.5	12
8	9.5	- 4.5	14	6.5	- 4.5	11
9	7.5	- 4.5	12	5.5	- 4.5	10
10	6.5	- 4.5	11	5.5	- 4.5	10
11	4.5	- 4.5	9	4.5	- 4.5	9

D. Comments on Another Spectral Sensitivity Measure

In Section A we introduced a spectral sensitivity measure to study the quantization properties of the reflection coefficients. Other types of sensitivity measures may also be used. In particular we have considered a measure which is similar to the total-squared error used for minimization in linear predictive analysis. By using Parseval's theorem in (3), the total-squared error is given by

$$E = \frac{G^2}{2\pi} \int_{-\pi}^{\pi} \frac{P_o(\omega)}{P(\omega)} \, d\omega \quad , \tag{27}$$

where $P_o(\ )$ is the power spectrum of the input speech signal and $P(\ )$ is the power spectrum of the all-pole filter:

$$P(\omega) = \left|H(e^{j\omega})\right|^2 = \frac{G^2}{\left|A(e^{j\omega})\right|^2} \quad . \tag{28}$$

The gain G is given by (13).

We have studied the properties of the error measure E in detail [2,7,24]. In particular, the minimization of E results in an all-pole model spectrum $P(\omega)$ that is a good approximation to the envelope of the signal spectrum $P_o(\omega)$. Because of this property, it seemed reasonable to study the use of this error E as a measure of the deviation between the two spectra. For the sake of normalization we have chosen to work with an error measure E' obtained from (27) by eliminating the factor G^2:

$$E' = \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{P_1(\omega)}{P_2(\omega)} \, d\omega \quad , \tag{29}$$

where $P_1(\omega)$ and $P_2(\omega)$ are now any two spectra. Also, the two spectra are normalized such that they have equal total energy.

For our study of spectral sensitivity we let $P_1(\omega)=P(k_i,\omega)$ and $P_2(\omega)=P(k_i+\Delta k_i,\omega)$, where $P(\cdot,\omega)$ is given by (28). The error between the two spectra is then given by

$$E'(\Delta k_i) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{P(k_i,\omega)}{P(k_i+\Delta k_i,\omega)} \, d\omega \quad . \tag{30}$$

The new measure of spectral sensitivity is defined as

$$\frac{\partial S'}{\partial k_i} = \lim_{\Delta k_i \to 0} \left| \frac{1}{\Delta k_i} \log \left[\frac{1}{2\pi} \int_{-\pi}^{\pi} \frac{P(k_i,\omega)}{P(k_i+\Delta k_i,\omega)} \, d\omega \right] \right| . \tag{31}$$

We have derived the spectral sensitivity in (31) analytically, without the need to resort to experimental data as was the case for the study of $\frac{\partial S}{\partial k_i}$ in (22). The result can be shown to be [23]

$$\frac{\partial S'}{\partial k_i} = \frac{2|k_i|}{1-k_i^2} \quad . \tag{32}$$

It is important to note that this is an exact result and it is true for each reflection coefficient, independent of the values of the other coefficients. A plot of $\frac{\partial S'}{\partial k}$ versus k also gives a U-shaped curve. Therefore, the spectral sensitivity in (32) has

the same general properties as the spectral sensitivity $\frac{\partial S}{\partial k}$ obtained experimentally in Section A. The only difference between the two is the actual shape of the sensitivity curve.

Substituting (32) in the optimality condition (23) and integrating it with L=1, we obtain the following optimal mapping for the sensitivity measure (31):

$$f'(k) = \text{sign}(k) \log \frac{1}{1-k^2} \quad , \tag{33}$$

where sign(k) is +1 if k is positive and -1 if k is negative. From (12) and (33), it is interesting to observe that $|f'(k_i)|$ is equal to the logarithm of the ratio of the normalized errors (or log error ratio) associated with the linear predictors of orders i-1 and i,

$$f'(k_i) = \text{sign}(k_i) \log \frac{V_{i-1}}{V_i} \quad . \tag{34}$$

We experimentally investigated the quantization properties resulting from the mappings given by (24) and (33). Through informal listening tests we found that the use of the log area ratios for quantization leads to uniformly better speech quality than that obtained using the log error ratios. This points out the important fact that not all reasonable spectral sensitivity measures lead to good results; the measure must somehow relate to perception. Our conclusion is that the spectral sensitivity measure in (22) relates more to perception than the measure in

(31) since it produces better results terms of speech quality.

VII. VARIABLE FRAME RATE TRANSMISSION

This section deals with time quantization, i.e. the rate and manner in which parameters are transmitted in time. For a constant frame rate scheme, parameters are transmitted at fixed time intervals. A variable frame rate scheme transmits parameters only when the speech characteristics have sufficiently changed. Parameter transmissions occur more frequently when speech characteristics are changing rapidly as in phoneme transitions, while the transmissions are spaced farther apart when speech characteristics are relatively constant as in steady state sounds. As compared to a constant frame rate transmission system, the variable frame rate transmission system could, if designed properly, yield lower transmission rates for the same speech quality. We describe below a variable frame rate scheme that we use in our speech compression system.

To determine if speech characteristics have sufficiently changed since the last transmission, we use a measure that is the logarithm of the ratio of the mean-squared values of the error signal obtained (i) when the optimal linear predictor parameters are used and (ii) when the last transmitted parameters are used. If the predictor parameters are assumed to have Gaussian probability distributions, then this measure is the same as the log likelihood ratio [25]. To see how the transmission scheme works, let us suppose that we have decided to transmit the parameters for frame 1. Denote the predictor

coefficients of frame 1 (reference frame) by $a_k^{(1)}, 1 \le k \le p$. For frame 2 (test frame), the optimal linear predictor coefficients $a_k^{(2)}$, $1 \le k \le p$, are first determined. The speech signal for frame 2 is passed through the inverse filters $A_1(z)$ and $A_2(z)$ given by

$$A_1(z) = 1 + \sum_{k=1}^{p} a_k^{(1)} z^{-k} , \tag{35}$$

$$A_2(z) = 1 + \sum_{k=1}^{p} a_k^{(2)} z^{-k} . \tag{36}$$

The mean-squared values of the output signals of these inverse filters are computed as

$$E^{(1)} = b_o^{(1)} R_o + 2 \sum_{k=1}^{p} b_k^{(1)} R_k , \tag{37}$$

$$E^{(2)} = R_o + \sum_{k=1}^{p} a_k^{(2)} R_k , \tag{38}$$

where R_k are the autocorrelation coefficients of the speech signal for frame 2 and $b_k^{(1)}$ are the autocorrelation coefficients of the impulse response of $A_1(z)$, i.e.

$$b_k^{(1)} = \sum_{i=0}^{p-k} a_i^{(1)} a_{i+k}^{(1)} , \quad a_o^{(1)}=1 , \quad 0 \le k \le p . \tag{39}$$

The deviation between the two sets of predictor coefficients $\{a_k^{(1)}\}$ and $\{a_k^{(2)}\}$ is computed using the distance measure

$$d = 10 \log_{10} (E^{(1)}/E^{(2)}) . \tag{40}$$

As mentioned earlier, the distance measure d in (40) becomes the log likelihood ratio when the predictor coefficients have Gaussian probability distributions. The next step is to compare the distance d against a threshold. If d is within the

threshold (success), the data for frame 2 is not transmitted; however, data transmission occurs if d exceeds the threshold (failure). In the former case, the above procedure is repeated for the successive test frames using frame 1 as the reference, until a failure occurs or the number of consecutive successes exceeds a preset limit. When one of these two conditions is satisfied, the data for frame 1 is transmitted along with the number of consecutive successes. At the receiver, we interpolate between parameter receptions to ensure smoother transitions in parameter updating.

In our experiments, we used an analysis rate of 100 frames/sec (i.e., parameters were extracted once every 10 msec). A satisfactory value of the threshold for the log likelihood ratio measure was found experimentally as 1.5 dB. Parameter transmissions were not allowed to be spaced by more than 80 msec (8 frames). Variable frame rate transmission was used only for log area ratios. Pitch and gain were transmitted, at a constant rate of 50 times/sec. With these specifications, we experimented with 14 sentences of speech material from 10 speakers (male and female). Fig. 13 shows the relative frequencies of occurrence of the different transmission interval sizes 10-80 msec and the corresponding percentage bit savings in transmitting log area ratios. The transmission rate for log area ratios varied between 24 and 45 frames/sec, with an average of 37. Thus, we achieved a total saving of 63% in transmitting log area ratios. In these experiments we found that the quality

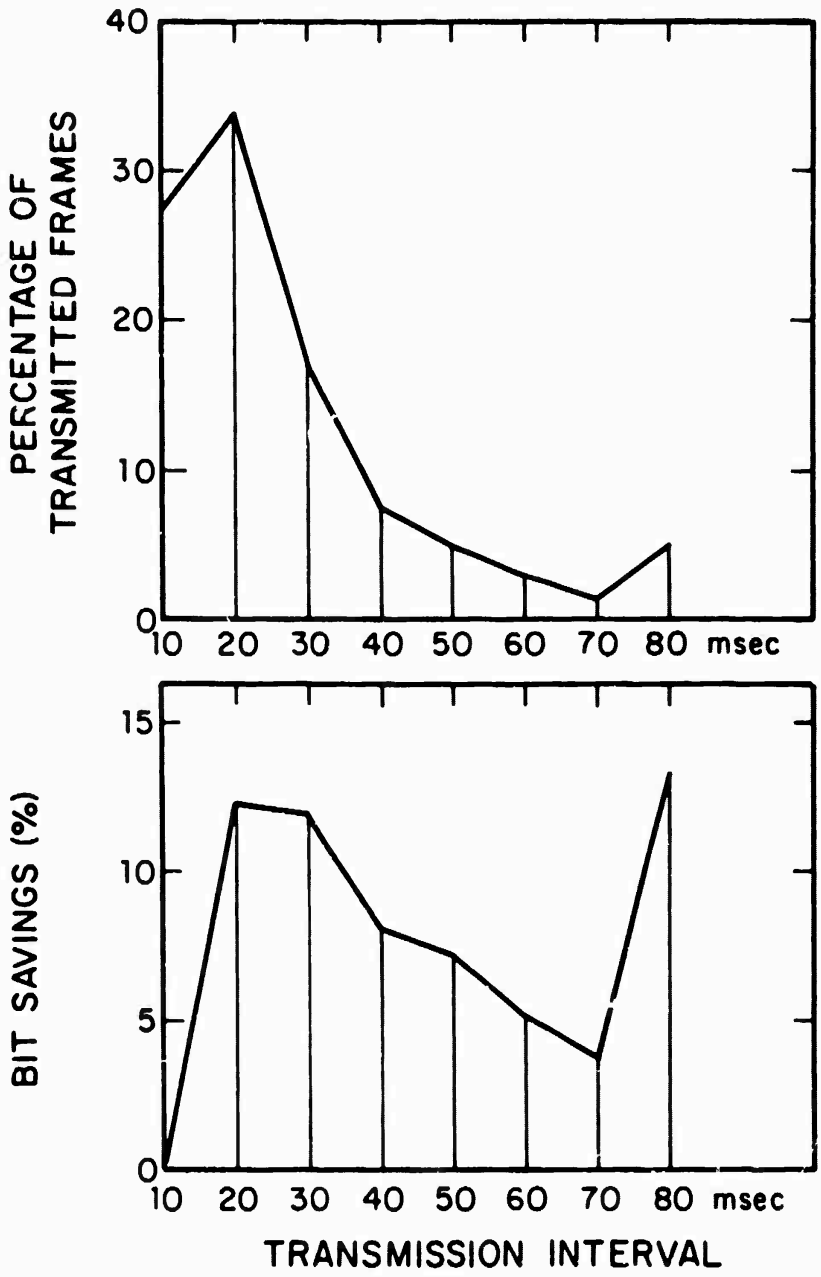


Fig. 13. Results using variable frame rate transmission. Histogram of the transmission interval (top), and percentage bit savings versus transmission interval (bottom).

of the synthesized speech dropped only slightly for the variable frame rate transmission (37 frames/sec on the average) relative to the constant frame rate transmission (100 frames/sec). However, when compared to a constant 50 frames/sec system, the above variable frame rate scheme produced distinctly better quality speech.

As an alternative to the log likelihood ratio measure described above, we made a preliminary investigation of another measure of spectral deviation using the log area ratios. This measure is simply the average of the absolute differences between respective log area ratios of the frame under test and the previously transmitted data frame. In another study (see Section XII-A) we found that the log area ratio error measure has an approximately linear relationship with the spectral error measure. This suggests that the log area ratio error measure might have a good correlation with speech quality. However, further testing is needed before any conclusive statement can be made.

VIII. VARIABLE WORDLENGTH ENCODING

We have investigated information theory approaches to transmission rate reduction. An encoding technique, called Huffman coding, has been chosen for our system, which makes use of the statistical distributions of the quantized values of the transmission parameters. Using the statistical data for each parameter, Huffman coding codes the values that are most likely to be transmitted with fewer bits. Thus the number of bits, or wordlength, required to code a set of values for a parameter is variable. It should be pointed out that no compromise whatsoever in speech quality is made when employing Huffman coding, because the coding does not result in any information loss; it merely transmits the information more efficiently.

Another encoding method we have used, called the delta encoding method, codes the change in a parameter from frame to frame. With delta encoding, the statistical distributions used for Huffman coding of pitch and gain became sharpened thus making the Huffman coding of these parameters more effective.

A. Huffman Coding

Huffman code is the optimal unambiguous variable wordlength code [26]. For each parameter it generates the lowest possible average transmission rate. That is, $\Sigma P_i L_i$ is minimized, where P_i is the probability of the ith value of a parameter, and L_i is the number of bits required to code that value. Furthermore,

the particular method we have used also minimizes the maximum code length $\max_i L_i$ and the total lengths of all codes $\sum_i L_i$ [27]. No two different parameter values result in the same code, and given the beginning of a code sequence, no further information is needed in order to know where a code begins or ends (i.e. Huffman code is unambiguous).

In order to find the Huffman code for a particular set of values, the frequencies of occurrence for these values must be known. The details of Huffman coding can be best explained by an example. Consider a parameter that takes on 7 possible values. Given below are the 7 values and the number of times each occurred:

value	occurrences	probability
0	600	3/7
1	200	1/7
2	200	1/7
3	150	3/28
4	100	1/14
5	100	1/14
6	50	1/28

To find the code, the two lowest frequencies are found and combined. That is, the frequencies of the values 6 and 5 would have a combined frequency of 150. The process of combining frequencies is continued until all have been combined, yielding a total frequency in this example of 1400. Fig. 14 shows how these different frequencies are combined producing a tree structure. The boxed numbers in Fig. 14 are the combined frequencies, and the numbers above the boxes indicate the order in which the combinations are formed. The depth of a node is

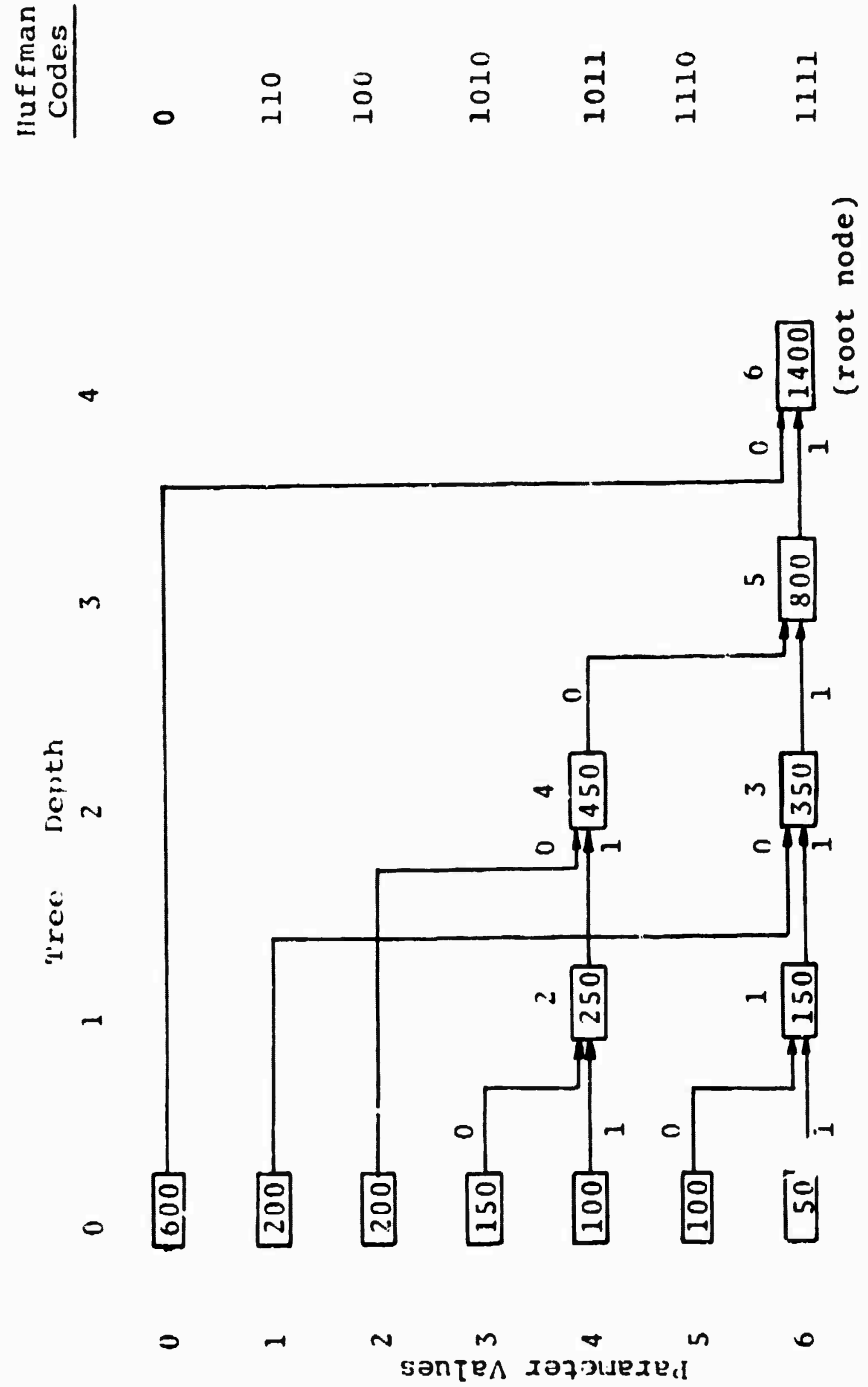


Fig. 14. An example illustrating the details of Huffman coding.

the maximum path length from an initial constituent node to that node. When two frequencies are equal, as are the uncombined frequency corresponding to value 3 and the combined frequencies of values 5 and 6, the frequency whose depth is the smaller is considered the lower. That is, the depth of the combined frequencies of 5 and 6 is 1, while the depth of the uncombined frequency corresponding to value 3 has a depth of 0, so the latter would be used in the next minimum pair.

Once the combinations have been completed, a tree has been formed and can be retraced from the root node (1400) to find the codes. Each arrow, or branch, will be assigned one bit, either 0 or 1, depending on whether it is the top or bottom branch into the node. (This assignment is arbitrary and can be reversed if desired.) The codes for the different values can be read from the tree as: $0 \rightarrow 0$, $1 \rightarrow 110$, $2 \rightarrow 100$, $3 \rightarrow 1010$, $4 \rightarrow 1011$, $5 \rightarrow 1110$, $6 \rightarrow 1111$. The average length is therefore the sum of the code lengths times the probabilities of their occurrence, or 2.43, compared to the simple binary code which requires 3 bits.

The minimum average length of the Huffman code can be approximated by the entropy of the parameter, i.e.

$$L_{min} \doteq - \sum_i P_i \log_2 P_i \ . \qquad (41)$$

In the example given above, the entropy is 2.39.

Even if the probability distribution of the parameter values was uniform for the above example, Huffman code would require an average length of 2.86, which would still provide a saving over simple binary encoding. Thus, the saving in the average code length of a parameter offered by Huffman coding is due in part to the number of parameter values being a non-integer power of 2 and in part to the probability distribution of the parameter values being non-uniform.

Huffman coding offers several advantages over simple binary coding of transmission parameters.

(a) Of primary importance, in the range of transmission rates that we are interested (2000 bps), Huffman coding reduces the transmission rate by approximately 20%. In so doing, it introduces no new approximations, as it codes only information, not acoustic phenomena. This property also allows it to be combined with other bit-saving techniques such as variable order linear prediction or variable frame rate transmission.

(b) Huffman coding allows any number of quantization levels. The number of quantization levels for a particular parameter does not have to be a power of 2 to produce efficient code. This property allows the number of levels to be chosen according to other criteria, such as equal step size or equal spectral sensitivity.

(c) Huffman coding has been proven optimal [26]. It therefore provides a useful standard against which to measure other

coding schemes.

Along with its advantages, Huffman coding also presents some disadvantages. Since, for a given parameter, the number of bits transmitted is not constant, the algorithms for packing and packetizing for ARPA Network applications must be more complex. Also, a tree search is required for decoding. This requires more storage, more time, and a more complex algorithm than would a table-lookup for the simple binary code. It may be possible to combine the trees for a number of parameters, thus reducing the storage required. Because Huffman coding is based on the statistical likelihood of a particular value occurring, good statistics over a fairly large data base must be found. Huffman coding is most useful when several values of the information to be coded are much more probable than the other values.

B. Delta Encoding

The delta encoding scheme codes the change in a parameter from frame to frame. We found this to be useful for parameters, notably pitch and gain, which change slowly but require a large number of quantization levels. In our experiments, we observed that the bit savings with the use of delta encoding by itself were not very significant. However, when we combined delta encoding with Huffman coding (by encoding the changes in parameter values with Huffman code), we achieved significant

savings in bit rate for both pitch and gain. Furthermore, such a combination removes some of the speaker-dependent aspects of these parameters. For example, the change in pitch for a female speaker is likely to be nearer that of a male speaker than are the actual values of pitch. Similarly, delta encoding makes the changes in gain comparable for loud and soft speech. Delta encoding thus improves the statistics for Huffman coding.

C. Statistics for Huffman Coding and Bit Savings

We describe below the data base we used to generate statistics for Huffman coding. Separate statistics for log area ratios, pitch, and gain are briefly described, and transmission rate reductions for these and two other parameters are given (reference [28] gives more details).

1. Data Base

We used 8 sentences, each from a different speaker, of whom 5 were male and 3 female, as a data base. Each sentence was sampled at 10 kHz and passed through a 50 Hz preemphasis filter. Two types of analysis were performed, the first computing and transmitting a new frame of parameters every 20 msec, including pitch and gain (constant frame rate transmission), and the second computing new parameters every 10 msec, but transmitting only when the log likelihood ratio exceeded a threshold of 1.5 dB (variable frame rate transmission). Pitch and gain were

computed and transmitted every 20 msec as in the first type of analysis. Eleven coefficients (fixed order: p=11) were used in both types of analysis. The log area ratios were used for transmission, and they were quantized as described in Section VI-C, using a quantization step size of 1 dB. Pitch and gain were both quantized logarithmically using 6 and 5 bits respectively. Histograms were then compiled for each log area ratio (one each for voiced and unvoiced), and for pitch and gain.

2. Log Area Ratios

For illustration, we have plotted the histogram for the log area ratio g_1 for voiced sounds in Fig. 15 and for g_8 for unvoiced sounds in Fig. 16. Other histograms have been documented in [28]. Briefly, the histograms for g_2 and g_3 for voiced sounds resemble the one in Fig. 15, the main difference being that the skewness of the histogram is to the right for g_2 instead of to the left as for both g_1 and g_3. All other histograms, namely, those for g_4-g_{11} for voiced sounds and for g_1-g_{11} for unvoiced sounds are basically similar to the histogram in Fig. 16. The histograms for variable frame rate transmission were quite similar to those obtained for constant frame rate transmission. In order to reduce the number of trees in Huffman coding, we experimented with combining statistics for several comparable log area ratios and representing them by one tree [28].

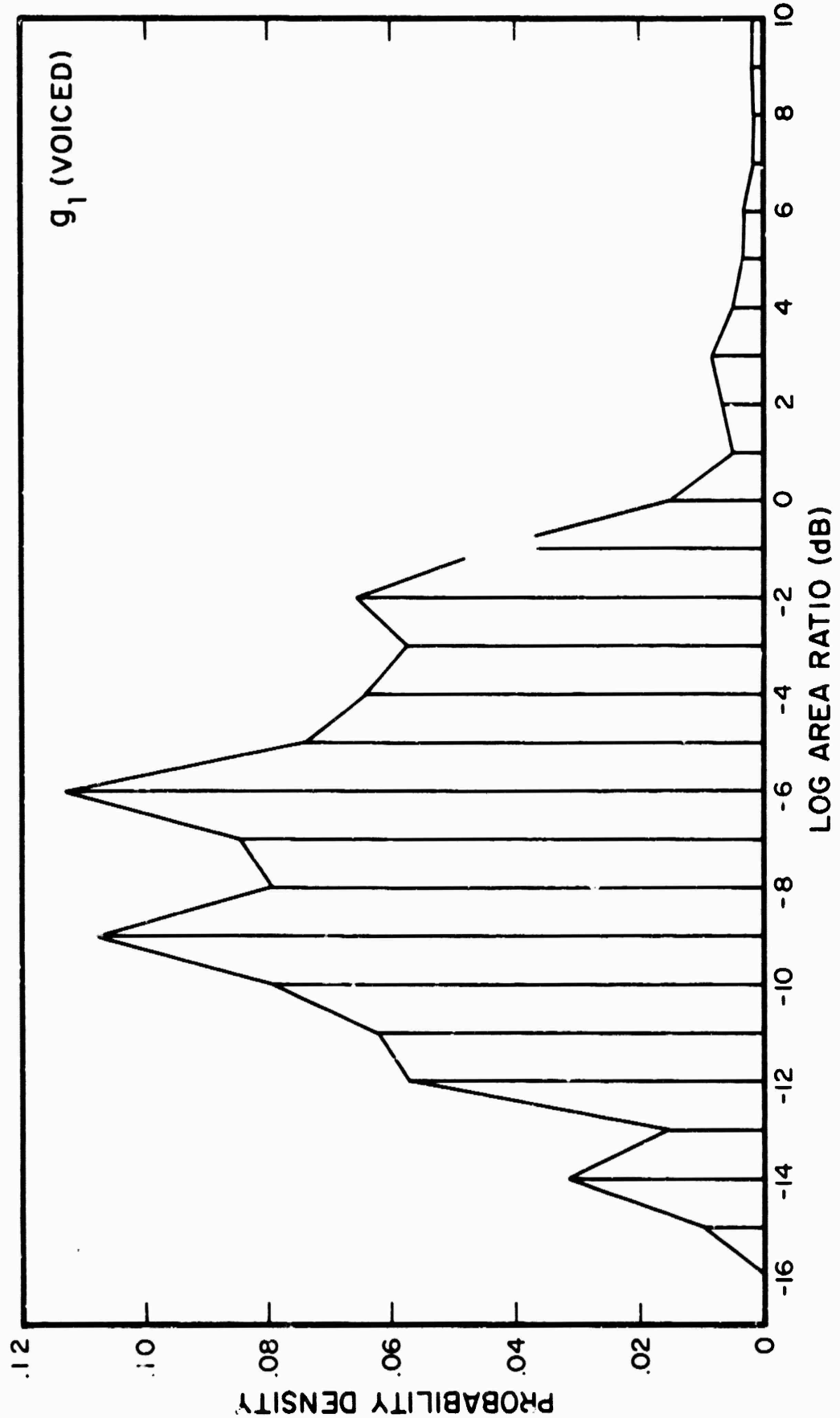


Fig. 15. Histogram of the quantized log area ratio g_1 for voiced sounds.

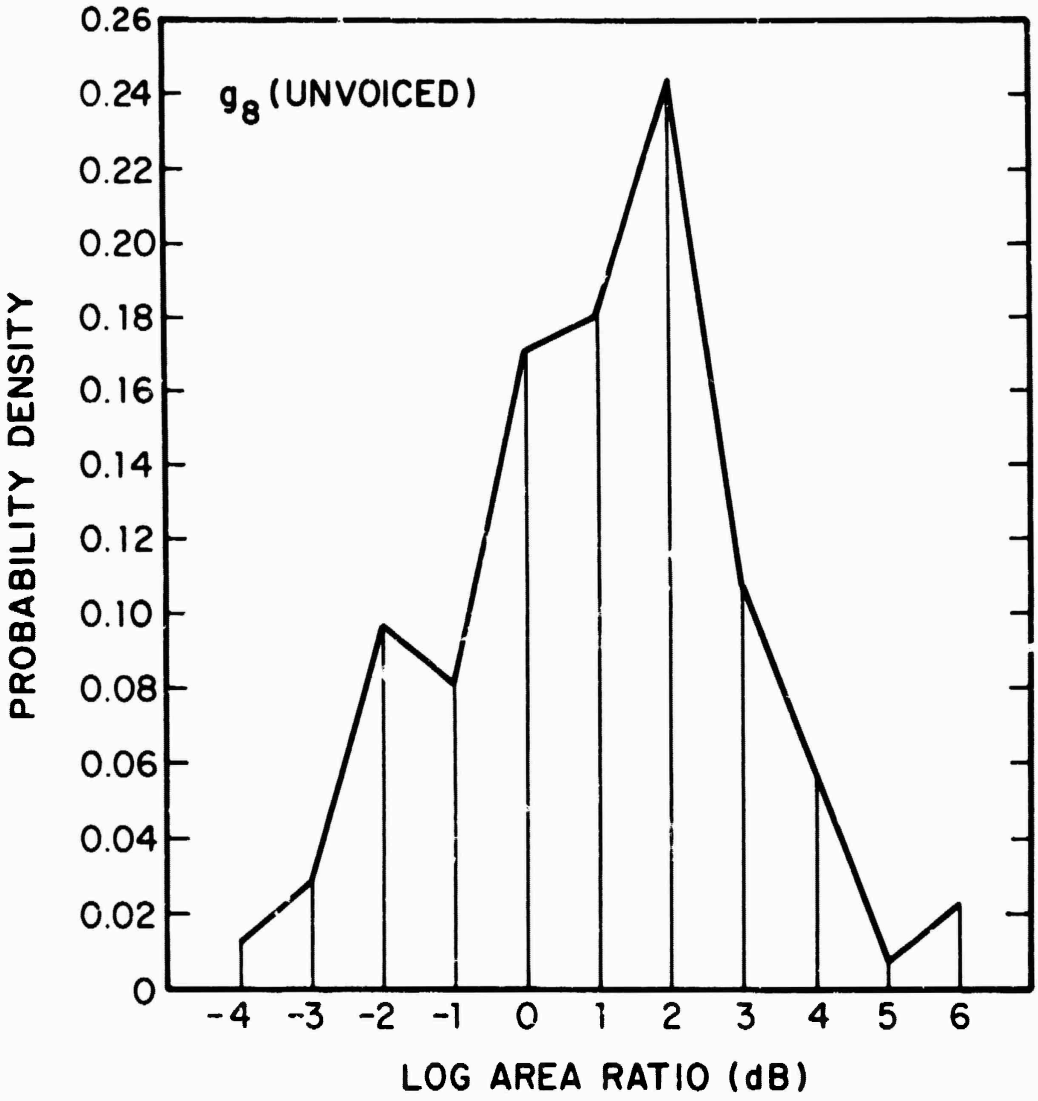


Fig. 16. Histogram of the quantized log area ratio g_8 for unvoiced sounds.

As pitch is, in general, a slowly varying parameter, we investigated coding both the pitch values and the change in pitch values from frame to frame. The changes, or delta values, are much more useful. The number of occurrences of zero change included the unvoiced portions of speech, but also included a good deal of the voiced portions. The average transmission rate for simple binary encoding of pitch was 300 bps (50 frames/sec). With Huffman coding of the actual pitch values, this dropped to 180 bps, and with Huffman coding of the delta values the rate dropped to 130 bps.

We have also experimented with another method of encoding the pitch. This method codes the most likely value with one bit, and uses 7 bits for each of the remaining values. Transmission rates obtained from this experiment were 164 bps for delta values, and 225 bps for actual values. The advantage of this encoding method is that it does not require any tree search for decoding.

4. Gain

The histogram for gain was found to be quite flat, so Huffman coding offered little improvement over simple binary coding. However, using Huffman coding on delta values of gain, we obtained a saving of 75 bps over the binary encoded data rate of 250 bps.

5. Coding Number of Poles

For variable order linear prediction using a maximum value of p=13, simple binary encoding of the number of poles used for analysis required 200 bps for 50 frames/sec transmission and 148 bps for variable frame rate transmission (basic analysis rate: 100 frames/sec). With Huffman coding (see the histogram in Fig. 7), these rates dropped to 159 bps and 117 bps.

6. Coding Transmission Interval

For variable frame rate transmission, the time interval in number of frames needs to be coded and transmitted. For the maximum interval size of 8 frames, the simple binary encoding required 111 bps. With Huffman coding (see the histogram in Fig. 13) it dropped to 93 bps.

IX. SYNTHESIS

In the receiver structure in Fig. 1, the role of decoder is straightforward, so we do not discuss it here. After decoding, deemphasis (or postemphasis) is done on the decoded parameters to undo the effect of preemphasis. For a fixed preemphasis, deemphasis can be performed by passing the synthesized speech through a fixed one-pole filter. For adaptive preemphasis, deemphasis can properly be done only before synthesis. We compute the inverse filter $A'(z)$ (prime denoting the use of decoded parameters) and multiply it by the preemphasis filter $(1-b'z^{-1})$ to obtain the inverse filter for the deemphasized case. So deemphasis increases the order of the predictor by one. The coefficients of the augmented predictor are used for synthesis.

The remainder of this section deals with the different aspects of the synthesizer. These are: excitation source, implementation of the synthesizer, and interpolation and resetting of the synthesizer parameters.

A. Excitation

We use voiced/unvoiced excitation. Voiced excitation consists of unit pulses separated by the received (or interpolated) pitch period. Unvoiced excitation consists of white noise samples (zero mean, unit variance, and uniformly distributed) produced at the sampling frequency using a random

number generator. Referring to Fig. 3b, the excitation signal u_n is multiplied by a suitable gain factor G. G is computed so that the energy of the input signal Gu_n of the synthesizer is equal to the energy of the linear prediction error signal. The latter is given by $R'_o V'_p$ where primes indicate the use of the decoded parameters (see equation (13)). Assuming that R'_o is the signal energy per sample, the gain factor G_u for unvoiced excitation is computed from $G_u^2 = R'_o V'_p$. The gain factor G_v for voiced excitation is computed from $G_V^2 = R'_o V'_p P$, where P is the pitch period in samples.

With the excitation model described above, we found that voiced fricatives such as [z] sounded "buzzy" and unnatural when synthesized using voiced excitation. Ideally such synthesis should use a proper mixture of both types of excitation. However, we obtained satisfactory results by synthesizing voiced fricatives as merely unvoiced sounds. To make this happen automatically, we readjusted the threshold for zero crossing rate used in the pitch extraction scheme at the analysis so that an unvoiced decision would be reached for analysis frames containing voiced fricatives.

Another possible improvement that we studied briefly was to modify the shape of the pulse excitation for voiced sounds. We ran an experiment using Rosenberg's polynomial excitation [29] to test its effect on the quality of the synthesis, but the results were not conclusive

B. Transfer Function

There are at least two ways in which to implement the transfer function of the synthesizer. The recursive filter or canonical form implementation uses the predictor coefficients. The second implementation applies Itakura's ladder structure [8] that uses the reflection coefficients. It has been shown that the second method of transfer function realization results in lower sensitivity to errors caused by finite wordlength computations [30]. Therefore, it should be used in real time implementation employing integer arithmetic. In our non-real-time floating point simulation experiments, we used only the recursive filter implementation. For such a situation, the two methods would give essentially the same results.

C. Parameter Setting and Interpolation

Decoded parameter values as supplied by the vector $\underline{x}'(t)$ in Fig. 1 are used to update or reset the parameters of the synthesizer. There are two types of parameter setting: time-synchronous and time-asynchronous. A particular case of the second is pitch-synchronous updating. Usually, the parameters are reset at a higher rate than the rate of parameter transmission. Thus, some interpolation must be performed between the decoded parameter values. In time-synchronous synthesis, parameters are interpolated and updated at some fixed rate. In pitch-synchronous synthesis, parameter interpolation and setting are done at every pitch pulse for voiced sounds and

time-synchronously for unvoiced sounds.

1. Time-Synchronous Versus Pitch-Synchronous Synthesis

We investigated both time-synchronous synthesis and pitch-synchronous synthesis in our experiments. Since pitch and filter parameters are not updated simultaneously in time-synchronous synthesis, one might suspect that this would introduce undesirable transients in the synthesized speech. However, from the large number of experiments that we performed, we found that the synthesized speech did not have any such transients. As an example, Fig. 17 shows the waveform of segments of speech synthesized time synchronously (vertical lines mark the instances when parameter updating was done). Further, our comparative study of time-synchronous synthesis and pitch-synchronous synthesis showed that the quality of the synthesized speech was actually better for time-synchronous synthesis in some experiments, while in others it remained essentially the same for both cases. In general, we found that speech quality was best when the synthesizer parameters were updated at a time corresponding to the time when they were extracted in the analysis. Thus, if time-synchronous analysis is used, time-synchronous synthesis should also be used.

Another reason why time-synchronous synthesis produced better speech quality follows from the result reported in Section XII-A that interpolation is a major source of error. This is not to suggest that interpolation should never be done.

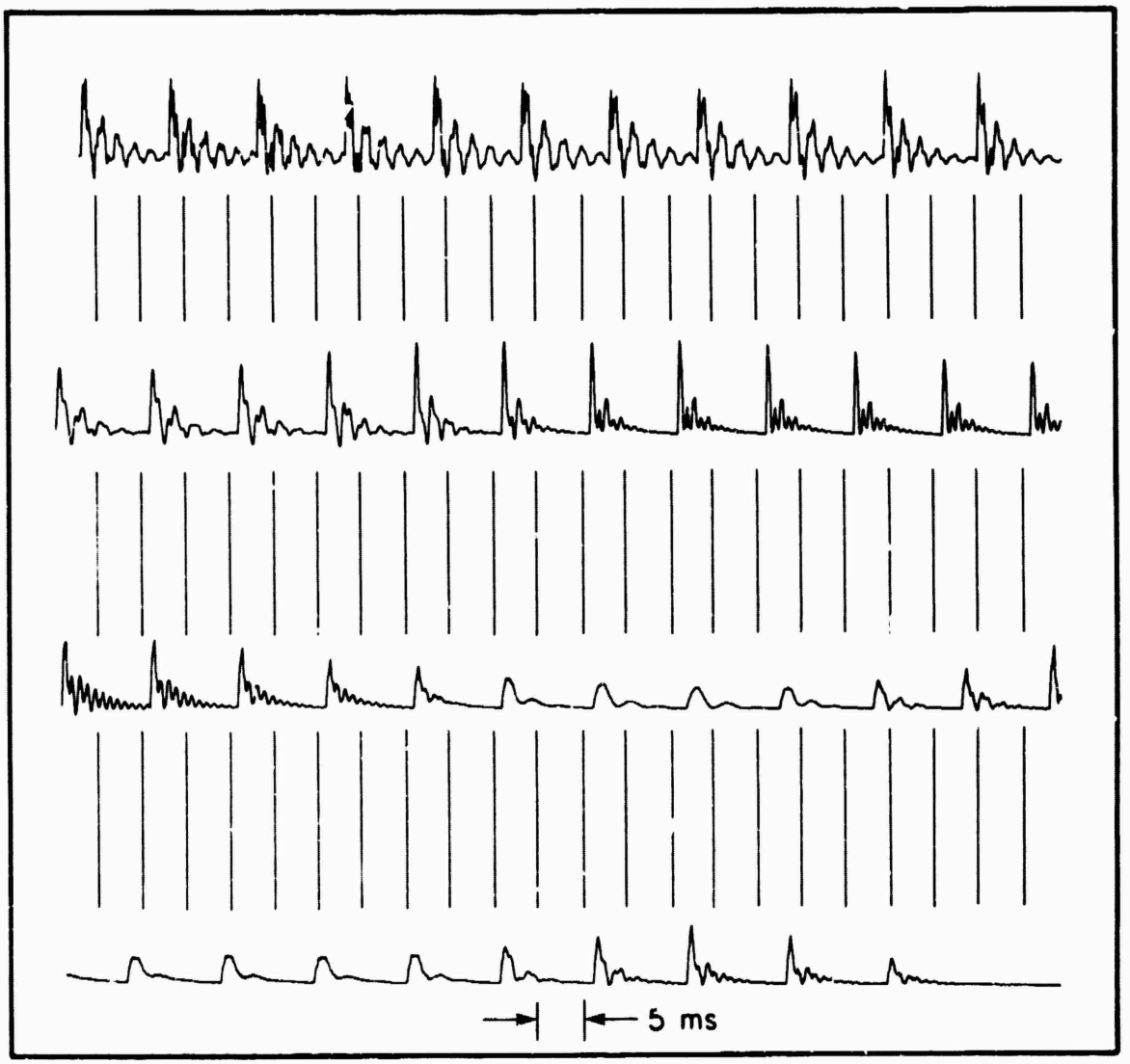


Fig. 17. Four segments of speech synthesized time synchronously. The vertical lines (5 msec apart) mark the instances when parameters of the synthesizer were updated.

In fact, for the 50 frames/sec constant frame rate transmission (2650 bps), use of interpolation (time-synchronous or pitch-synchronous) certainly improved the speech quality. Our finding cited above should rather be interpreted as a caution to use interpolation only if needed. Returning to the two synthesis approaches, if time-synchronous analysis is used, then pitch-synchronous synthesis would require interpolation every pitch period in general. For time-synchronous synthesis, however, no interpolation is needed for those instances at which analysis parameters have been extracted and transmitted. The fact that less interpolation is performed in time-synchronous synthesis perhaps explains the resulting improvement in speech quality over pitch-synchronous synthesis.

It should be recalled that in our study we used the recursive filter implementation of the synthesizer. We expect the results to come out similar even when the ladder structure is used.

2. Interpolation Study

In our speech compression system, we interpolated both pitch and energy logarithmically. For the synthesizer filter, we used different sets of parameters for linear interpolation. These included reflection coefficients, log area ratios and autocorrelation coefficients of the all-pole filter. Stability of the filter is preserved under interpolation in all the three cases. The different interpolation parameters resulted in

slight differences in the spectrum of the linear prediction filter, but the quality of the synthesized speech as judged from informal listening tests did not show any perceivable differences. In view of the lack of difference in speech quality between the different interpolation parameters, we used, in many of our recent experiments, log area ratios for interpolation since they were available directly from the decoder.

X. SIMULATION OF SPEECH COMPRESSION SYSTEM

In previous sections we frequently alluded to our simulation experiments. We briefly describe below some of the details of our simulated speech compression system. We also give typical transmission rates when using the different techniques discussed above separately and in different combinations. A result of significant importance is that when we incorporated all our bit-saving techniques, we obtained good quality speech at average rates of 1500 bps.

A. Software Simulation

We simulated the entire speech compression system with its many different variations on our time-sharing computer facility comprising two PDP-10 computers called System A and System B. All computations were done on System A using 36-bit word floating point arithmetic. The analog-to-digital converter (ADC) and the digital-to-analog converter (DAC) were located in System B. Digitizing speech from a tape and recording (or listening to) the synthesized speech were done on System B in single user mode so as to provide the fast service rate needed for these real-time functions. Sampled speech files and synthesized speech files were transferred between the two computer systems using the ARPA Network link. In all the developmental and simulation phases of our work, we heavily used the IMLAC PDS-1 display facility, which can be operated from either of the two systems.

The synthesized speech was passed through a 12-bit DAC and low-pass filtered sharply at 5 kHz before presenting it to listeners or recording it on tape.

Due to limitations of our pitch extraction scheme, occasionally (less than 1% of the time) we encountered pitch errors (mainly pitch doubling errors, especially for high-pitched female speakers). Since we were principally interested in testing many of the quantization and encoding methods, we hand edited these errors to prevent the pitch discrepancies from biasing the listening judgments. The overall time taken for analysis and synthesis in our simulation system was about 50-60 times real time.

B. Typical Transmission Rates

For the data given below, we used p=11 for fixed order linear prediction and a maximum p=11 for variable order linear prediction. Pitch and gain were quantized using 6 bits and 5 bits respectively. Log area ratios were quantized using the data given in Table I. For the fixed order case this required, before further encoding, 41 bits/frame for unvoiced sounds and 43 bits/frame for voiced sounds. The log likelihood ratio threshold used for the variable frame rate system was 1.5 dB. When variable wordlength encoding was used, we used Huffman coding and delta encoding as mentioned in Section VIII. Several different speech compression systems and their average transmission rates are given in Table II.

TABLE II

System No.	Analysis Frame Rate (frames/sec)	Order	Transmission Frame Rate	Huffman Coding	Average Transmission Rate (bps)
1	50	Fixed	Fixed	No	2650
2	50	Fixed	Fixed	Yes	2000
3	50	Variable	Fixed	Yes	1750
4	100	Fixed	Fixed	No	5300
5	100	Fixed	Variable	No	2150
6	100	Fixed	Variable	Yes	1650
7	100	Variable	Variable	Yes	1500

Referring to Table II, the quality of the synthesized speech differed little between systems 1, 2 and 3, and between 5, 6 and 7. Speech quality of system 5 was only slightly lower than that of system 4, in spite of a reduction in the transmission rate by a factor of about 2.5. This illustrates the successful performance of our variable frame rate transmission scheme. Although the bit rate of system 5 is lower than that of system 1 by about 20%, speech quality was found to be actually better for system 5 than for system 1. This suggests that starting with a higher analysis rate and transmitting only when necessary produces a better dynamic modeling of speech from the point of view of perception.

Using system 6 in Table II, we processed a tape containing an 11-sentence dialogue provided by the Stockholm Speech Communication laboratory to the participants of the 1974 Stockholm Speech Conference. The dialogue, which is between a female telephone operator (pitch range 108-417 Hz) and a male customer (pitch range 67-323 Hz), provides difficult test material for any vocoder. Using this dialogue, we demonstrated good quality synthesized dialogue at 1650 bps at an ARPA Network Speech Compression (NSC) meeting and at two other conferences [31,32].

XI. REAL TIME IMPLEMENTATION

In the second year of our speech compression project, we worked in cooperation with the other sites in the ARPA community towards implementation of a linear predictive vocoder that transmits speech in real time over the ARPA Network. This work has not yet been completed. In this section, we summarize the work we have done thus far.

A. Signal Processing System

To date, our effort in the development of a signal processing system for implementing the vocoder has been largely a matter of system definition and information exchange. In defining the system, we have considered the needs of both the speech compression project and the speech understanding project. The requirements of these projects indicate that the system will have three distinct purposes: (a) To function as a real-time data acquisition system, supporting ADC's and DAC's at sampling rates up to 20 kHz, and providing a means of storing and retrieving speech utterances; (b) To allow the implementation of real-time speech analysis and synthesis, and to make possible the transfer of coded speech over the ARPA Network; (c) To provide signal processing computational power to multiple users, functioning as a peripheral to our TENEX time-sharing system and serving to remove some of the computing load from it.

In cooperation with the other sites involved in these two projects, we have been investigating various items of hardware and software with which to implement the system. This cooperation has resulted in the network-wide selection of the SPS-41 as the signal processing computer, one of the most critical parts of the system. This cooperation has also given the sites involved a considerable leverage with the manufacturer, Signal Processing Systems (SPS), resulting in significant hardware and software improvements to the original version of the SPS-41. These improvements included a dual-port memory option and the availability of a double-precision autocorrelation routine. Our complete signal processing system comprising the SPS-41 and PDP-11 will be interfaced to the ARPA Network.

We have disseminated informa ion as it became available from the manufacturers, particularly SPS, and exchanged information with the other sites in order to avoid duplication of effort and ensure maximum compatibility.

Support software for the SPS-41

In close cooperation with the Information Sciences Institute, we have been working on an on-line loader system for the SPS-41. The software package supplied by SPS includes only an off-line loader which is unsuitable for real-time applications.

The on-line system consists of two parts, the Overlay Executive (EXEC) and the Automatic Reformatter (ARF). The EXEC is an SPS-41 program which loads information from the PDP-11 into the SPS-41. ARF reformats the output of the SPS-41 assembler in a way acceptable to the EXEC. It also provides a mechanism for attaching meaningful labels to SPS-41 program segments and locations. A user's guide for ARF is currently being prepared.

B. Variable Speed Speech

In the implementation of the receiving end or synthesizer of the speech compression system, it is necessary to have a buffer whose size will depend on the expected maximum delay in the network transmission of the vocoded bit stream. However, there might be times when the expected maximum delay is exceeded. In such cases, the buffer at the synthesizer will be empty for some period of time till the data appears again. The data might then arrive at such a rate that the buffer overflows. This is an undesirable situation since speech will be lost. One solution to this problem is to speed up the rate of speech synthesis until the condition is normal again. Also, before the buffer runs out of data, it might be desirable to slow down the rate of synthesis until the data arrives. However, this involves predicting in advance the occurrence of the excessive delay. We have already demonstrated the feasibility of variable speed synthesis in an NSC meeting (May 1974). This method

involved merely redefining the duration of the synthesis frame appropriately. The effectiveness and acceptability of this method to ameliorate the consequences of excessive network delays should be tested.

XII. MISCELLANEOUS T

Some additional issues that we have investigated are reported in this section.

A. Measures for Objective Evaluation of Speech Quality

The following two reasons motivated us to develop measures for objective evaluation of speech quality.

1. Evaluation of speech quality has been done mostly through subjective listening tests. It would be desirable to develop objective measures that correlate well with the scores obtained in subjective listening tests. Besides their theoretical appeal, these measures would ensure uniformity in evaluation as well as enable the evaluation to be done by computer. Also, they can be used in the design of better speech quality vocoders. While there exist methods in the literature for objectively evaluating the intelligibility of speech in the presence of stationary noise [33], little has been done regarding the objective evaluation of either the intelligibility or the quality of vocoded speech.

2. In many of our experiments, we specifically observed that a change in speech quality due to any one improvement in quantization, interpolation, etc., was most often not perceivable, while when several such improvements were added together there was a clearly perceivable improvement in speech

quality. It would be helpful therefore to incorporate some objective measures of speech quality within the speech compression system, which would generate performance scores and hence enable one to make relative judgments of the smaller differences.

For our linear predictive speech compression system we used the two measures, (i) log-area-ratio error measure and (ii) spectral error measure, to determine, for a given speech frame, the deviation between the synthesized speech and the original speech. Based on the error data computed for a large number of speech sounds, appropriate measures could be developed for speech quality evaluation. Below we describe how we computed the error data in our simulation system. Speech parameters were extracted at a rate of 200 frames/sec to provide the reference data with enough resolution in time. They were quantized and transmitted at a lower rate required by the specific compression system under evaluation. After decoding, the parameters were interpolated to produce the test data of 200 frames/sec. The log area ratio error was obtained as the average of the absolute differences between the sets of log area ratios from the reference and test data. The spectral error was computed by averaging in frequency the absolute differences between the values of the log spectra of the linear predictor with coefficients from the reference and test data. For each measure, the time history of the error within a speech utterance, the time-averaged value of the error and its

variance, and the maximum error observed could all be potentially useful in the quality evaluation. We did not get an opportunity thus far to compare these scores with formal subjective evaluation results. However, by investigating the measures (i) and (ii) for several linear predictive vocoders and diverse speech utterances, we obtained the following two results.

(a) The error (log area ratio or spectral) due to interpolation was much larger than the error due to quantization. This result was used to interpret the quality difference between time-synchronous and pitch-synchronous methods of synthesis (Section IX). An important inference suggested by the result is that better parameter interpolation approaches than the simple linear scheme should be developed.

(b) Scatter plots between the spectral and the log area ratio error measures obtained using different quantization step sizes indicated a fairly strong linear relationship between the two measures. Fig. 18 shows the scatter plot obtained from 4 speech utterances and using 4 quantization step sizes (0.5,1,2,3 dB). The coefficient of correlation for the least squares linear fit also shown in Fig. 18 was 0.88. A useful implication of this result is that the log area ratio error measure can be substituted wherever the spectral error measure is needed, which is computationally advantageous. As an example, we cite the use of the log

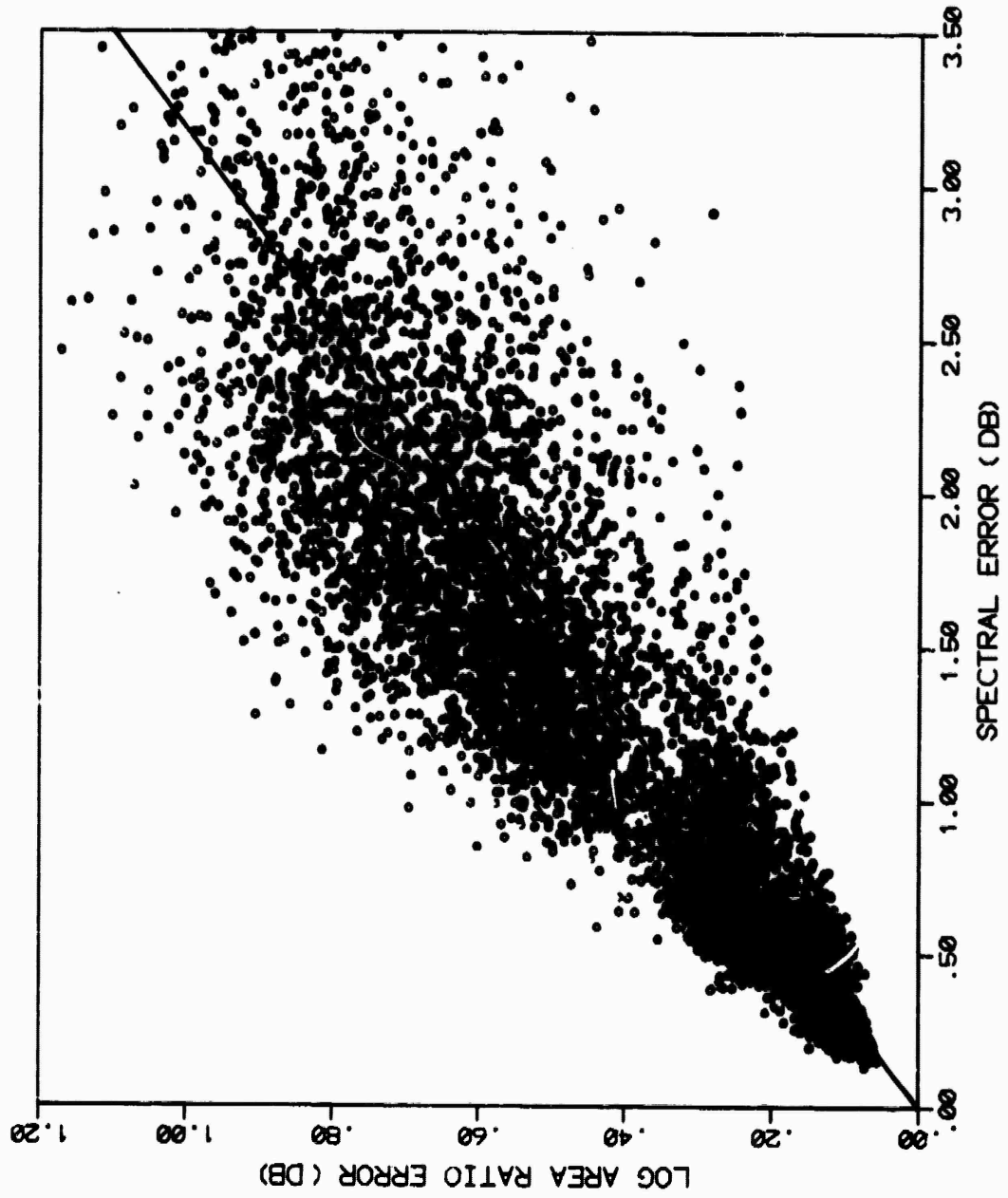


Fig. 18. Scatter plot of log area ratio error versus spectral error. The linear characteristic passing through the origin is the best least squares fit of a linear relationship between the two errors.

area ratio measure in variable frame rate transmission for detecting changes in the speech spectrum.

B. Variable Sampling Frequencies

It is often desirable to be able to test the performance of a speech compression system at different sampling rates, without actually sampling at all those rates. This can be achieved by applying our recently developed method of selective linear prediction [24] to speech sampled at only the highest desirable rate. Briefly, in this method, the spectral matching properties of the autocorrelation method [6,7] are used to model a selected portion of the speech spectrum by an all-pole spectrum. The method is based on the idea that the autocorrelation coefficients in (8) can be computed from the spectrum instead of the signal. Thus, the speech signal is sampled at the highest rate and the short-time spectrum is computed for every frame. Different sampling rates can then be simulated by computing the autocorrelation coefficients from the part of the spectrum corresponding to each sampling rate. The problems of sharp filtering and down sampling that exist in time domain methods can therefore be avoided by simply working in the frequency domain.

C. Formant Bandwidth Correction

When the spectral characteristics of the synthesized speech were compared with those of the original speech, it was observed that occasionally the bandwidths of the formants of the

synthesized speech were relatively large. This was also manifested in the time waveform as rapidly decaying sinusoids. Such increases in bandwidths were found to occur mainly in nasals and nasalized vowels. This phenomenon is perhaps due to the limitations of the linear prediction method which assumes an all-pole filter. Thus, a pole-zero-pole cluster in a nasal sound may get represented by a wide bandwidth formant in the linear prediction method. Synthesis experiments were conducted where, when necessary, bandwidth corrections were made after interpolation of the synthesizer parameters. The biggest problem that was encountered was the need to identify the ordered formants. Any error in such identification was found to introduce undesirable "blips" in the synthesized speech. Even when the formants were determined accurately, informal listening tests did not indicate any significant improvement in quality by the bandwidth correction.

D. Parameter Smoothing

A study of the time series of analysis parameters (energy, pitch and reflection coefficients) indicated that occasionally some parameters changed rather rapidly, possibly contributing to the "roughness" or "unevenness" that was sometimes present in the synthesized speech. These rapid variations can be smoothed out by proper low-pass filtering. We used a three-point

smoothing filter with weights 0.25, 0.50 and 0.25. Smoothing was done just prior to interpolation at the synthesizer. Genuine jumps in parameter values were preserved by not smoothing in transitions between voiced and unvoiced sounds, and when parameter changes exceeded preset thresholds (10 Hz for pitch and 3 dB for energy). Identical low-pass filters were used for smoothing pitch, energy and reflection coefficients (or log area ratios).

Several synthesis experiments were performed using smoothed parameters. Informal listening tests showed that with smoothing, speech quality improved in some instances in that the "unevenness" observed without smoothing disappeared. But, smoothing also made the synthesized speech sound less "crispy" or more "smeared" in many instances.

XIII. CONCLUSIONS

For linear predictive speech compression systems, we have developed many methods of reducing the redundancy in the speech signal while maintaining good speech quality at the synthesis. Included among these methods are preemphasis of the incoming speech, adaptive optimal selection of predictor order, optimal selection and quantization of transmission parameters, variable frame rate transmission, optimal encoding, and improved synthesis methodology. When we incorporated all of these in a floating point simulation of linear predictive vocoder, we obtained synthesized speech with high quality at transmission rates as low as 1500 bps.

Speech quality would perhaps decrease when the vocoder is implemented in real time using relatively small wordlength computers or hardware. This would necessitate development of still other methods of improving speech quality. We feel that such improvements can be achieved by developing interpolation schemes which track the dynamic behavior of speech more closely.

REFERENCES

[1] Atal, B.S. and S.L. Hanauer, "Speech Analysis and Synthesis by Linear Prediction of the Speech Wave," J. Acoust. Soc. Am., vol. 50, pp. 637-655, 1971.

[2] Makhoul, J.I. and J.J. Wolf, "Linear Prediction and the Spectral Analysis of Speech," NTIS No. AD-749066, BBN Report No. 2304, Bolt Beranek and Newman Inc., Cambridge, Mass., 237 pp., Aug. 1972.

[3] Faddeev, D.K. and V.N. Faddeeva, Computational Methods of Linear Algebra (English transl. by R.C. Williams). San Francisco: W.H. Freeman, pp. 144-147, 1963.

[4] Markel, J.D., "Digital Inverse Filtering - a New Tool for Formant Trajectory Estimation," IEEE Trans. Audio Electroacoust., vol. AU-20, pp. 129-137, June 1972; also, SCRL Monograph 7, Speech Commun. Res. Lab., Santa Barbara, Calif., Oct. 1971.

[5] Robinson, E.A., Statistical Communication and Detection, New York, pp. 274-279, 1967.

[6] Itakura, F. and Saito, S., "A Statistical Method for Estimation of Speech Spectral Density and Formant Frequencies," Electron. Commun. (Japan), vol. 53-A, no. 1, pp. 36-43, 1970.

[7] Makhoul, J., "Spectral Analysis of Speech by Linear Prediction," IEEE Trans. Audio Electroacoust., vol. AU-21, pp. 140-148, June 1973.

[8] Itakura, F., et al, "An Audio Response Unit Based on Partial Autocorrelation," IEEE Trans. Comm., vol. COM-20, pp. 792-797, Aug. 1972.

[9] Itakura, F. and S. Saito, "On the Optimum Quantization of Feature Parameters in the PARCOR Speech Synthesizer," Conf. Record, 1972 Conf. on Speech Comm. and Processing, Newton, Mass., pp. 434-437, Apr. 1972.

[10] Wakita, H., "Direct Estimation of the Vocal Tract Shape by Inverse Filtering of Acoustic Speech Waveforms," IEEE Trans. Audio Electroacoust., vol. AU-21, pp. 417-427, Oct. 1973.

[11] Atal, B.S. and M.R. Schroder, "Adaptive Predictive Coding of Speech Signals," The Bell System Technical Journal, vol. 49, pp. 1973-1986, Oct. 1970.

[12] Magill, D.T. and C.K. Un, "Residual Excited Linear

Predictive Vocoder," presented at the 87th meeting of the Acoust. Soc. Amer., New York, April 23-26, 1974.

[13] Haskew, J.R., J.M. Kelly, and T.H. McKinney, "Results of a Study of the Linear Prediction Vocoder," IEEE Trans. Comm., vol. COM-21, pp. 1008-1014, Sept. 1973.

[14] Markel, J.D. and A.H. Gray, Jr., "A Linear Prediction Vocoder Simulation Based Upon the Autocorrelation Method," IEEE Trans. Acoustics, Speech and Signal Processing, vol. ASSP-22, pp. 124-134, Apr. 1974.

[15] Markel, J.D. and A.H. Gray, Jr., "On Autocorrelation Equations as Applied to Speech Analysis," IEEE Trans. Audio Electroacoust., vol. AU-21, pp. 69-79, Apr. 1973.

[16] Grenander, U. and G. Szego, Toeplitz Forms and Their Applications, Berkeley: Univ. Calif. Press, 1958.

[17] Sondhi, M.M., "New Methods of Pitch Extraction," IEEE Trans. Audio Electroacoust., vol. AU-16, pp. 262-266, June 1968.

[18] Akaike, H., "Use of an Information Theoretic Quantity for Statistical Model Identification," Proc. 5th Hawaii Intl. Conf. on System Sciences, pp. 249-250, 1972.

[19] Akaike, H., "A New Look at the Statistical Model Identification," IEEE Trans. Automatic Control, Dec. 1974.

[20] Makhoul, J. and R. Viswanathan, "Adaptive Preprocessing for Linear Predictive Speech Compression Systems," Presented at the 86th meeting of the Acoust. Soc. Amer., Los Angeles, Oct. 30 - Nov. 2, 1973 (also ARPA NSC Note 5).

[21] Viswanathan, R. and W. Russell, "Quantization Properties for Linear Predictive Vocoders," ARPA NSC Note 33, July 1974.

[22] Gold, B. and C.M. Rader, Digital Processing of Signals, McGraw-Hill, New York, 1969.

[23] Makhoul, J. and R. Viswanathan, "Quantization Properties of Transmission Parameters in Linear Predictive Systems,", BBN Report No. 2800, Bolt Beranek and Newman Inc., Cambridge, Mass., Apr. 1974.

[24] Makhoul, J., "Selective Linear Prediction and Analysis by Synthesis in Speech Analysis," BBN Report No. 2578, Bolt Beranek and Newman Inc., Cambridge, Mass., Apr. 1974.

[25] Itakura, F., "Minimum Prediction Residual Principle Applied to Speech Recognition," Proc. IEEE Symposium on Speech

Recognition, CMU, Pittsburgh, PA., pp. 181-185, Apr. 1974.

[26] Huffman, D.A., "A Method for the Construction of Minimum-Redundancy Codes," Proc. of the I.R.E., vol. 40, pp. 1098-1101, Sept. 1952.

[27] Schwartz, E.S., "An Optimum Encoding with Minimum Longest Code and Total Number of Digits," Information and Control, vol. 7, pp. 37-44, 1964.

[28] Cosell, L. and J. Makhoul, "Variable Wordlength Encoding," ARPA NSC Note 34, Aug. 1974 (Also presented at the 88th meeting of the Acoust. Soc. Amer., St. Louis, Nov. 7-10, 1974).

[29] Rosenberg, A.E., "Effect of Glottal Shape on the Quality of Natural Vowels," J. Acoust. Soc. Amer., vol. 49, no. 2 (part 2), pp. 583-590, 1971.

[30] Crochiere, R., "Digital Ladder Filter Structures and Coefficient Sensitivity," Report No. 103, Res. Lab. Electronics, MIT, Cambridge, Mass., Oct. 1971.

[31] Viswanathan, R. and J. Makhoul, "Current Issues in Linear Predictive Speech Compression," Proc. 1974 EASCON Conf., Washington, D.C., pp. 577-585, Oct. 1974.

[32] Viswanathan, R. and J. Makhoul, "Towards a Minimally Redundant Linear Predictive Vocoder," Presented at the 88th meeting of the Acoust. Soc. Amer., St. Louis, Nov. 7-10, 1974.

[33] Kryter, K.D., The Effects of Noise on Man. New York: Academic Press, 1970.