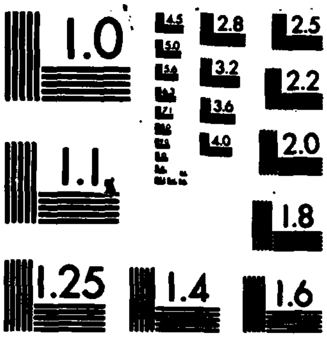
AD-A137 569 FAST ALGORITHMS FOR IMPROVED SPEECH CODING AND RECOGNITION(U) STANFORD UNIV CA INFORMATION SYSTEMS LAB J M TURNER ET AL. 31 DEC 83 ISL-M736-3 N00014-82-K-0492 1/2

UNCLASSIFIED F/G 12/1 NL


1.0
4.5
5.0
5.6
6.3
7.1
8.0
2.8
3.2
3.6
4.0
2.5
2.2
2.0
1.1
1.8
1.25
1.4
1.6

2

INFORMATION SYSTEMS LABORATORY

STANFORD ELECTRONICS LABORATORIES
DEPARTMENT OF ELECTRICAL ENGINEERING
STANFORD UNIVERSITY · STANFORD, CA 94305

FAST ALGORITHMS FOR IMPROVED SPEECH CODING AND RECOGNITION

Final Technical Report to the

Office of Naval Research

Contract Number : N00014-82-K-0492

Reporting Period : 1 August 1982 - 31 December 1983

Issued: 31 December 1983

John M. Turner

ISL REPORT M736-3

FAST ALGORITHMS FOR IMPROVED SPEECH CODING AND RECOGNITION

Final Technical Report to the

Office of Naval Research

Contract Number : **N00014-82-K-0492**

Reporting Period : **1 August 1982 - 31 December 1983**

Issued: **31 December 1983**

John M. Turner

ISL REPORT M736-3

ABSTRACT

This report summarizes the research activities at the Information Systems Laboratory, Stanford University, for the "Fast Algorithms for Improved Speech Coding and Recognition" project during the past sixteen months. This research effort has studied estimation techniques for processes that contain Gaussian noise and jump components, and classification methods for transitional signals by using recursive estimation with vector quantization. The major accomplishments presented are an algorithm for joint estimation of excitation and vocal tract response, a pitch pulse location method using recursive least squares estimation, and a stop consonant recognition method using recursive estimation and vector quantization.

M. Morf - Principal Investigator and J. Turner - Research Associate

W. Stirling, J. Shynk, and S-S. Huang

TABLE OF CONTENTS

1. INTRODUCTION

During the course of this research contract, estimation techniques for processes that contain Gaussian noise and jump components, and classification methods for transitional signals by using recursive estimation with vector quantization were studied. These signal processing tools have possible application to a wide range of physical signals, although this research studied their use for speech processing. The major accomplishments presented are an algorithm for joint estimation of excitation and vocal tract response, a pitch pulse location method using recursive least squares estimation, and a stop consonant recognition method using recursive estimation and vector quantization.

JOINT ESTIMATION OF EXCITATION AND VOCAL TRACT RESPONSE

Historically, the development of estimation theory and signal modeling techniques have usually presumed that the processes involved had Gaussian statistics. Most naturally occurring processes tend to be Gaussian. However, many man-made signals have additional components that can be characterized as harmonic structures or jump processes or impulsive noise. For example, the rotating blade in an aircraft generates an artifact when the blade crosses the wing, likewise the main rotor and tail rotor of helicopters produce signals depending on the orientation of the fuselage, underwater acoustical signals from man-made sources, radar/sonar returns generated by pulsed sources, and in general any signal that has been processed in a nonlinear fashion are within the class of non-gaussian signals.

Estimation techniques were developed for signals composed of a Gaussian noise component and a jump process component driving a linear system. In particular, simultaneous estimation of the system parameters (ARMA) and the jump excitation were introduced. The technique evolved from simple pulse in noise detection to composite pulses and noise from an ARMA structured system. A decision-directed approach was used to estimate the unknown prior statistics of the pulse process. A full description of these techniques was presented in the first ONR technical report, M736-1, Feb. 1983.

In this study, the estimation technique was applied to speech signals attempting to improve the estimate of pitch and vocal tract response. Most speech modeling techniques handle the response and excitation separately. The semiperiodic opening of the vocal chords emits a pulse of air to excite the vocal tract (throat, tongue, and mouth) provides an example of jump and noise excitation that has been much studied. The complex interaction of the vocal chords, vocal tract and nose, warrant simultaneous estimation of the response function and the excitation.

PITCH DETECTION BY LEAST SQUARES LATTICE ALGORITHM

There are many advantages of recursive estimation techniques and particularly when implemented in the form of a lattice filter. An overview of recursive least squares estimation and lattice filters was presented in the second ONR technical report, M736-2, Jan. 1984. Within the Least Squares Lattice algorithm, a "likelihood" variable is calculated which indicated the occurrence of unexpected or non-gaussian components in the signal. The derivative of this variable multiplied by other signal parameters appears to be a good detector of pitch pulses in speech. The development and experimental results of this pitch detection method are presented.

RESEARCH ON RECOGNITION OF STOP CONSONANT

Recursive Estimation and Vector Quantization have been two very active areas of research in the last few years. Each area has developed new mathematical tools for analyzing and characterizing signals. These techniques are trying to satisfy different objectives; adaptive signal modeling or efficient signal quantization, respectively. However there is a natural marriage of these two powerful mathematical tools that often provides a more appropriate solution to problems in signal modeling, coding, and classification.

For adaptive speech modeling, the time varying nature of speech requires that quickly changing burst sounds as well as fairly steady vowels sounds be efficiently approximated. The recursive orthogonalizing properties of the ladder structure allow speech transitions to be tracked precisely while still yielding consistent parameters for steady sounds. Recursive estimation generates a full signal model for each data sample causing a considerable increase in the number of parameters handled. For coding and transmission of signals, the recursive estimation generates a good signal model but the problem of efficient parameter encoding remains. In coding or classification applications, only a small number of 'states of the world' are of interest rather than the continuum of parameter values generated by RLS.

Vector Quantization (VQ) design algorithms have been used to design low bit rate data compression and data classification systems. For speech recognition, vector quantization techniques have been developed for speaker dependent and independent word recognition. VQ is well suited for data compression or data classification once the codewords have been determined from a representative training data set.

Experiments on combining recursive estimation and vector quantization were begun in this ONR contract. Using recursive estimation to track changing signal characteristics and vector quantizations to systematically classify the resulting parameter, brings together adaptive processing with limited state output. This idea was first applied to speech for recognition of transitional sounds, which are currently very difficult to distinguish. This approach acknowledges that speech contains only a finite number of identifiable sound units (in each language), but that some sounds happen quite quickly. This type of classification technique distinguishes transitional states in the signal that are themselves of interest.

A classification scheme using parameter trajectory information was developed that allows transitional signal characteristics to be identified. The transitions in the data can be tracked using recursive estimation rather than being coarsely approximated by LPC (or equivalent) parameterizations from fixed speech windows. By having a signal model at every data sample, the trajectory of the parameters can be readily determined. This new information assisted in determining transitional components from steady state components.

A classified vector quantization approach was also developed that allows quantization precision to be specified for various signal components. No longer must the steady state signal components dominate the vector quantized states. The results for recognizing stop consonants within a limited test are very encouraging.

2. JOINT ESTIMATION OF EXCITATION AND VOCAL TRACT RESPONSE

2.1 INTRODUCTION

Many speech analysis techniques attempt to deconvolve the speech waveform into an excitation component and a response function. The standard approach is to estimate the vocal tract model parameters first and then the excitation signal from the residual errors (or directly from a bandlimited version of the original speech signal). A new approach for simultaneously identifying the system model parameters and detecting the unobserved random pulse-type inputs has been developed. A key component of this procedure is the application of a new decision-directed algorithm to estimate the period of the pitch pulse process. This decision-directed algorithm incorporates an exact, recursive estimator to compute the rate of a discrete-time point process used to characterise the arrival times of the pitch pulse process. An overview of this approach is presented here. The complete description was contained in the first ONR technical report, M736-1.

A common assumption of speech analysis is that a speech waveform can be modeled as the output of a linear system driven by an approximately Gaussian noise part (for unvoiced speech) plus a jump component, (periodic pulses for voiced speech). Typically, it is assumed that the linear system used to model the vocal tract consists of an all-pole filter (an autoregressive or AR representation) whose coefficients are slowly time-varying. The all-pole model used to characterize the vocal tract and the mixed driving process (a white Gaussian noise plus a pulse process) admits the representation

$$y_t + \sum_{i=1}^{k} a_i y_{t-i} = \sum_{i=1}^{k} b_i n_{t-i} + b_0 n_t + v_t . \qquad (2.1)$$

where $\{y_t\}$ is the observed speech waveform, $\{n_t\}$ is a binary (0,1) sequence denoting the epochs of the pitch pulses, $\{v_t\}$ denotes a WGN process, and the coefficients a_i and b_i denote the model coefficients. The estimation/detection problem is to simultaneously estimate these coefficients and to detect the occurrence of the pulses (i.e., to detect the events $n_t = 1$). Standard least squares techniques are used to estimate the a_i coefficients. The unique aspects of this analysis consist of the approach used to detect the events $n_t = 1$ and the joint estimation of the model parameters and detection of the pulse input (excitation).

The estimation of the b_i coefficients follows in a straightforward manner once the pulses have been detected. The detection problem is rendered difficult by the absence of reliable *a priori* information about the probability of the event $n_t = 1$. The problem of binary detection with unknown priors leads to the application of so-called decision-directed (DD) detectors. DD detectors [DS], [KD] use the results of the past decisions to estimate the rate (i.e., the *a priori* probability) of the signal, which is used to adjust the parameters of the detector (assuming that the previous decisions were correct). A method of dealing with nonstationary priors in a DD algorithm was developed in [SM]. Specifically, the speech problem results in a pulse process that is intermittent (present for voiced speech only), and, when present, is of a highly structured nature (the pitch process exhibits a nearly periodic structure). An algorithm to simultaneously estimate the vocal tract parameters and to detect and estimate the pitch pulse waveform as well is presented here.

2.2 DECISION-DIRECTED DETECTOR

Consider a discrete-time point process (DTPP) $\{n_t\}$ for $t = 1,2,3,\ldots$, such that

$$Pr(n_t = 1 \mid \mathbf{B}_{t-1}) = 1 - Pr(n_t = 0 \mid \mathbf{B}_{t-1}) = \lambda_t \tag{2.2}$$

where λ_t is the random rate of the process $\{n_t\}$ and $\mathbf{B}_{t-1}$ is the sigma field generated by all the factors that affect the probability of a pulse occurring at time $t-1$. To simplify the development, assume that the effect of the jump is restricted to isolated time points, $b_i = 0$ for $i > 0$. The prediction error process is ϵ_t.

$$\epsilon_t = y_t - \hat{\mathbf{A}}_t^T \mathbf{Y}_t \quad \text{where} \quad \mathbf{Y}_t = [-y_{t-1}, \ldots, -y_{t-k}]^T \quad \mathbf{A} = [a_1, \ldots, a_k]^T \tag{2.3}$$

The symbol ˆ denotes the least squares estimate of the vector **A**. The detection problem is to decide between the two hypotheses H_0, noise only and H_1, pulse plus noise.

$$\begin{aligned} H_0&: \epsilon_t = v_t \\ H_1&: \epsilon_t = b + v_t \end{aligned} \tag{2.4}$$

The Bayes decision rule with respect to λ_t is N_t.

$$N_t = \begin{cases} 1, & \text{if } (1-\lambda_t) f(\epsilon_t \mid n_t{=}0) < \lambda_t f(\epsilon_t \mid n_t{=}1) \\ 0, & \text{otherwise} \end{cases} \tag{2.5}$$

where $f(\cdot \mid n_t{=}0)$ and $f(\cdot \mid n_t{=}1)$ are the density functions of ϵ_t under hypotheses H_0 and H_1, respectively. The output of the detector is the sequence $\{N_t\}$. Let λ_t' denote the rate of N_t. The philosophy of the DD approach is to estimate λ_t', and to use this estimate for subsequent operation. For the case where $v_t \sim N(0,1)$, the likelihood ratio test (LRT) of (2.5) assumes the form

$$N_t = \begin{cases} 1, & \epsilon_t > T(\hat{\lambda}_t') \\ 0, & \epsilon_t < T(\hat{\lambda}_t') \end{cases} \tag{2.6}$$

where $T(\lambda) = b/2 - [\log\lambda - \log(1-\lambda)] / b$ and $\hat{\lambda}_t'$ is an estimate of λ_t'.

Suppose that the rate of N_t can be modeled as a finite-state Markov chain with state vector $\rho = [\rho_1, \ldots, \rho_m]^T$, where $\rho_1 <, \ldots, < \rho_m$, with transition probabilities given by

$$Pr(\lambda_t' = \rho_j \mid \lambda_{t-1}' = \rho_i) = q_{ij}(t) \tag{2.7}$$

with initial distributior $\cdot = [\pi_\cdot \quad ., \pi_m]^T$ where $\pi_i = Pr(\lambda_0 = \rho_i)$.

Define $\mathbf{x}_t = [x_1(t),...,x_m(t)]^T$ by

$$x_i(t) = \begin{cases} 1, \text{ if } \lambda_t' = \rho_i \\ 0, \text{ otherwise} \end{cases} , \quad i = 1,2,\ldots,m \tag{2.8}$$

Thus, $\lambda_t' = \rho^T \mathbf{x}_t$. This formulation was first introduced by Segall [Se]. The vector $\mathbf{x}_t$ can be viewed as the state vector of a system obeying dynamics and observation equations of the form

$$\begin{aligned} \mathbf{x}_{t+1} &= \mathbf{Q}_t^T \mathbf{x}_t + \mathbf{u}_t \\ N_t &= \rho^T \mathbf{x}_t + \epsilon_t \end{aligned} \tag{2.9}$$

where $\mathbf{Q}_t = \{q_{ij}(t)\}$. The processes $\{\mathbf{u}_t\}$ and $\{\epsilon_t\}$ are Martingale Difference sequences with respect to the family of sigma fields $\{\mathbf{B}_t\}$ with $\mathbf{B}_t = \sigma\{N_1, \ldots, N_t, \mathbf{x}_1, \ldots, \mathbf{x}_{t+1}\}$.

A general estimator for this problem was developed in [St] for the case where the transition matrix, Q_t, is not only time dependent, but is realization-dependent as well. Suppose the transition matrix is conditioned on $\mathbf{F}_t$, and admits the structure in (2.10).

$$q_{ij}(t) = \begin{cases} s_{ij}(t, \mathbf{F}_{t-1}), & \text{if } N_t = 1 \\ r_{ij}(t, \mathbf{F}_{t-1}), & \text{if } N_t = 0 \end{cases} \tag{2.10}$$

The matrices $S_t = \{s_{ij}(t, \mathbf{F}_{t-1})\}$ and $R_t = \{r_{ij}(t, \mathbf{F}_{t-1})\}$ thus define the dynamics of the Markov chain. Note that these matrices are conditioned on the past data, represented by $\mathbf{F}_{t-1}$. The estimator takes the form in (2.11).

$$\begin{aligned} \hat{\mathbf{x}}_{t+1|t} &= \Delta_t \hat{\mathbf{x}}_{t|t-1} + \frac{S_t^T diag(\rho)\hat{\mathbf{x}}_{t|t-1} - \Delta_t \Sigma_t \rho}{\rho^T \mathbf{x}_{t|t-1} - (\rho^T \mathbf{x}_{t|t-1})^2} \nu_t \\ \Delta_t &= \mathbf{R}_t^T - (\mathbf{R}_t - \mathbf{S}_t)^T diag(\rho) \\ \Sigma_t &= \hat{\mathbf{x}}_{t+1|t} \, \hat{\mathbf{x}}_{t+1|t}^T \end{aligned} \tag{2.11}$$

The estimated rate is given by

$$\hat{\lambda}_{t+1|t}' = \rho^T \hat{\mathbf{x}}_{t+1|t} \,. \tag{2.12}$$

2.3 APPLICATION TO PITCH DETECTION

Consider the waveform $\{\epsilon_t\}$ defined by (2.3), and suppose that $v_t \sim N(0,1)$, b is a constant, and $\{n_t\}$ is a DTPP that is pseudo-periodic, in the sense that, once a pulse occurs, the probability of another pulse occurring soon is small, but increases as time progresses (an example of such a process is the sequence of glottal pulses of voiced speech). Also, suppose that the repetition interval (or pseudo-period) of this process may also be changing (e.g., as the pitch period is modulated, as with a singing voice). The near periodicity of the signal may, however be directly incorporated into the structure of the $\mathbf{Q}_t$ matrix as introduced above. Define the elements of $\mathbf{Q}_t$ as follows:

$$q_{ij}(t) = \begin{cases} I_{(J_t,\alpha_t)} + [1 - I_{(J_t,\alpha_t)}]\hat{z}_1(J_t+1 \mid J_t), & i = 1,\ j = 1 \\ I_{(J_t,\alpha_t)}, & i=1, j>1 \\ [1-I_{(J_t,\alpha_t)}]\hat{z}_j(J_t+1 \mid J_t), & i>1,\ j=1 \\ [1-I_{(J_t,\alpha_t)}](1-\psi), & i=j>1 \\ [1-I_{(J_t,\alpha_t)}]\psi, & i=3, j=2 \text{ and} \\ & i=m-1, j=m \\ [1-I_{(J_t,\alpha_t)}]\dfrac{\psi}{2}, & i=j+1, j>2 \text{ and} \\ & i=j-1, 2 \leq j < m \end{cases} \qquad (2.13)$$

where $\psi \in (0,1)$ is a constant;

$$I_{(J_t,\alpha_t)} = \begin{cases} 1, & \text{if } J_t < t < \alpha_t \\ 0, & \text{otherwise} \end{cases} \qquad (2.14)$$

is the indicator function;

$$J_t = \max\left\{ j : \sum_{s=1}^{j} N_s = \sum_{s=1}^{j-1} N_s + 1,\ j \leq t \right\} \qquad (2.15)$$

is the last time (up to and including t) that a pulse was detected; and

$$\alpha_t = J_t + \frac{1}{\lambda'_{J_t+1|J_t} + \sigma_{J_t}} \qquad (2.16)$$

where the conditional expectation of the estimation error is

$$\sigma_t^2 = E^{F}{}'(\tilde{\lambda}'_{t+1|t}) = \rho^T diag(\rho)\ \hat{\mathbf{x}}_{t+1|t} - (\hat{\lambda}'_{t+1|t})^2 \qquad (2.17)$$

In addition to the estimation of the model parameters and the detection of the pitch epochs, the speech analysis problem requires the estimation of the variance of the input noise process v_t, and the tracking of slowly-varying model parameters. The pitch pulse is usually not of a single sample time duration, but may persist for several sample times. Thus, the combined estimation/detection estimator must be generalized to allow input noise variance estimation and the estimation of composite pulses (i.e., pulses that persist for several time samples). The generalized algorithm is (2.18) where $\mathbf{Y}_t$ and $\mathbf{A}$ are defined as in (2.3).

$$\hat{\mathbf{A}}_t = \hat{\mathbf{A}}_{t-1} + \mathbf{P}_t\mathbf{Y}_t\,[y_t - \hat{\mathbf{A}}_{t-1}^T\mathbf{Y}_t - \sum_{i=1}^{k}\hat{b}_i'(t-1)N_{t-i}] \tag{2.18}$$

The matrix P_t is given by

$$\mathbf{P}_t = \frac{1}{\alpha_1}\left[\mathbf{P}_{t-1} - \frac{\mathbf{P}_{t-1}\mathbf{Y}_t\,\mathbf{Y}_t^T\mathbf{P}_{t-1}}{\alpha_1 + \mathbf{Y}_t^T\mathbf{P}_{t-1}\mathbf{Y}_t}\right]. \tag{2.19}$$

The unnormalized intensity estimate of the composite pulse profile is

$$\hat{b}_i'(t) = (\hat{S}_t)\ \hat{b}_i(t) \tag{2.20}$$

$$\hat{S}_t = \hat{S}_{t-1} + \frac{1}{\sum_{s=1}^{t}\alpha_2^{t-s}}\left[(\epsilon_t - \sum_{i=1}^{k}\hat{b}_i'(t-1)N_{t-i})^2 - \hat{S}_{t-1}\right] \tag{2.21}$$

$$\hat{b}_i(t) = \hat{b}_i(t-1) + \frac{N_{t-1}}{\sum_{s=1}^{t-i}\alpha_3^{t-s-i}N_s}[\epsilon_t - \hat{b}_i(t-1)] \tag{2.22}$$

The parameters α_1, α_2, and α_3 are the weighting factors for the model coefficients, the energy in the deconvolved waveform, and the pulse intensity, respectively. The process N_t is given by (2.5). Fig. 2.1 illustrates the block diagram of the joint estimation and excitation detection system.

The operation of the system with these transition dynamics is essentially as follows. Once a pulse is detected, the Markov chain is forced into its lowest state, ρ_1; thus raising the threshold and reducing the probability of false alarms over the interval immediately following the detection of a pulse. Once the time interval $\alpha_t - J_t$ has elapsed, the Markov chain is restored to its state at the time that the last pulse was detected. Fig. 2.2a illustrates the residual from the AR approximation and the adaptive threshold used to detect the pulses. Note that the residuals are clearly non-gaussian. Fig. 2.2b shows the residuals after the detected pulses have been removed. Fig. 2.2c shows the estimated pulse rate as defined by (2.12). The dash line indicated the pulse rate uncertainty as given by (2.17). The advantages of this procedure are: 1) the near-periodic nature of the process may be explicitly modeled; and 2) the "period" of the pulse train is adjusted adaptively, and the probability of false alarms decreases as σ_t decreases.

The ability of this algorithm to track pitch period variation and detect unvoiced speech is illustrated in Fig. 2.3. Fig. 2.3a is the beginning of the phrase " Thieves who rob .. " . The estimated pitch rate is shown in Fig. 2.3b; note the transition in pitch period and detection of unvoiced regions. In this example the pitch pulse was assumed to consist of three successive time samples. The estimated weighting coefficients are illustrated in Fig. 2.4.

2.4 CONCLUSIONS

A new approach to the pitch detection problem of speech analysis has been presented. This solution provides a mechanism to account for the structure of the pitch period, and thereby allows a reduction in pitch detection errors (false alarm rate). The key feature of this procedure is a new decision-directed algorithm that incorporates a finite-state Markov chain model for the rate of the process, and provides an exact, recursive nonlinear estimator for the rate. The algorithm allows the estimation of time-varying model parameters and the variance of the input WGN process.

The algorithm has been applied to samples of actual speech, and promising results have been obtained. It should be emphasized that much more work must be performed in order to validate this algorithm in actual speech analysis, but these preliminary results appear encouraging.

A further description of this decision-directed method of estimating a joint noise and jump process was presented in the first ONR project report, M736-1, Feb. 1983.

2.5 REFERENCES

[DS] Davisson, L.D. and Schwartz, S.C., "Analysis of a Decision-Directed Receiver With Unknown Priors," *IEEE Trans. Inform. Theory,* vol. IT-16, no. 3, pp. 270-276, May 1970.

[KD] Kazakos, D. and Davisson, L.D., "An Improved Decision-Directed Detector", *IEEE Trans. Inform. Theory,* vol IT-26, no. 1, pp. 113-116, Jan. 1980.

[Se] Segall, A., "Recursive Estimation from Discrete-Time Point Processes," *IEEE Trans. Inform. Theory,* vol. IT-22, no. 4, pp. 422-430, July 1976.

[SM] Stirling, W.C. and Morf, M., "A New Decision-Directed Algorithm for Nonstationary Priors," Proceedings 21-st IEEE CDC Conf., Orlando, Fl. Dec. 1982.

[St] Stirling, W.C., "Simultaneous Jump Excitation Modeling and System Parameter Estimation," PhD Dissertation, Stanford University, 1982.

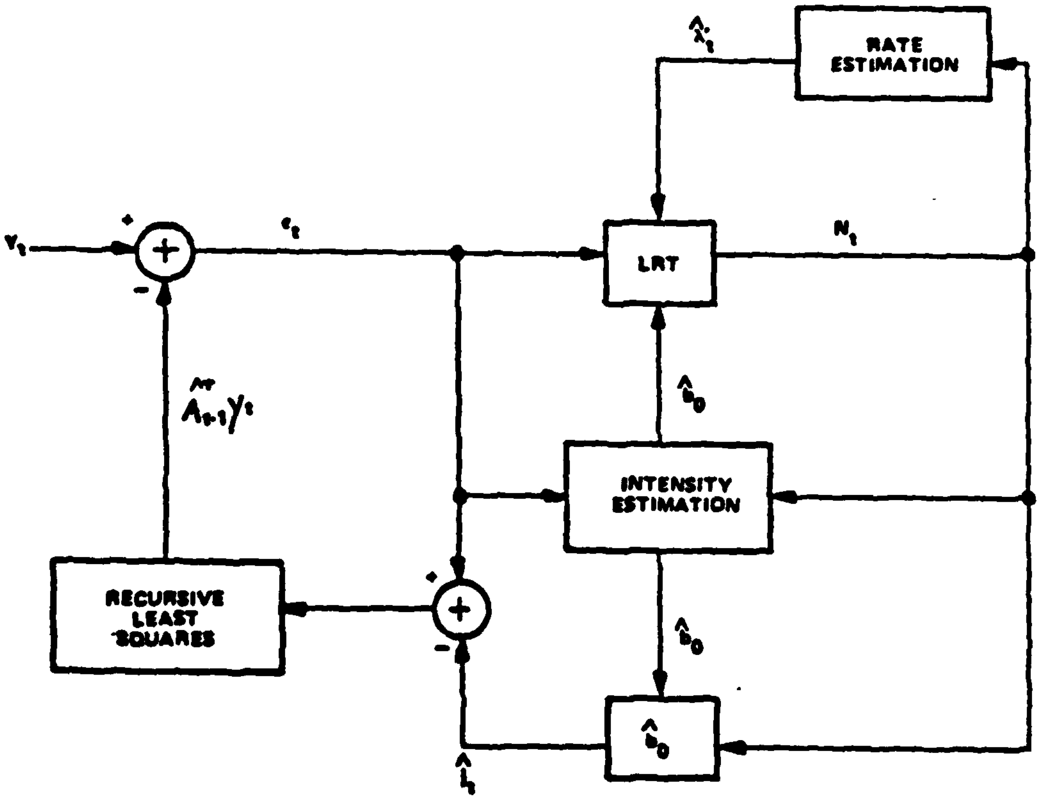


Figure 2.1 System Model Parameter Estimation/Input Pulse Detection Algorithm for Simple Pulses

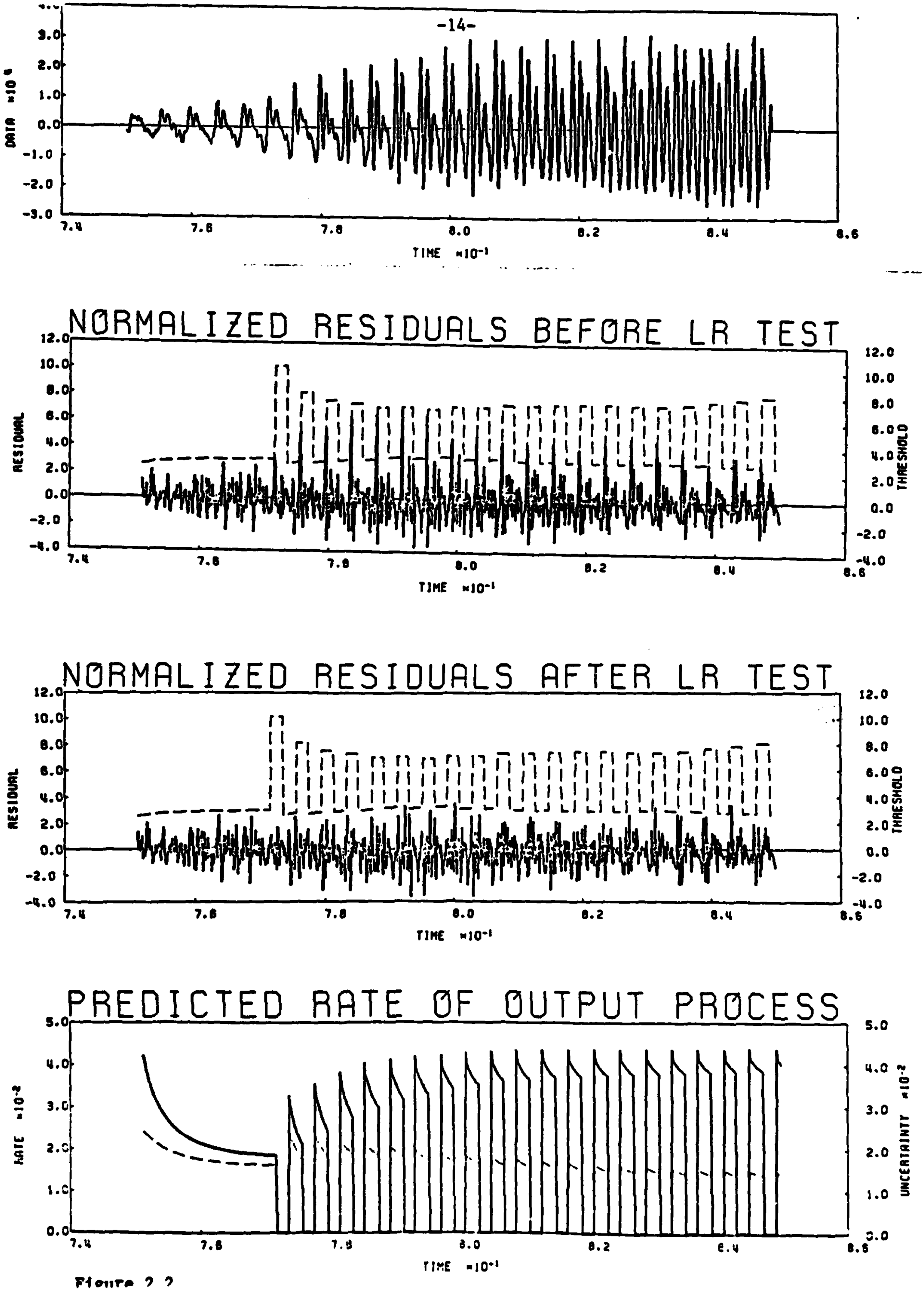

DATA ×10⁸
3.0
2.0
1.0
0.0
-1.0
-2.0
-3.0
7.4
7.6
7.8
8.0
8.2
8.4
8.6
TIME ×10⁻¹
NORMALIZED RESIDUALS BEFORE LR TEST
RESIDUAL
THRESHOLD
12.0
10.0
8.0
6.0
4.0
2.0
0.0
-2.0
-4.0
TIME ×10⁻¹
NORMALIZED RESIDUALS AFTER LR TEST
RESIDUAL
THRESHOLD
TIME ×10⁻¹
PREDICTED RATE OF OUTPUT PROCESS
RATE ×10⁻²
UNCERTAINTY ×10⁻²
5.0
4.0
3.0
2.0
1.0
0.0
TIME ×10⁻¹

Figure 2.2

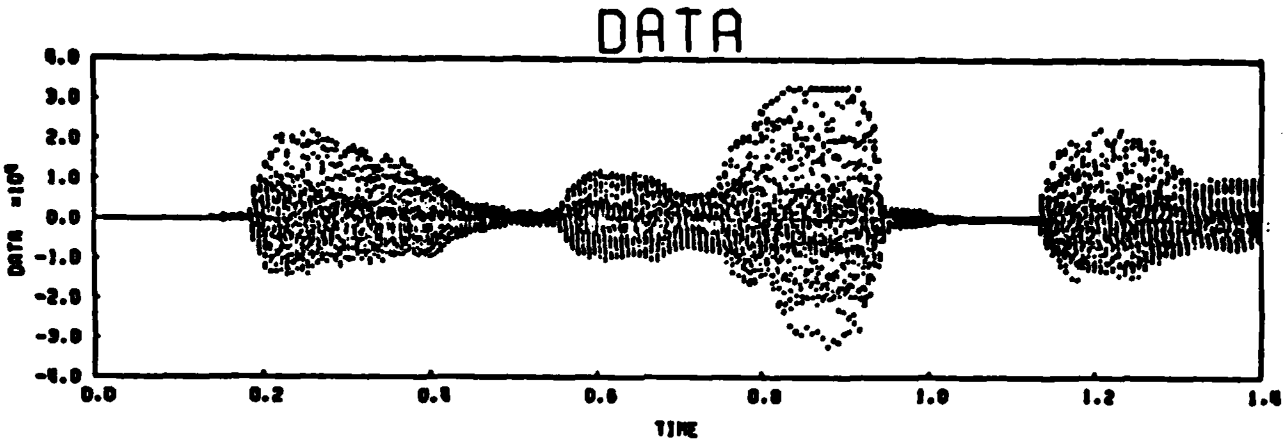


Figure 2.3a The Words: "Thieves Who Rob . . ."

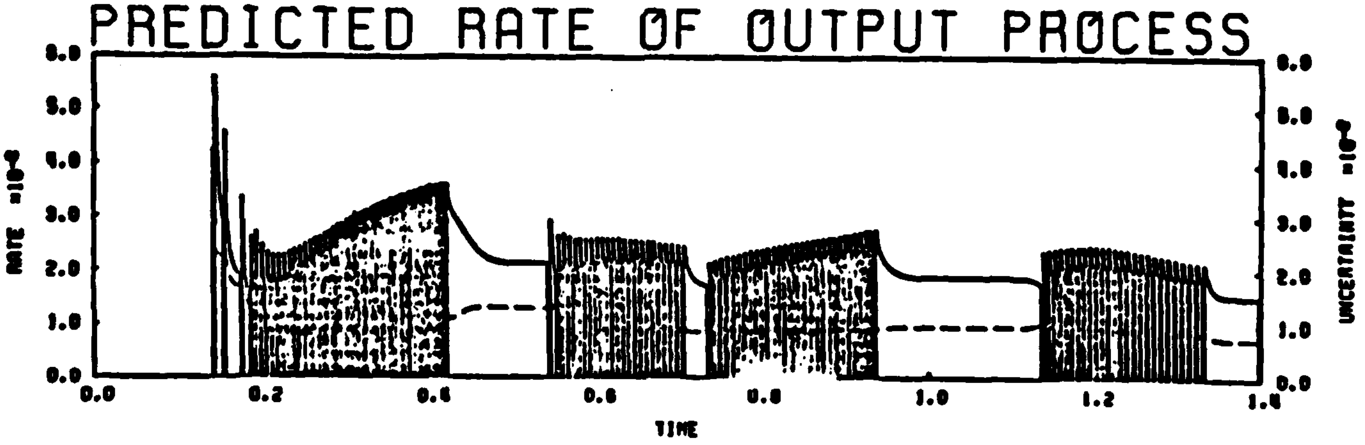


Figure 2.3b Estimated LRT Pulse Rate $\lambda_{t+1|t}$ For "Thieves Who Rob . . ."

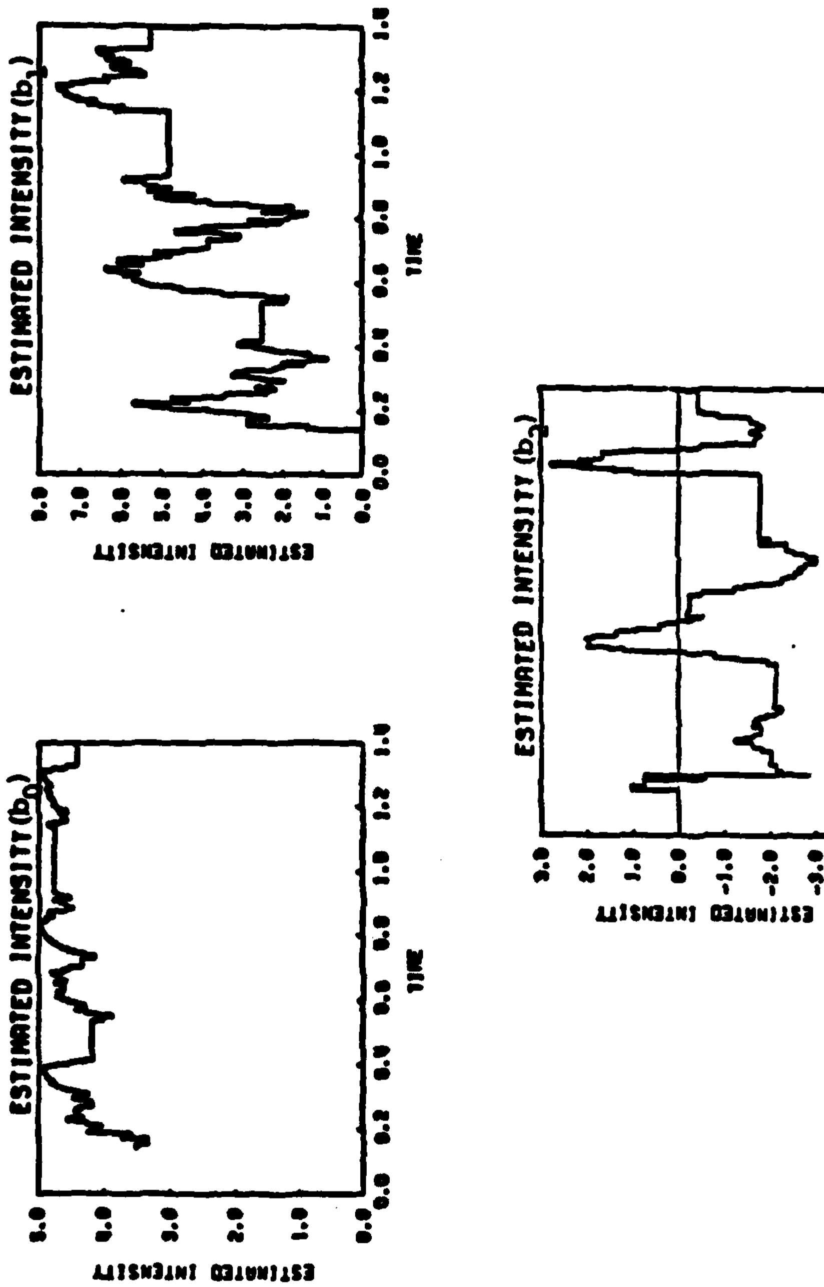


Figure 2.4 Estimated Intensity Profile For "Thieves Who Rob . . ."

3. PITCH DETECTION BY LEAST SQUARES LATTICE

3.1 INTRODUCTION

A new method of pitch detection for speech has been developed that is based upon the unnormalized pre-windowed least squares lattice algorithm. It is an extension of a previously studied method [LM] that involved the forward residuals and the so-called likelihood variable. By incorporating information from the forward residual covariance, well defined pitch pulse locations are produced from which the period can easily be determined.

A well known pitch detection method (LPC-10) [NSA] using the average magnitude difference function is discussed in Section 3.2. The unnormalized pre-windowed least squares lattice algorithm, which is fundamental to our approach, is summarized in Section 3.3. The new method of pitch detection and the pitch variable is presented in Section 3.4. Simulation results using sampled speech and comparisons are made with LPC-10 are in Section 3.5.

An efficient speech representation that captures the basic patterns in speech is essential for speech transmission at low bit rates or for speech recognition. The most popular parametric speech model consists of a linear filter with time varying coefficients driven by a time varying excitation process. The Linear Predictive Coding (LPC) [MG], [RS] model has an all pole filter with regularly updated coefficients excited by either white noise or a periodic pulse sequence. The filter represents the time varying nature of the vocal tract. The filter parameters determine the spectral characteristics of the resulting sound for both types of excitation. The periodic pulses generate voiced sounds such as vowels while unvoiced or hiss sounds are produced by the white noise process. Thus, the important parameters of such a speech model are: (1) filter coefficients, (2) voiced or unvoiced decision, (3) period of the pitch pulses (if voiced), and (4) signal energy. Based on the above parametric speech model, Fig. 3.1 displays the corresponding speech transmission system. The analysis component of the system determines the speech parameters which are then encoded for transmission across the channel. At the receiver, a synthesis filter characterized by the received coefficients is driven by the appropriate excitation process to generate a waveform which hopefully sounds like the original speech.

The temporal information carried by the periodic pulses or the change from noise to pulses is perceptually very important. The effect of errors in the excitation cause severe distortion in the synthesized speech. Errors in estimating the filter coefficients cause changes in the spectrum of the sound which tends to muffle the speech sound. Several techniques have been developed to estimate the filter coefficients. Unfortunately, the periodic excitation component is the most difficult to estimate. Our research activities in this area have been directed at better determination of the occurrence of pitch pulses.

3.2 STANDARD PITCH ESTIMATION TECHNIQUE

The pitch detection procedure used in LPC-10, the National Security Agency standard for 2400 bit per second speech transmission, was used as a benchmark for pitch period estimates, see Fig. 3.2. The transmitter is comprised of the necessary components required to determine the parameters of the above speech model. Note that the reflection coefficients (RC), energy (RMS), voiced/unvoiced (VUV) decision, and pitch for a segment of speech are encoded for transmission. A speech segment is typically 180 samples (8000 Hz sampling rate).

The pitch information is obtained by a series of operations on the speech waveform as indicated by Fig. 3.3. First, the speech is filtered by a low pass Butterworth filter (800 Hz bandwidth). This output is then whitened by a low order adaptive inverse filter to remove the speech formants. The average magnitude difference function (AMDF) of the resulting waveform is then computed as in (3.1) where x_k is the low pass and inverse filtered speech and L is the length of the speech segment [RSCFM].

$$F_n = \frac{1}{L}\sum_{k=0}^{L-1} |x_k - x_{k-n}| \ , \qquad n = -(L-1), \ldots 0 \ldots , (L-1) \tag{3.1}$$

Deep nulls occur in F_n at delays corresponding to the pitch period of a voiced sound having a quasi-periodic structure. From this information, a pitch decision algorithm involving dynamic programming determines the pitch period for the speech segment. The voiced/unvoiced decision is made from a zero crossing analysis of the speech and the energy of the low pass filtered speech.

3.3 PRE-WINDOWED LEAST SQUARES LATTICE

Since the pitch detection scheme utilizes parameters from the Least Squares Lattice estimation algorithm, this algorithm will be briefly introduced here. The "unnormalized" pre-windowed least squares lattice algorithm [Lee] was first derived from the well known multi-channel Levinson (LWR) algorithm for stationary processes. A more complete description of recursive least squares estimation is presented in [T]. The LWR solution involves solving the so-called normal equations, (3.2) recursively for the forward and backward predictor coefficients a_i and b_i.

$$\mathbf{R}_p \begin{bmatrix} 1 \\ a_1 \\ a_2 \\ \cdot \\ \cdot \\ a_p \end{bmatrix} = \begin{bmatrix} R_p^e \\ 0 \\ \cdot \\ \cdot \\ 0 \end{bmatrix} \qquad \mathbf{R}_p \begin{bmatrix} b_p \\ \cdot \\ \cdot \\ b_2 \\ b_1 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ \cdot \\ \cdot \\ 0 \\ R_p^r \end{bmatrix} \tag{3.2}$$

The (ensemble) covariance matrix of the process is $\mathbf{R}_p$ and R_p^e and R_p^r are the forward and backward prediction error (i.e. residual) covariances. The forward and backward residuals $e_{p,T}$ and $r_{p,T}$ are obtained from the predictor coefficients and the process y_T.

$$e_{p,T} = y_T + \sum_{i=1}^{p} a_i \, y_{T-i} \qquad r_{p,T} = y_{T-p} + \sum_{i=1}^{p} b_i \, y_{T-p+i} \tag{3.3}$$

In the derivation of the LWR algorithm, the mean square prediction error is minimized or equivalently the following orthogonality property is satisfied at each order-update recursion (E denotes expectation).

$$\mathrm{E}(e_{p,T} \, y_k) = 0 \ , \quad T-p \le k \le T-1 \tag{3.4}$$

When the desired filter order N is obtained, the recursions terminate resulting in only $O(N^2)$ computations compared to $O(N^3)$ required to simply invert $\mathbf{R}_p$.

It can be shown that the LWR algorithm leads naturally to a lattice filter structure that computes the forward and backward residuals. However, the reflection coefficients are *fixed* (time-independent) since the recursions are strictly an order-update solution for a stationary process with known second order statistics $\mathbf{R}_p$. As a consequence, the LWR lattice solution is incapable of tracking statistical variations.

Consequently, the pre-windowed lattice algorithm was developed to track *nonstationary* processes without any knowledge of the underlying statistics. Because the statistics are assumed unknown, the sum of squared prediction errors weighted by λ is minimized instead.

$$\sum_{k=0}^{T} \lambda^{T-k} e_{p,k}^2 \quad . \tag{3.5}$$

The exponential forgetting factor, λ $(0<\lambda\leq 1)$, permits more rapid tracking of statistical variations in the process. The resulting solution extends the LWR solution to include *time-update* recursions so that the reflection coefficients become time varying in general.

In order to introduce the time-update expressions, subscript T has to be appended to the coefficients to indicate that they are time-dependent. The forward and backward predictor coefficients become $a_{i,T}$ and $b_{i,T}$. The sample covariance of the process, $\mathbf{R}_{p,T}$ is defined as in (3.6).

$$\mathbf{R}_{p,T} = \mathbf{Y}_{p,T}^{T}\ \mathbf{Y}_{p,T} \qquad \text{where} \qquad \mathbf{Y}_{p,T} = \begin{bmatrix} y_0 & \cdot & \cdot & y_p & \cdot & \cdot & y_T \\ 0 & & & \cdot & & & \cdot \\ \cdot & & & \cdot & & & \cdot \\ \cdot & & & \cdot & & & \cdot \\ 0 & \cdot & 0 & y_0 & \cdot & \cdot & y_{T-p} \end{bmatrix} \tag{3.6}$$

An auxiliary set of coefficients is necessary to facilitate the time-update expressions. The particular quantity of interest is known as the *likelihood variable*, $\gamma_{p,T}$, and acts like an adaptive weighting factor involving previous data.

$$\gamma_{p,T} = [\, y_T\ , \ldots , y_{T-p}\,]\ \ \mathbf{R}_{p,T}^{-1}\ \ [\, y_T\ , \ldots , y_{T-p}\,]^T \tag{3.7}$$

The resulting algorithm is denoted pre-windowed since the data matrix (3.6) assumes that data prior to y_0 is exactly zero. Without going into the details of the derivation, we now discuss the algorithm and the corresponding lattice structure of Fig. 3.4.

The input/output expressions for the forward and backward residuals of each lattice section use $K_{p+1,T}^{e}$ and $K_{p+1,T}^{r}$, the forward and backward reflection coefficients.

$$\begin{aligned} e_{p+1,T} &= e_{p,T} - K_{p+1,T}^{r}\, r_{p,T-1} \\ r_{p+1,T} &= r_{p,T-1} - K_{p+1,T}^{e}\, e_{p,T} \end{aligned} \tag{3.8}$$

The lattice structure and (3.8) follow directly from the LWR solution except that in this case, the coefficients are time dependent (denoted by the subscript T). The order-update expressions for the forward and backward residual covariances also follow from the Levinson solution.

$$R^e_{p+1,T} = R^e_{p,T} - K^r_{p+1,T}\,\Delta_{p+1,T} \qquad (3.9)$$
$$R^r_{p+1,T} = R^r_{p,T} - K^e_{p+1,T}\,\Delta_{p+1,T}$$

The sample partial correlation coefficient (PARCOR) is the $\Delta_{p+1,T}$. When appropriately normalized, the partial correlation coefficient becomes the reflection coefficients which have the desirable numerical feature of being bounded by ± 1.

$$K^e_{p+1,T} = \Delta_{p+1,T}\,R^{-e}_{p,T} \qquad K_{p+1,T} = \Delta_{p+1,T}\,R^{-r}_{p,T-1} \qquad (3.10)$$

Here $R^{-e}_{p,T}$ and $R^{-r}_{p,T-1}$ are the matrix inverse of $R^e_{p,T}$ and $R^r_{p,T-1}$, respectively. The order-update expressions for the covariances (3.9) are employed initially when the time index is not greater than the desired filter length N.

The remaining recursions in the algorithm involve the likelihood variable and represent the major difference between the LWR solution and the adaptive lattice solution. The PARCOR variable can also be time-updated.

$$\Delta_{p+1,T} = \lambda\,\Delta_{p+1,T-1} + e_{p,T}\,r_{p,T-1} \,/\, (1 - \gamma_{p-1,T-1}) \qquad (3.11)$$

When the time index exceeds the filter order, the covariances are time-updated instead as in (3.12).

$$R^e_{p+1,T} = \lambda\,R^e_{p+1,T-1} + e^2_{p+1,T} \,/\, (1 - \gamma_{p,T-1}) \qquad (3.12)$$
$$R^r_{p+1,T} = \lambda\,R^r_{p+1,T-1} + r^2_{p+1,T} \,/\, (1 - \gamma_{p,T})$$

The likelihood variable is updated as in (3.13).

$$\gamma_{p,T} = \gamma_{p-1,T} + r^2_{p,T}\,R^{-r}_{p,T} \qquad (3.13)$$

It can be shown that the range of $\gamma_{p,T}$ is between zero and one.

The complete set of order and time-update recursions of the unnormalized pre-windowed adaptive lattice algorithm with exponential weighting are given by (3.8) to (3.13).

3.4 PITCH DETECTION BASED ON LEAST SQUARES LATTICE

The method of pitch prediction is an extension of previous results [LM] obtained with the unnormalized pre-windowed lattice algorithm of Section 3.3. This previously studied scheme utilized information contained in the forward residuals and the likelihood variable to determine pitch pulse locations in the speech waveform. The results were promising since well defined pitch pulses could be identified. However, in addition to these desired pulses, spurious less dominant ones were also present. Removing these from the waveform required a high degree of heuristic factors that resulted in limited success.

The new method of pitch detection enhances the previous results by employing the forward residual covariance. Consequently, more clearly defined pitch pulses can be obtained so that less heuristic factors are required to identify the desired pulse locations. The significance of the lattice variables used in the pitch estimation process; forward residuals, likelihood variable, forward residual covariance is discussed.

Forward Residuals

Consider a data sequence y_k where the time index k ranges from a finite time in the past (denoted zero) to the present time T. The p^{th} order forward residual $e_{p,T}$ is then defined as the difference between the actual value y_T and a linear least squares estimate $\hat{y}_{T|T-1,T-p}$ that involves only p previous data samples $(y_{T-p}, \ldots, y_{T-1})$.

$$e_{p,T} = y_T - \hat{y}_{T|T-1,T-p} \tag{3.14}$$

This estimate results from the projection of y_T on the space spanned by the p previous measurements. The coefficients for a linear predictor are $a_{p,T,k}$.

$$\hat{y}_{T|T-1,T-p} = -\sum_{k=1}^{p} a_{p,T,k}\, y_{T-i} \tag{3.15}$$

Now, $e_{p,T}$ represents the new information in y_T that is not present in the p previous measurements. As a result, it can provide information concerning waveform changes that may not be as obvious in the original process. It is precisely this feature of the residuals that is important for pitch detection. If one observes a voiced segment of speech, the quasi-periodic structure is easily seen. However, it is difficult to consistently identify waveform locations from which to reliably extract the pitch period. This occurs since there is a high degree of correlation between speech samples [RSCFM].

Since the residuals are a whitened form of the speech process, they provide more clearly defined events from which to identify the pitch period. There is much other information contained in the residuals, extraneous to pitch detection, that must be removed or masked. This function is provided by the likelihood variable and the forward residual covariance. Since $e_{p,T}$ is not truely a whitened process, ie. innovations and since as much uncorrelation as possible is required, only the highest order residual $e_{N,T}$ is considered. The true innovations involve all past data, $y_0 , \ldots , y_{T-1}$.

Likelihood Variable

The definition of the likelihood variable $\gamma_{p,T}$ from (3.7) in terms of the sample covariance $\mathbf{R}_{p,T}$ is (3.16).

$$\gamma_{p,T} = Y^T_{T:T-p} \, \mathbf{R}^{-1}_{p,T} \, Y_{T:T-p} \qquad \text{where} \qquad Y_{T:T-p} = [\, y_T \, , \ldots , y_{T-p} \,]^T \tag{3.16}$$

For a (zero mean) Gaussian process, the p^{th} order likelihood function is $p(Y_{T:T-p})$ where $\mathbf{R}_p$ is the ensemble covariance of the process.

$$p(Y_{T:T-p}) = (2\,\pi)^{-p/2} \mid \mathbf{R}_p \mid^{-1/2} \exp(\, -1/2 \; Y^T_{T:T-p} \, \mathbf{R}_p^{-1} \, Y_{T:T-p} \,) \tag{3.17}$$

Thus $\gamma_{p,T}$, called the likelihood variable is an estimate of the exponent of the likelihood function. Although not obvious from this result, it has been shown (by simulation) that $\gamma_{p,T}$ is a good indicator of deviations from a Gaussian distribution [LM, ML]. This is of course desirable for pitch detection since the speech model consists of a Gaussian component for unvoiced segments and a non-Gaussian quasi-periodic component for voiced segments. Thus, sudden changes in $\gamma_{p,T}$ should indicate the onset of voiced segments in speech. In fact, simulation results show that $\gamma_{p,T}$ does change significantly for voiced speech segments.

The likelihood variable detects general statistical deviations (see Section 3.5). Consequently other speech characteristics such as plosives, which are not quasi-periodic, are also detected by

$\gamma_{p,T}$.

Nevertheless, promising results have been obtained by multiplying together the forward residual signal and the derivative of $\gamma_{p,T}$. Simulations have shown that much of the extraneous information contained in the forward residuals is removed to expose well defined pulse locations from which to identify the pitch period. However, as mentioned before, spurious pulses generally remained which were then removed by a combination of thresholding and an exponentially decaying function that basically extracts the largest peaks over the waveform. These heuristic methods can be reduced by the enhanced pitch detection method. The new method which utilizes the forward residual covariance further reduces the occurrence of these spurious pulses *without thresholding*.

The role of the forward residuals is important since $\gamma_{p,T}$ corresponds precisely to their normalized sum (squared).

$$\gamma_{p,T} = \sum_{k=0}^{p} R^{-e}_{k,T+k-p} \; e^{2}_{k,T+k-p} \tag{3.18}$$

Thus $\gamma_{p,T}$ contains information from p measurements see (3.16). For the same reason that $e_{N,T}$ is used, only the highest order quantity $\gamma_{N,T}$ is used for pitch detection; simulations have shown that $\gamma_{N,T}$ produces better results (than lower orders) - namely well defined pitch pulse locations.

Forward Residual Covariance

Recall the time-update expression for the forward residual covariance is (3.19).

$$R^{e}_{p+1,T} = \lambda \, R^{e}_{p+1,T-1} + e^{2}_{p+1,T} \, / \, (1 - \gamma_{p,T-1}) \tag{3.19}$$

It is essentially the sum of the present and all previous (exponentially weighted) forward residuals squared. Since, the effect of the initial order-update recursion becomes negligible, especially with the exponential "forgetting" factor, λ, the effect of the initial order-update (3.9) can be ignored. Simulations indicate that the effect of $\gamma_{p,T-1}$ on $R^{e}_{p,T-1}$ is small and can be ignored.

$$R^{e}_{p+1,T} = \sum_{k=0}^{T} \lambda^{T-k} \, e^{2}_{p+1,k} \tag{3.20}$$

As a consequence of this lengthy memory, the covariance does not change significantly except for large (magnitude) increases in the residuals. This may occur, for example, when the variance of the underlying process increases as in the case of voiced segments of speech. Furthermore, the covariance does not change much for decreases in the residuals. The degree of change is affected directly by the value of λ, the memory factor in the algorithm.

This is a desirable feature, not shared by the likelihood variable, since the covariance can detect a specific event of the waveform. Thus it becomes possible to consistently track recurring large-magnitude increases in the speech waveform. By further masking (multiplying time signals together) the forward residuals with the derivative of the covariance, a single event in each period of the voiced segment can then be emphasized and therefore be more easily detected. In fact, simulation results show that employing the covariance does enhance significantly the pitch pulse locations. Consequently, the need for thresholding is reduced and windowing can be used instead of an exponentially decaying function.

Simulation results also show that the highest order covariance $R^e_{N,T}$ provides better results than lower orders; this appears to be related to the reduced correlation of the forward residuals $e_{N,T}$.

Method of Pitch Detection

The fundamental concepts underlying the new method of pitch detection have now been discussed. Those concepts can be combined into a single pitch detection variable. Recall that the likelihood variable detects changes in the process statistics. Consequently, it's derivative (i.e. first order time difference) indicates the intensity of those changes.

$$\delta\gamma_{N,T} = \gamma_{N,T} - \gamma_{N,T-1} \tag{3.21}$$

If the forward residuals are multiplied by (3.21), then statistical changes in the process can be emphasized. However, since (3.21) detects more events than that required for pitch detection, the residuals are multiplied by the derivative of the forward residual covariance.

$$\delta R^e_{N,T} = R^e_{N,T} - R^e_{N,T-1} \tag{3.22}$$

This will then emphasize only those statistical changes that also include an increase in variance. Thus the complete pitch detection variable, denoted $\eta_{N,T}$, is (3.23).

$$\eta_{N,T} = \mathrm{POS}\ (\ e_{N,T}\ \ \delta\gamma_{N,T}\ \ \delta R^{e}_{N,T}\ ) \tag{3.23}$$

where POS simply retains positive results of the quantity in parentheses.

Equation (3.22) clearly indicates a specific event in each period of a voiced speech waveform, see Section 3.5. For unvoiced speech, pitch pulses are not produced by (3.22) so that the need for separate voiced/unvoiced decision logic is eliminated. A summary of the (scalar case) pre-windowed lattice algorithm with (3.20) to (3.23) incorporated follows on the next page.

SUMMARY : PITCH DETECTION VARIABLE

UNNORMALIZED PRE-WINDOWED LATTICE ALGORITHM

```
Initialization:

    $R^e_{0,-1} = R^r_{0,-1} =$ a priori estimate        $N =$ filter order

For each observation $y_T$ , $T \geq 0$:

    $e_{0,T} = r_{0,T} = y_T$     $\gamma_{-1,T} = 0$     $R^e_{0,T} = R^r_{0,T} = \lambda R^e_{0,T-1} + y_T y_T$

    For $p = 0 , \ldots \min \{N,T\} - 1$:

        $\Delta_{p+1,T} = \lambda \Delta_{p+1,T-1} + e_{p,T} r_{p,T-1} / (1 - \gamma_{p-1,T-1})$

        $\gamma_{p,T} = \gamma_{p-1,T} + r^2_{p,T} / R^r_{p,T}$

        $K^r_{p+1,T} = \Delta_{p+1,T} / R^r_{p,T-1}$        $K^e_{p+1,T} = \Delta_{p+1,T} / R^e_{p,T}$

        $e_{p+1,T} = e_{p,T} - K^r_{p+1,T} r_{p,T-1}$     $r_{p+1,T} = r_{p,T-1} - K^e_{p+1,T} e_{p,T}$

      If $T \leq N$ then:

        $R^e_{p+1,T} = R^e_{p,T} - K^r_{p+1,T} \Delta_{p+1,T}$

        $R^r_{p+1,T} = R^r_{p,T-1} - K^e_{p+1,T} \Delta_{p+1,T}$

      Else:

        $R^e_{p+1,T} = \lambda R^e_{p+1,T-1} + e^2_{p+1,T} / (1 - \gamma_{p,T-1})$

        $R^r_{p+1,T} = \lambda R^r_{p+1,T-1} + r^2_{p+1,T} / (1 - \gamma_{p,T})$

    $\gamma_{N,T} = \gamma_{N-1,T} - r^2_{N,T} / R^r_{N,T}$

    $\delta\gamma_{N,T} = \gamma_{N,T} - \gamma_{N,T-1}$     $\delta R^e_{N,T} = R^e_{N,T} - R^e_{N,T-1}$

    $\eta_{N,T} = \mathrm{POS} ( \delta R^e_{N,T} \delta\gamma_{N,T} e_{N,T} )$
```

3.5 SPEECH DATA RESULTS

Some simulation results obtained with the new pitch detection approach using the variable $\eta_{N,T}$ are presented. The following phonetically balanced sentences, developed by the Advanced Research Projects Agency (DARPA), were studied.

File 1: 'cats and dogs each hate the other' ; male speaker

File 2: 'the pipe began to rust while new' ; female speaker

Both sentences were sampled at 8000 Hz. with 16-bit integer quantization. The analysis lattice employed in the simulations had the following parameter specifications; $R^e_{0,-1} = R^r_{0,-1} = 100{,}000$, $\lambda = .99$, and $N = 10$. This value of λ corresponds essentially to a window length of 100 samples which greatly exceeds the filter length N used.

File 1

The speech waveform of File 1 is shown in Fig. 3.5. The voiced segments are clearly visible as those areas of (relatively) large magnitude. The first 2000 samples of this sentence which corresponds to the word 'cats' is examined in detail. The consonant 'c' is visible beginning at about sample 300 while the onset of the vowel 'a' occurs near sample 700, see Fig. 3.6. The unvoiced letters 'ts' are not visible in this plot. The segment of interest for pitch detection is the vowel /a/ since it corresponds to a quasi-periodic voiced segment of speech. The goal is to extract the pitch information from this segment. The variables used in $\eta_{N,T}$ are shown separately then the full pitch estimate.

The ten reflection coefficients $K_{N,T}$ are shown in Fig. 3.7. This combined reflection coefficient $K_{N,T}$ corresponds to SIGN ($K^e_{N,T}\, K^r_{N,T}$) where SIGN simply applies the sign of $K^e_{N,T}$ (which is the same as $K^r_{N,T}$ - see (3.10)) to the product in parentheses. A sudden change occurs in all coefficients at the location of 'c' which is due to a change in the likelihood variable $\gamma_{N,T}$ (caused by a change in the process statistics), whose influence on $K_{N,T}$ is through (3.10)-(3.12). The periodic structure of the coefficient waveforms is caused precisely by the periodic nature of the voiced segment 'a'. In fact, this periodicity appears in all lattice variables, which is not surprising since they each contain some combination of the forward residuals.

The forward residual $e_{N,T}$ is shown in Fig. 3.8 and its covariance $R^{e}_{N,T}$ in Fig. 3.9. Both of these variables appear in the pitch detection variable $\eta_{N,T}$. We observe that $e_{N,T}$ does correspond to a partially "whitened" version of the speech waveform of Fig. 3.6. Certain events are emphasized more than others so that the periodic structure is well defined. It is this result that permits clearly defined pitch pulses to be exposed when $e_{N,T}$ is appropriately masked by $\delta\gamma_{N,T}$ and $\delta R^{e}_{N,T}$. From the covariance waveform, the periodic structure consists of very abrupt increases and exponentially decaying decreases (due to λ). In addition, 'c' produces very little change in the waveform which is desirable for pitch detection. Both of these results are of course due to variance changes occurring in the original speech waveform.

The likelihood variable $\gamma_{N,T}$ waveform is displayed in Fig. 3.10. It is seen that both 'c' and 'a' significantly affect $\gamma_{N,T}$ or, in other words, $\gamma_{N,T}$ detects (equally well) both types of statistical variations; the onset of the unvoiced plosive 'c' and the voiced vowel 'a'. This is desirable for pitch detection but, as mentioned previously in Section 4, there is in a sense more information than necessary.

Next, $\delta\gamma_{N,T}$ and $\delta R^{e}_{N,T}$ are presented in Figs. 3.11 and 3.12, respectively. As expected from Fig. 3.9, the dominant pulses of $\delta R^{e}_{N,T}$ are positive and little emphasis is placed on 'c'; such is not the case with $\delta\gamma_{N,T}$. However, when the forward residuals are masked by these quantities, well defined pitch pulse locations are obtained with relatively few spurious pulses as indicated in Figs. 3.13 and 3.14. Fig. 3.15 shows further improvement when $\delta\gamma_{N,T}$ masks $\delta R^{e}_{N,T}$, but even better results are obtained when both $\delta\gamma_{N,T}$ and $\delta R^{e}_{N,T}$ mask the forward residual $e_{N,T}$ as shown in Fig. 3.16.

A more detailed look at $\eta_{N,T}$ for samples 1000 - 1400 shows the quasi-periodic structure of 'a', Fig. 3.17. Note that the less dominant pulses in any period of $\eta_{N,T}$ (if they exist) tend to cluster about the desired dominant pitch pulse locations. Hence the need for thresholding is reduced since windowing can be used to extract a pitch pulse location centered near the cluster.

For comparison with pitch results obtained with LPC-10, Fig. 3.18 displays a portion of $\eta_{N,T}$ (samples 800 - 1800). The upper row of numbers corresponds to the pitch periods obtained with the new method and the lower row contains those determined by the LPC-10 algorithm; the dotted lines indicate the boundaries of the 180 sample frames for which the LPC-10 pitch periods were obtained. The new method using $\eta_{N,T}$ provides results comparable to those of the NSA standard.

The words 'and dogs' (samples 3000 - 5000), from the same sentence (File 1), shown in Fig. 3.19 were also analyzed. The onset of 'a' is visible at about sample 3200 with 'o' beginning at about 4600; the highly sinusoidal structure of the nasal 'n' ranges from 3500 to 4500 and the two consonants 'd' actually occur as one at about sample 4500 ('gs' is not visible in this plot). Here the $\eta_{N,T}$ does not produce (significant) pitch pulses for much of the highly sinusoidal structure of the nasal 'n', see Fig. 3.20. However, by examining more closely the range 3600 - 4400 and by changing scales, pitch pulse locations are indeed present, Fig. 3.21. Thus increased dynamic range results from the new pitch detection method which is a direct consequence of the product $e_{N,T}\ \delta\eta_{N,T}\ \delta R^{e}_{N,T}$. This result is in a sense a trade-off required to obtain such well-defined pitch pulses. Nevertheless this effect is not a problem since pitch pulse locations can be determined on a local basis by windowing (e.g. 50 - 200 samples) so that the range of $\eta_{N,T}$ over a window length is relatively small.

File 2

The pitch period of female speakers is typically less than that of male speakers so that it is generally more difficult to consistently determine. From the second sentence, the onset of the word 'while' is displayed in Fig. 3.22. The consonants 'wh' are barely noticeable so that the onset of the vowel 'i' occurs almost immediately at sample 1700. Good results are obtained with $\eta_{N,T}$, Fig. 3.23 and 3.24. The results concur with those of LPC-10, see Fig. 3.25.

3.6 CONCLUSIONS

As a consequence of the lattice filter algorithm, the information needed to compute the pitch period is available at each time instant. On-line pitch detection is therefore possible and moreover additional parallel processing is *not* needed to determine pitch pulse locations. That is, the recursions required to compute the reflection coefficients that characterize the parametric speech model also compute *simultaneously* the pitch variable. Furthermore, the voiced/unvoiced decision is inherent in the masking technique; either a pitch pulse is present (voiced) or it is not (unvoiced).

As an extension of previous results using the likelihood variable, the new method minimizes the need for thresholding since more distinct pitch pulses are generated. As a consequence of this, the exponentially decaying function used to determine the period can be replaced by a simpler windowing technique.

3.7 REFERENCES

[Lee] D.T.L. Lee, "Canonical Ladder Form Realizations and Fast Estimation Algorithms," Ph.D. Dissertation, Stanford University, Stanford, CA, Aug. 1980.

[LM] D.T.L. Lee and M. Morf, "A Novel Innovations Based Time-Domain Pitch Detector," *IEEE Int. Conf. ASSP*, pp. 40-44, Denver, CO, April 1980.

[MG] Markel, J. D. and A. H. Gray, Jr., *Linear Prediction of Speech*, Springer-Verlag, New York, 1976.

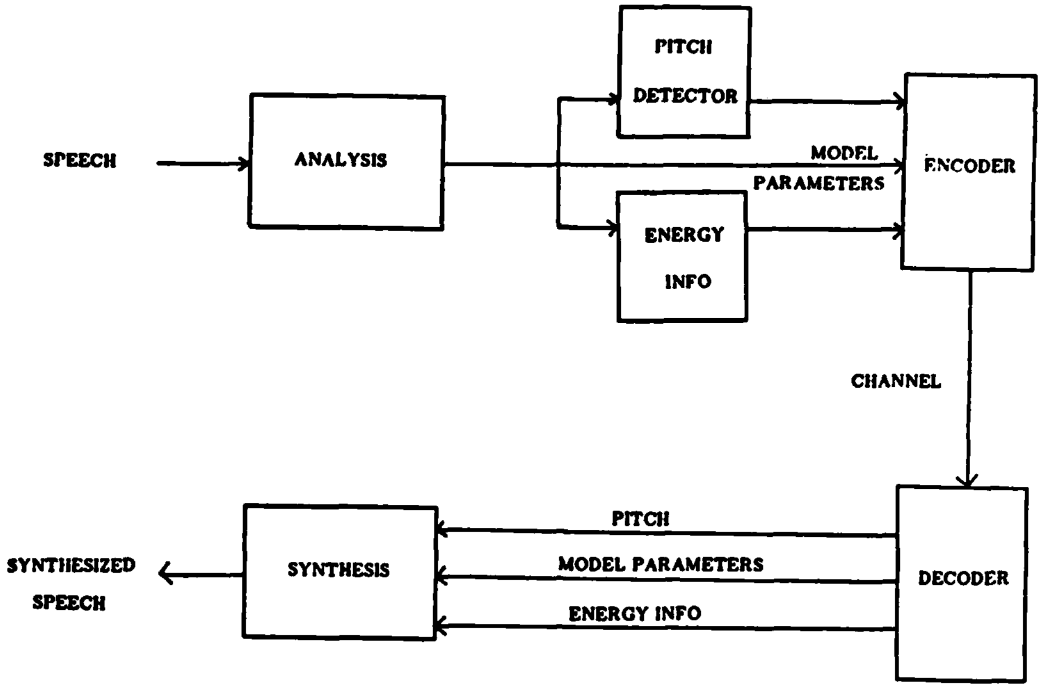
[ML] M. Morf and D.T.L. Lee, "Fast Algorithms for Speech Modeling," Technical Report M308-1, Information Systems Laboratory, Stanford University, Stanford, CA , Dec. 15, 1978.

[NSA] National Security Agency, Documentation and Program Sources for LPC-10, Version 43A, Feb. 1982.

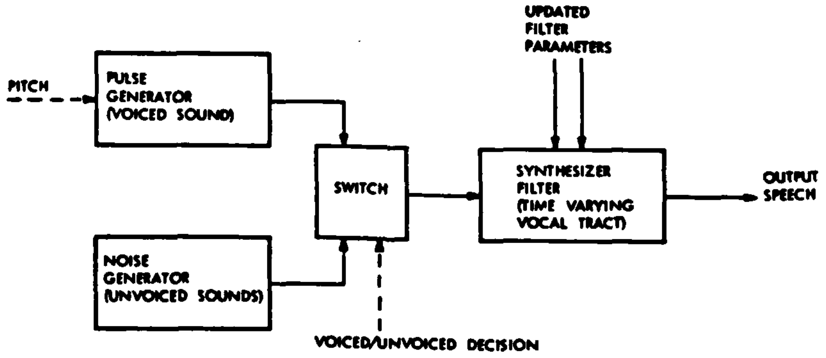
[RS] L.R. Rabiner and R.W. Schafer, *Digital Processing of Speech Signals*, Prentice-Hall, Englewood Cliffs, NJ, 1978.

[RSCFM] M.J. Ross, H.L. Shaffer, A. Cohen, R. Freudberg, and H.J. Manley, "Average Magnitude Difference Function Pitch Extractor," *IEEE Trans. ASSP*, pp. 353-362, Oct. 1974.

[T] Turner,J., "Recursive Least Squares Estimation and Lattice Filters", a chapter in *Adaptive Filters*, Cowan,N. and Grant,P. (editors) , Prentice Hall, to be published 1984.

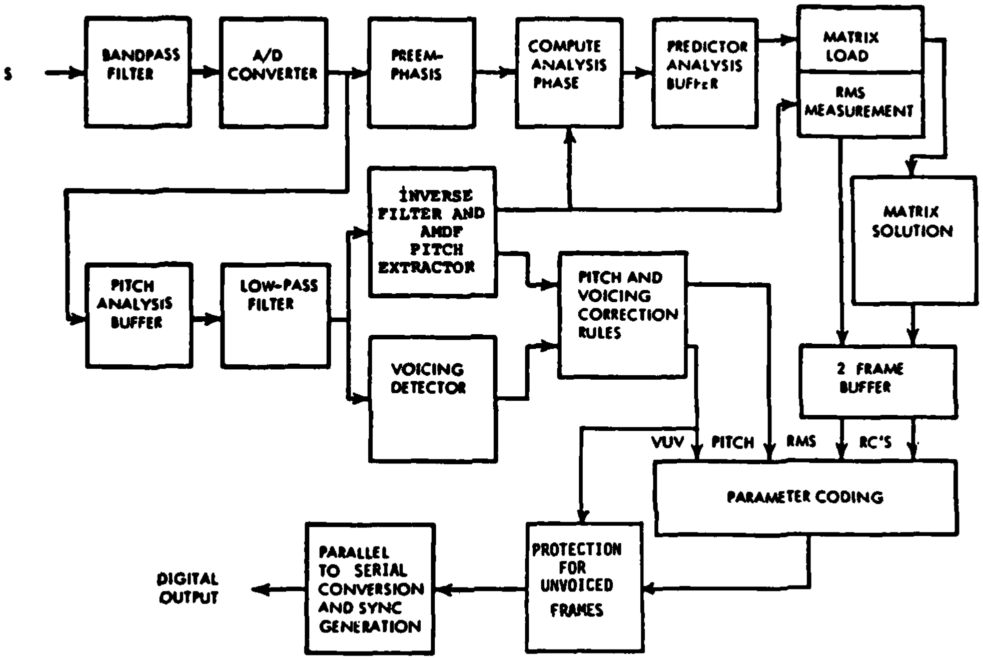


SPEECH TRANSMISSION SYSTEM

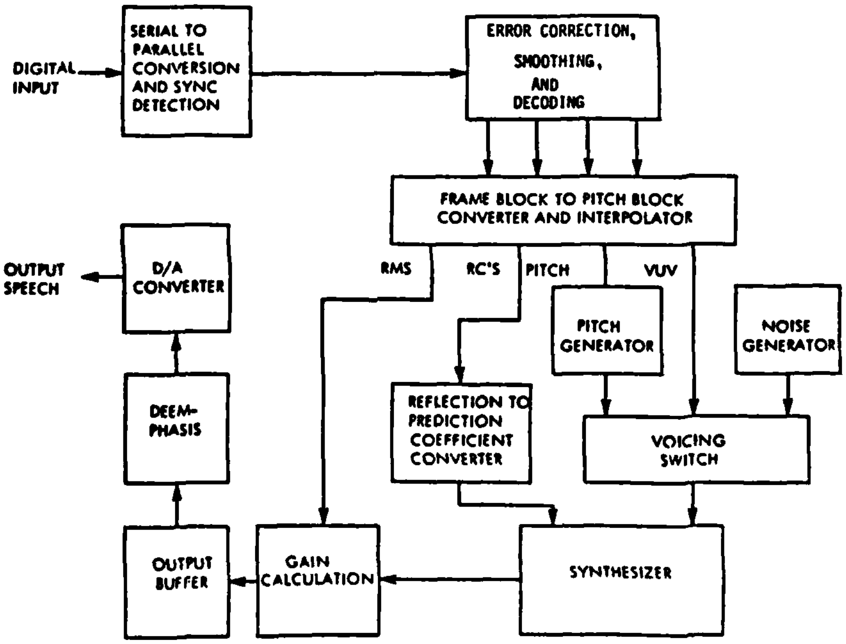

UPDATED FILTER PARAMETERS
PITCH
PULSE GENERATOR (VOICED SOUND)
SWITCH
SYNTHESIZER FILTER (TIME VARYING VOCAL TRACT)
OUTPUT SPEECH
NOISE GENERATOR (UNVOICED SOUNDS)
VOICED/UNVOICED DECISION

Model of the speech production mechanism.

Figure 3.1

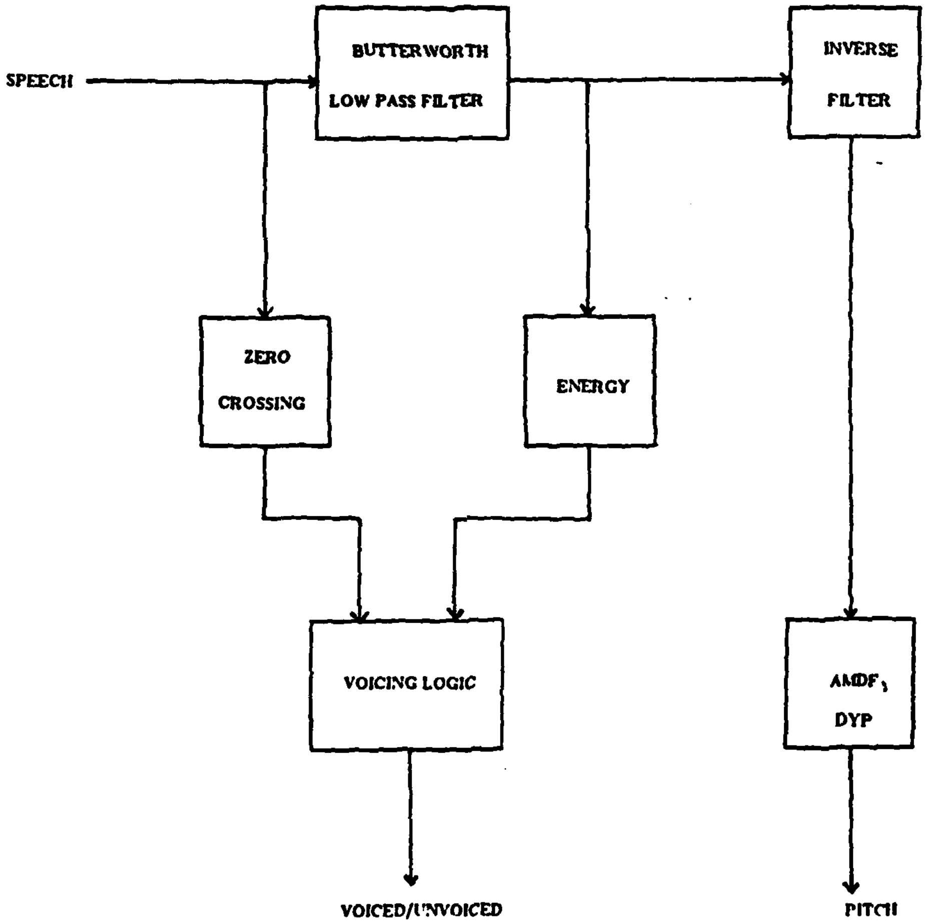


LPC-10 TRANSMITTER


DIGITAL INPUT
SERIAL TO PARALLEL CONVERSION AND SYNC DETECTION
ERROR CORRECTION, SMOOTHING, AND DECODING
FRAME BLOCK TO PITCH BLOCK CONVERTER AND INTERPOLATOR
RMS
RC'S
PITCH
VUV
OUTPUT SPEECH
D/A CONVERTER
PITCH GENERATOR
NOISE GENERATOR
DEEM-PHASIS
REFLECTION TO PREDICTION COEFFICIENT CONVERTER
VOICING SWITCH
OUTPUT BUFFER
GAIN CALCULATION
SYNTHESIZER

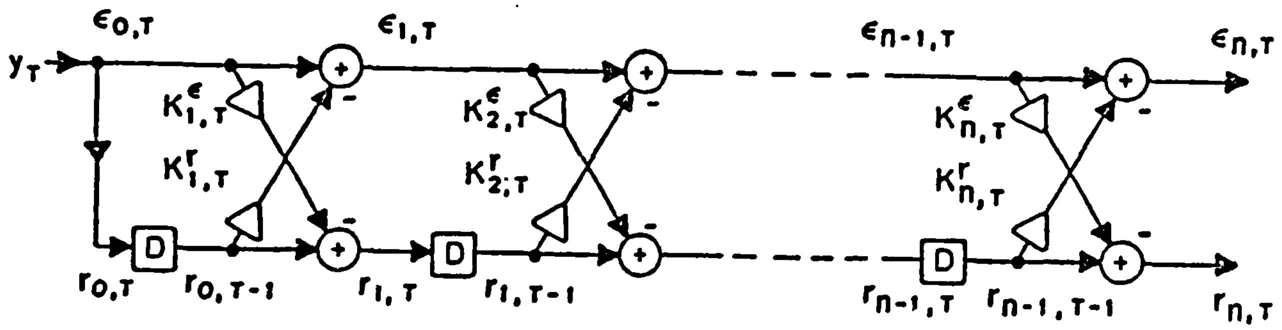
LPC-10 RECEIVER

Figure 3.2



LPC-10 PITCH DETECTION

Figure 3.3



UNNORMALIZED LATTICE FILTER

Figure 3.4

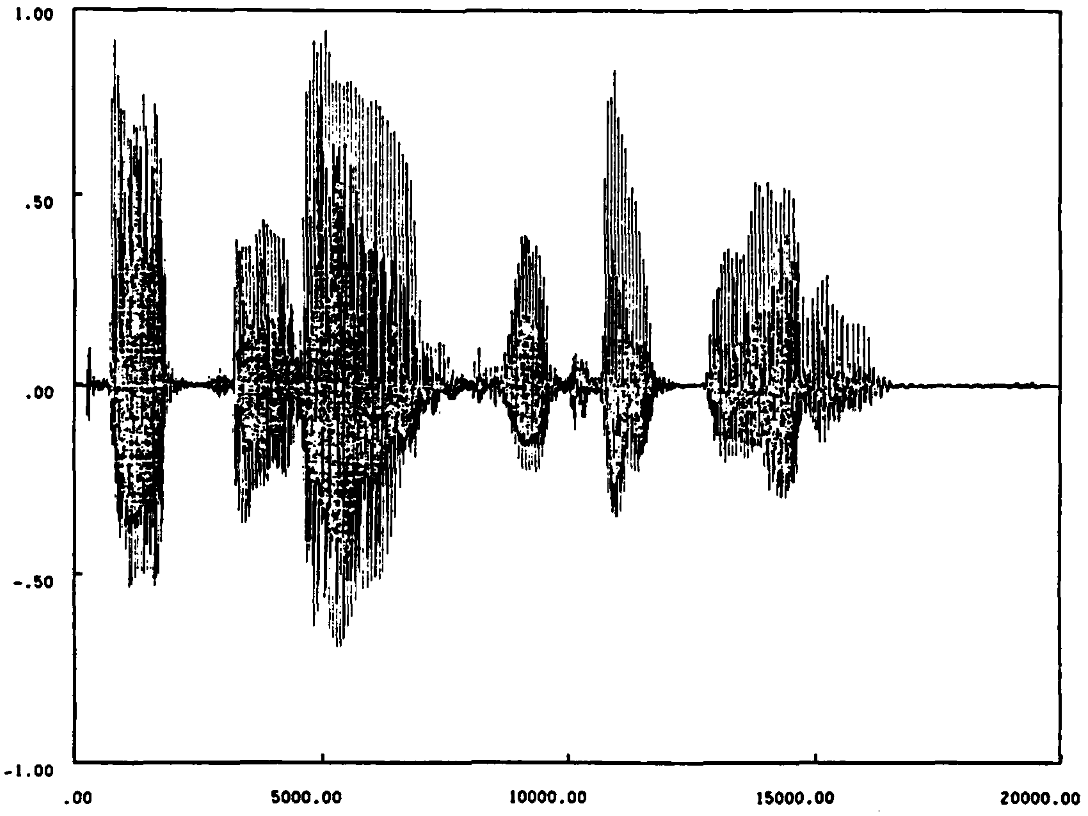


Figure 3.5 16-bit digitized speech- FILE 1

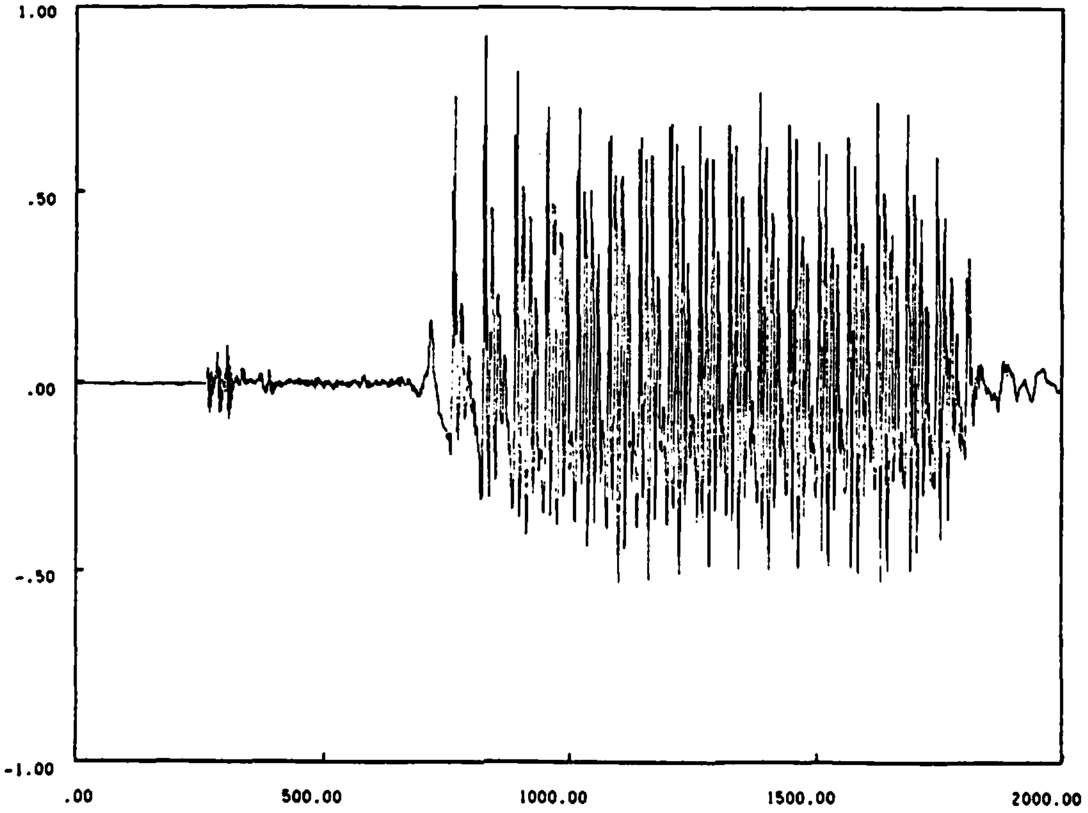


Figure 3.6 16-bit digitized speech- FILE 1

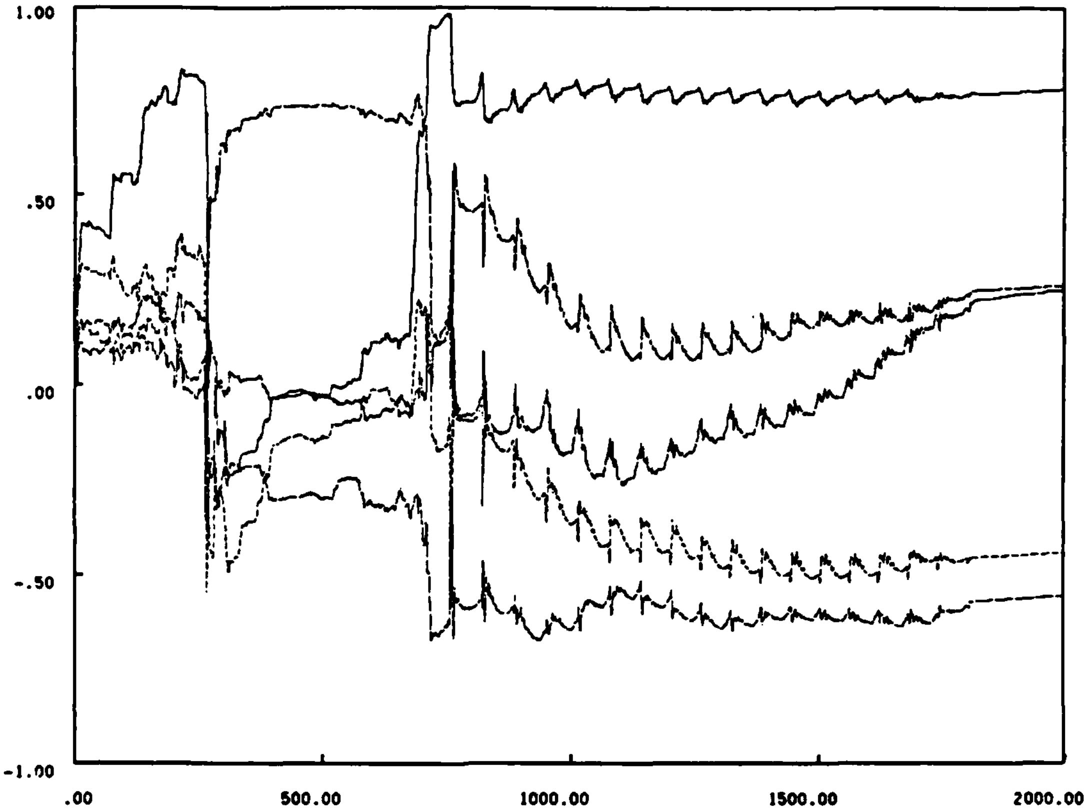
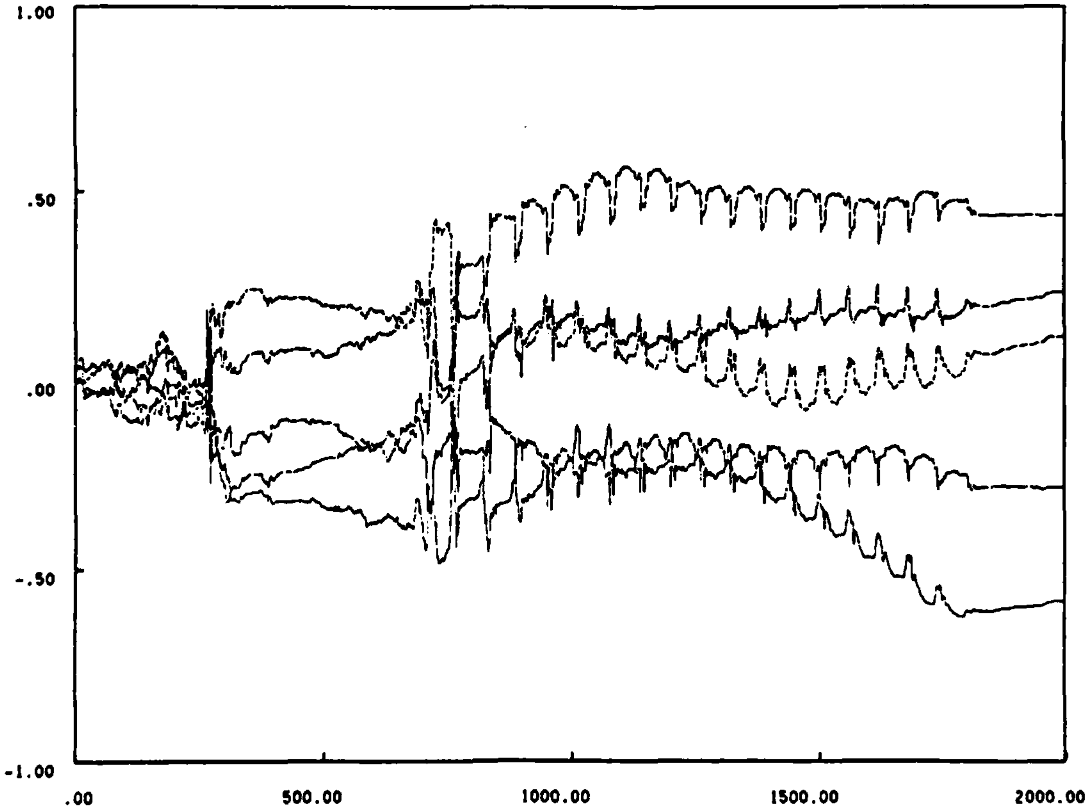


Figure 3.7 Reflection coefficients of speech data in FILE 1 ,orders 1- 5



Reflection coefficients of speech data in FILE 1 ,orders 6-10

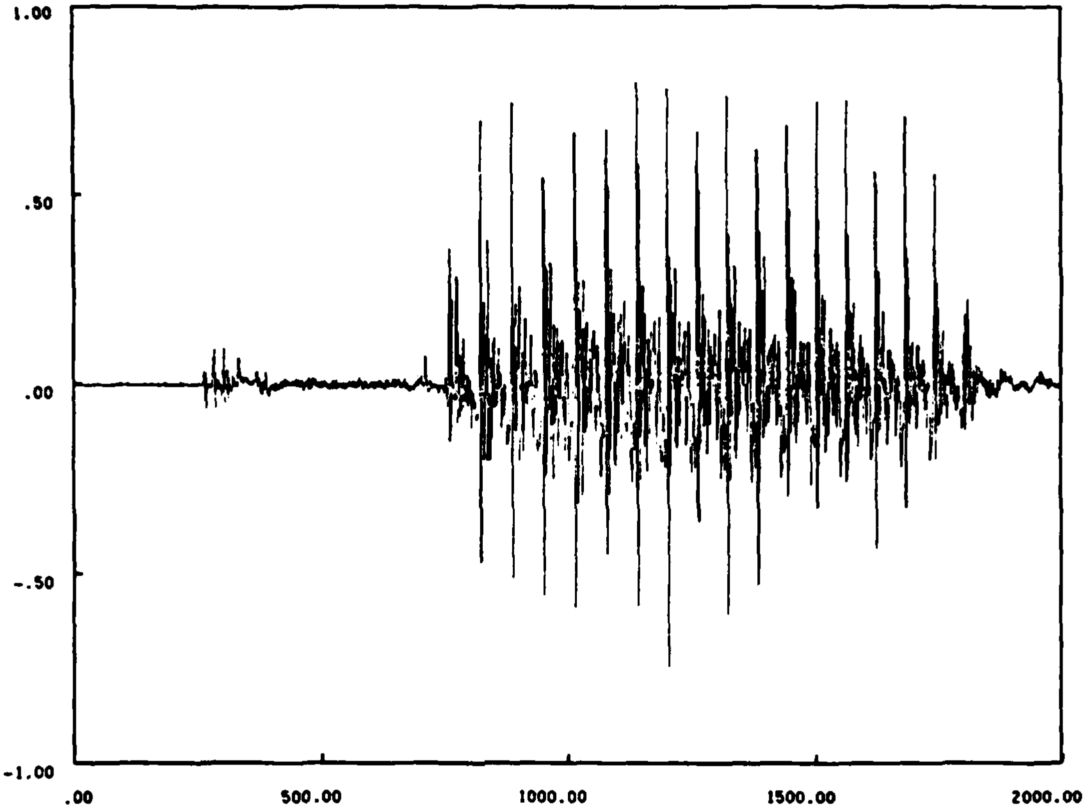


Figure 3.8 FORWARD RESIDUAL (E) of speech data in FILE 1 ,order 10

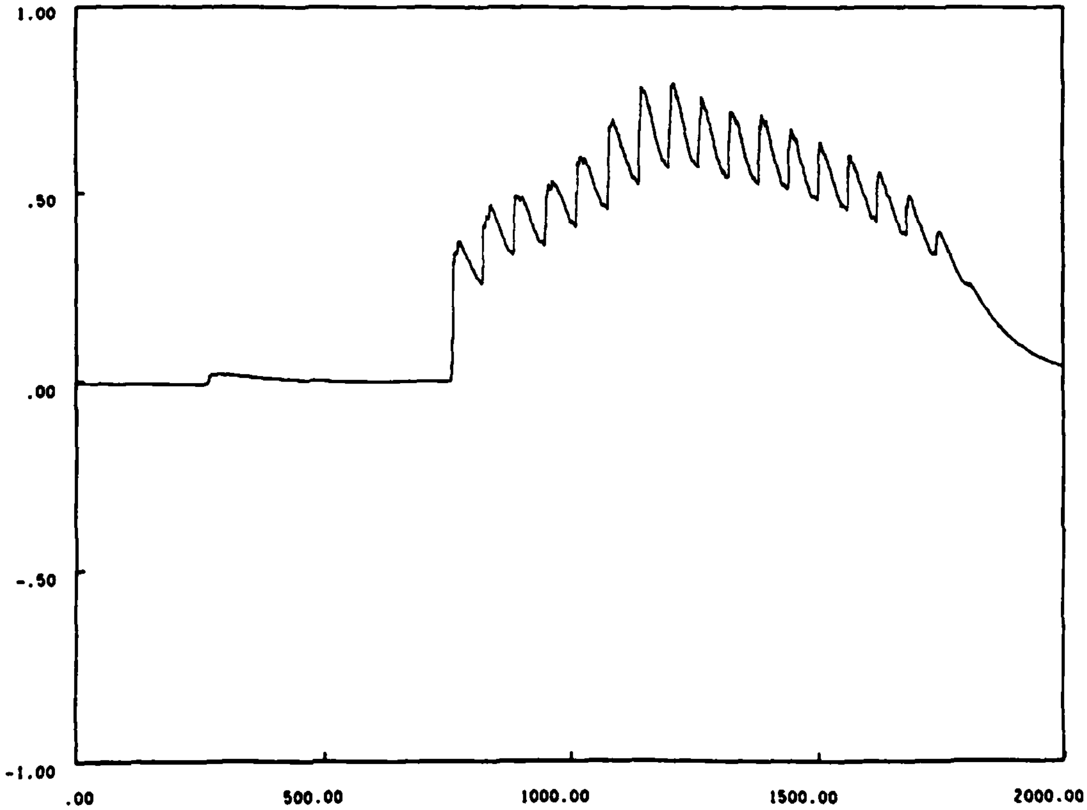


Figure 3.9 FORWARD RESIDUAL COVARIANCE (Re) of speech data in FILE 1 ,order 10

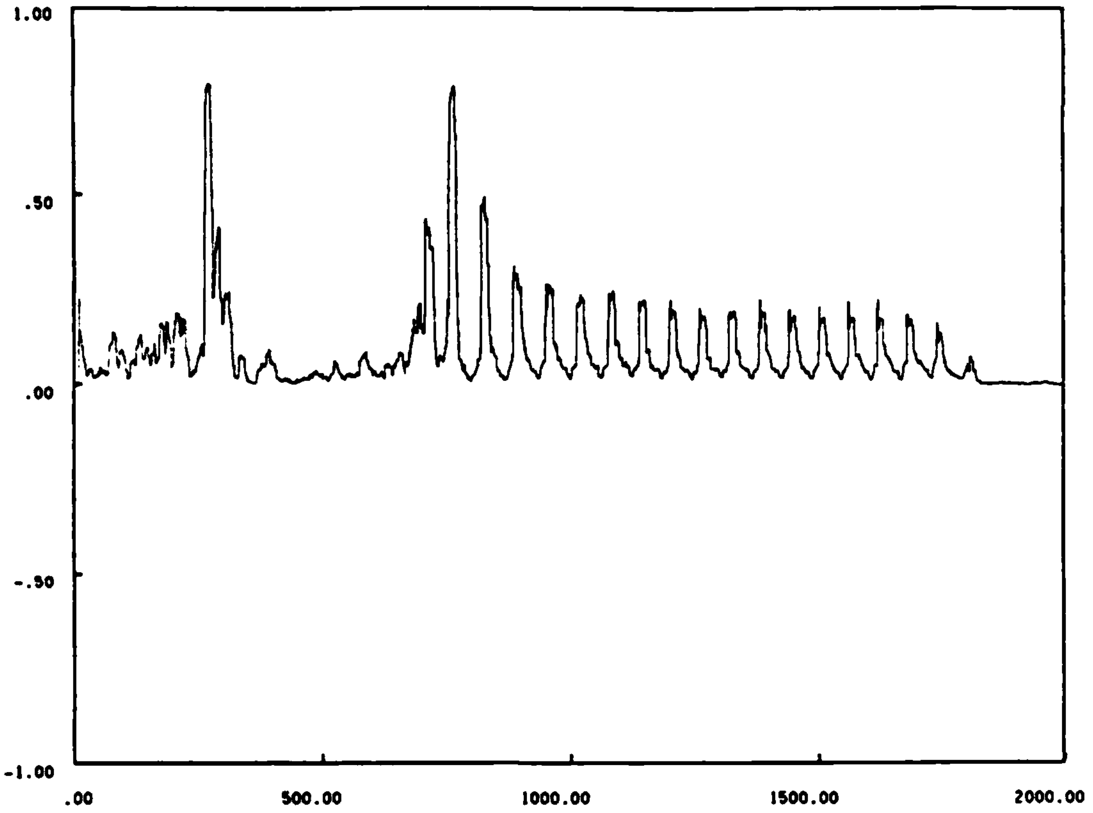


Figure 3.10 GAMMA (6) of speech data in FILE 1 ,order 10

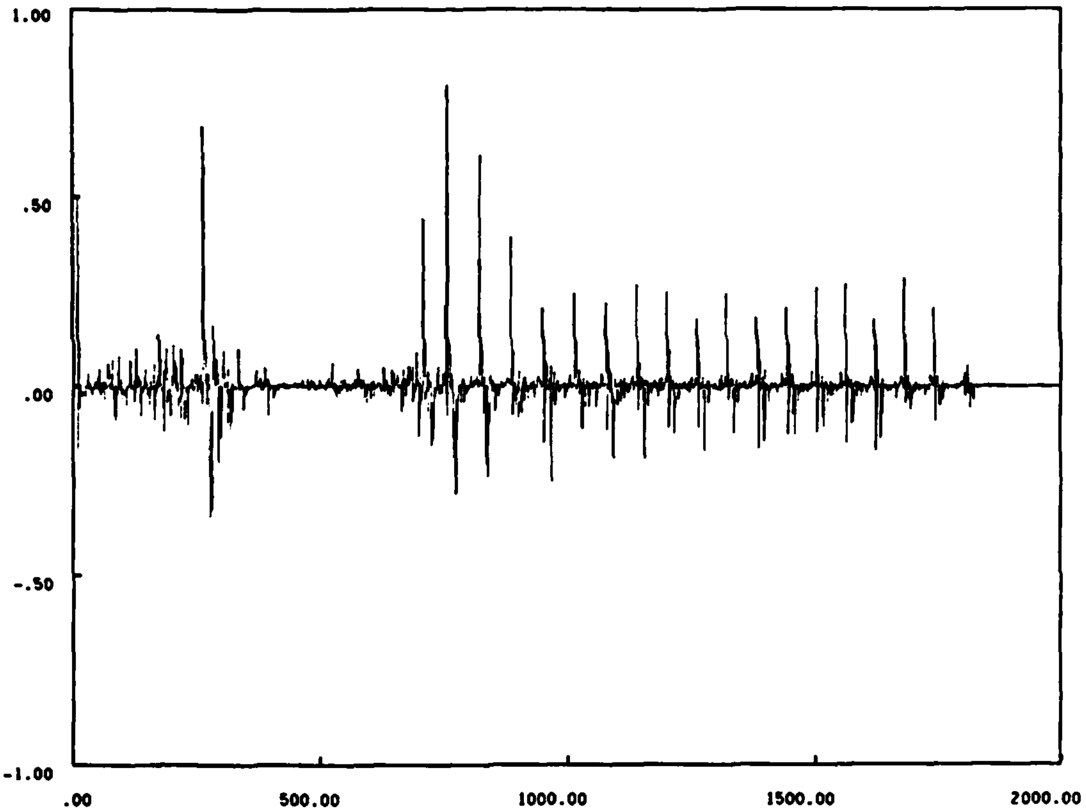


Figure 3.11 DERIVATIVE OF G (dG) of speech data in FILE 1 ;order 10

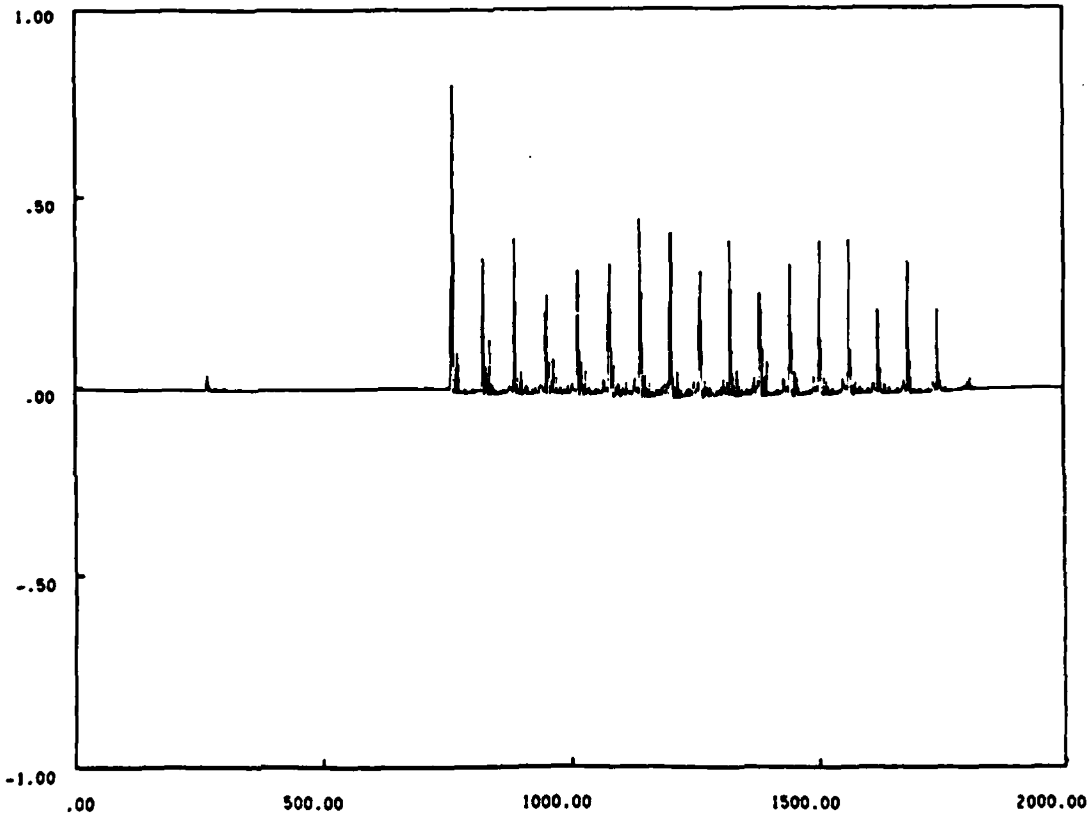


Figure 3.12 DERIVATIVE OF Re (dRe) of speech data in FILE 1 ;order 10

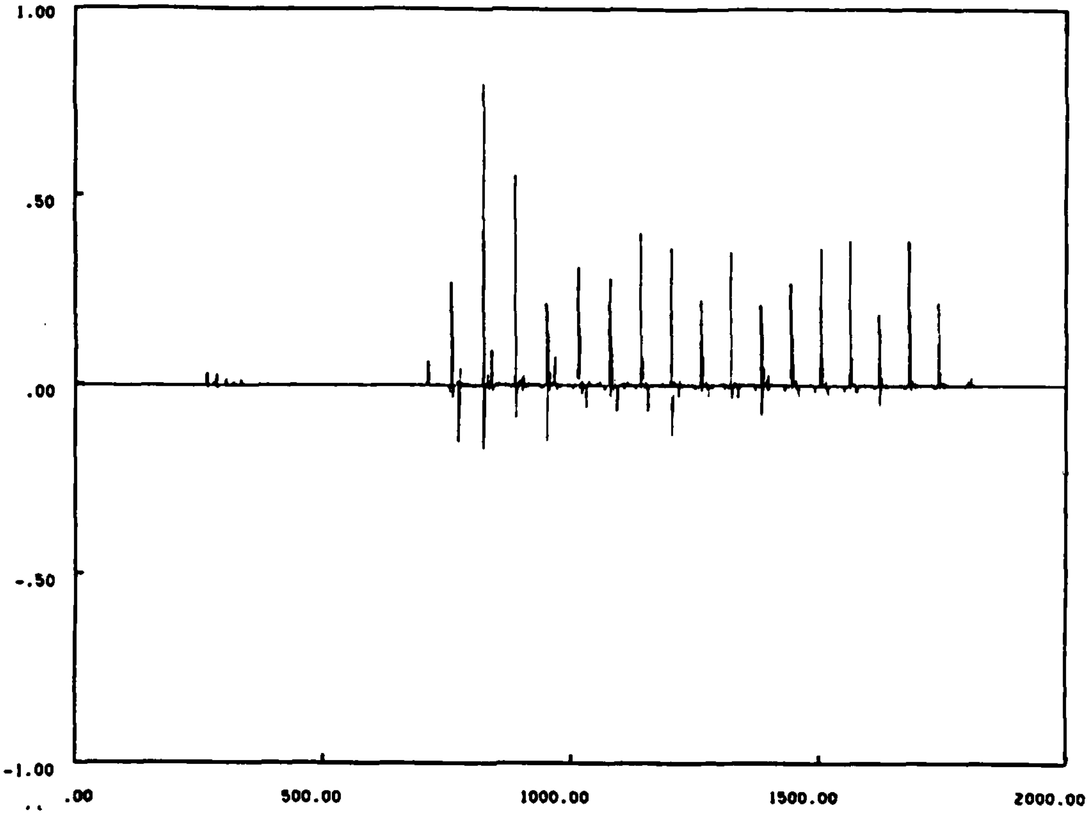


Figure 3.13 dG x E of speech data in FILE 1 ;order 10

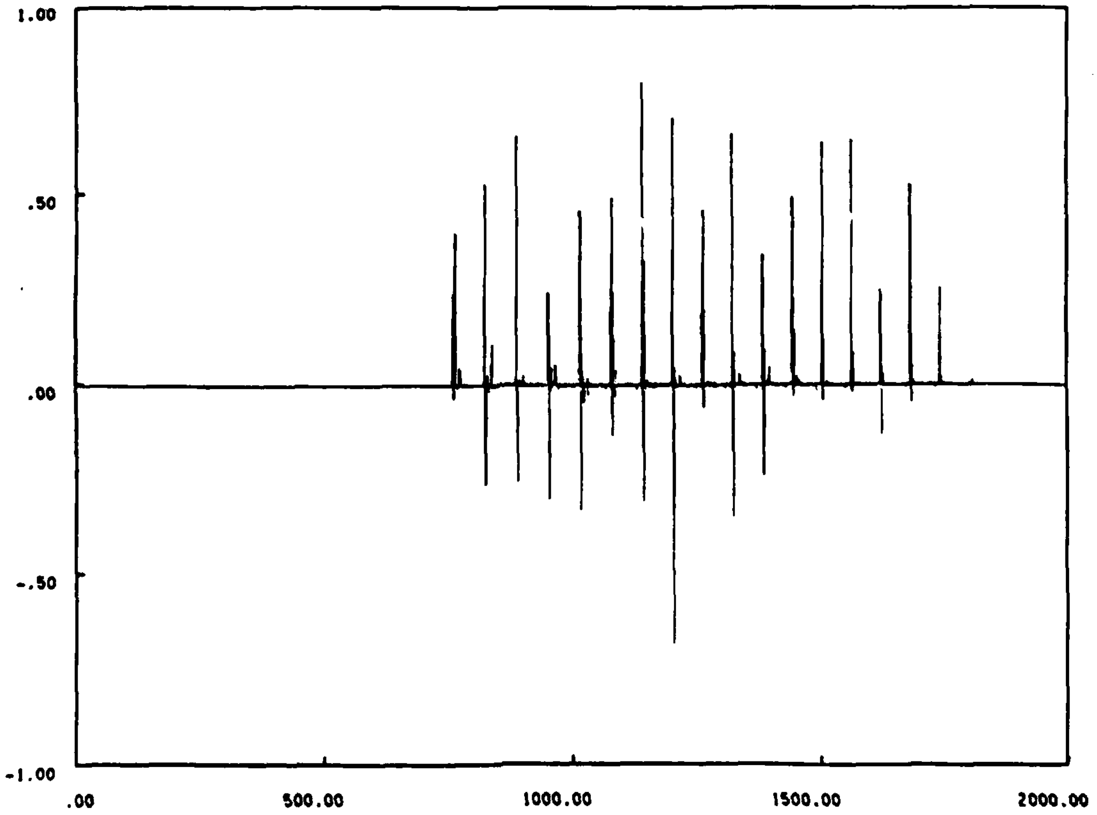


Figure 3.14 dRe x E of speech data in FILE 1 ;order 10

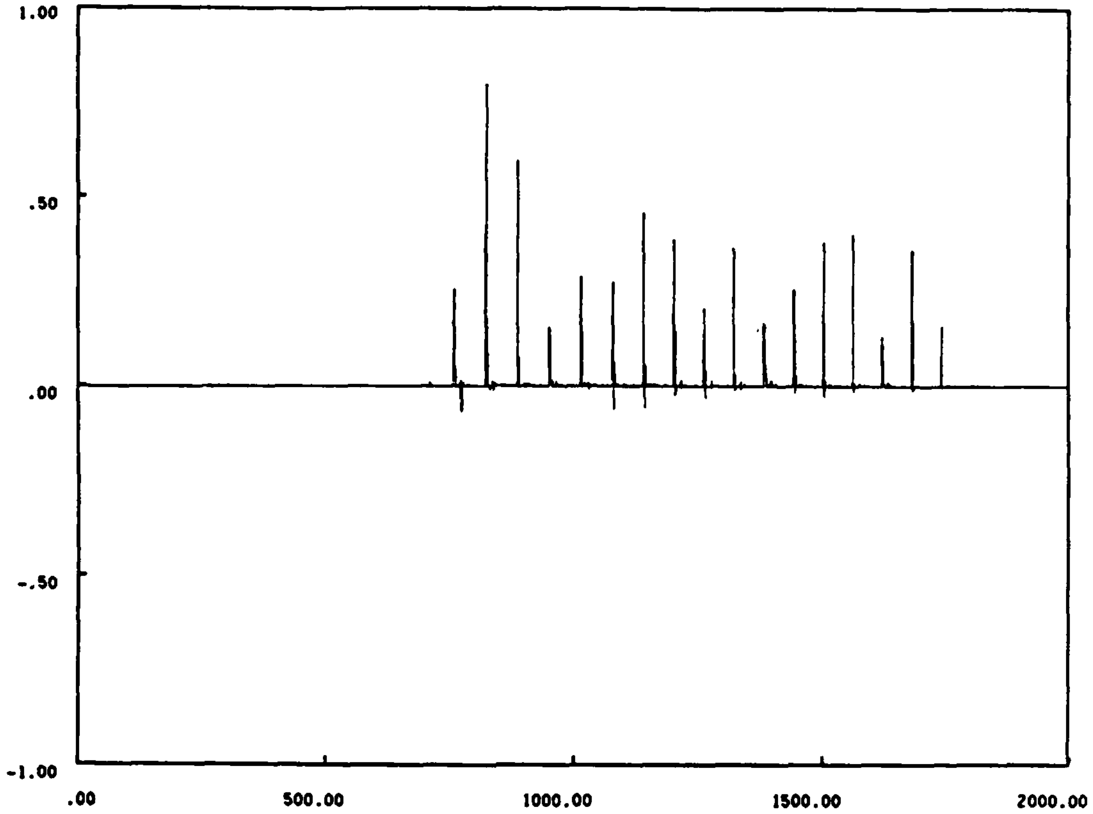


Figure 3.15 dB x dRe of speech data in FILE 1 ,order 10

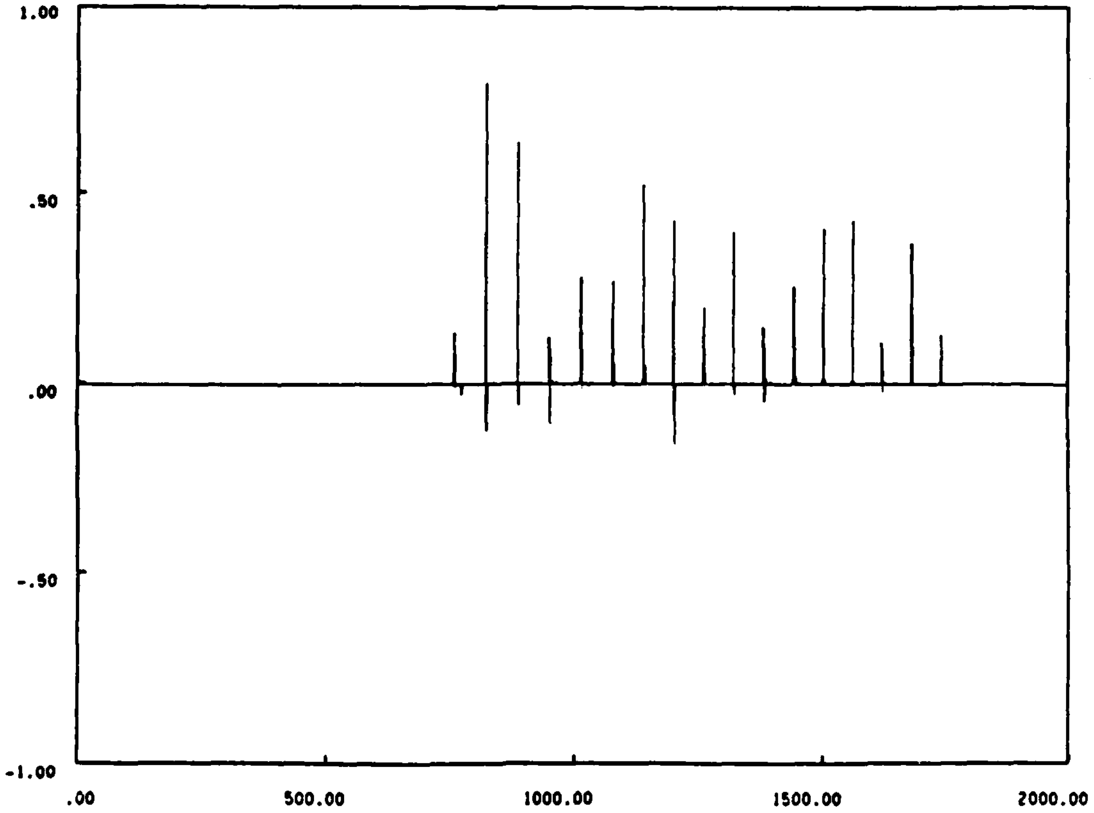


Figure 3.16 dB x dRe x E of speech data in FILE 1 ,order 10

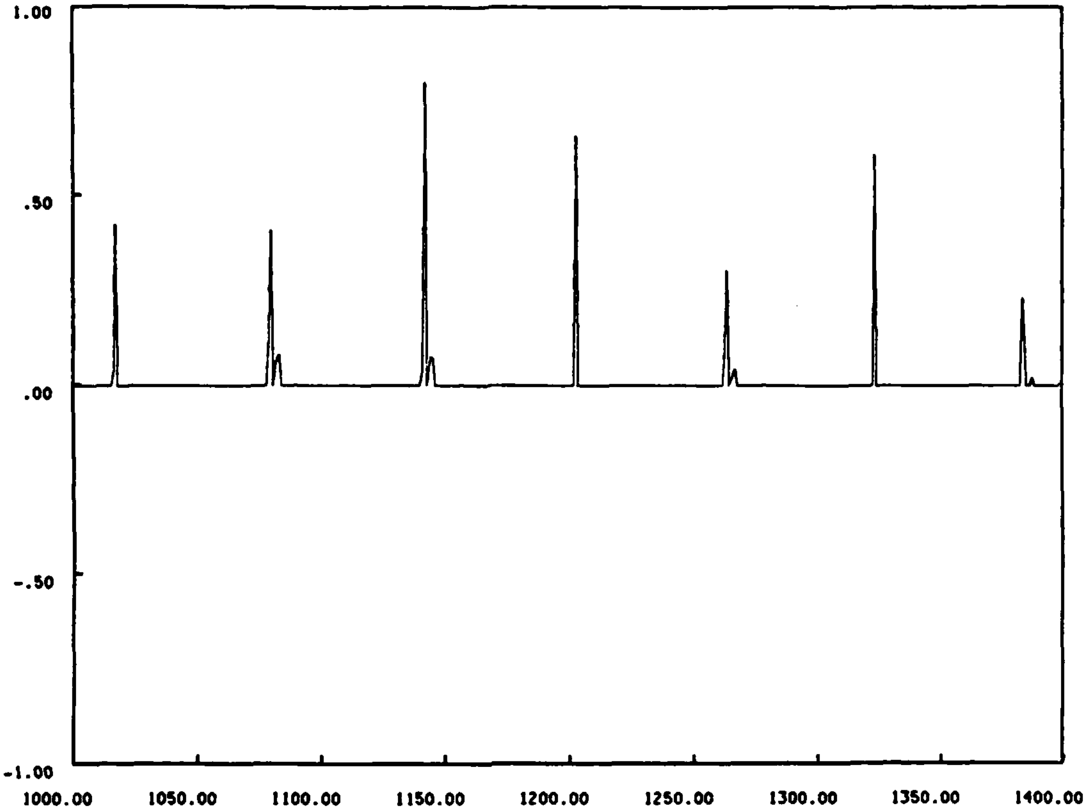
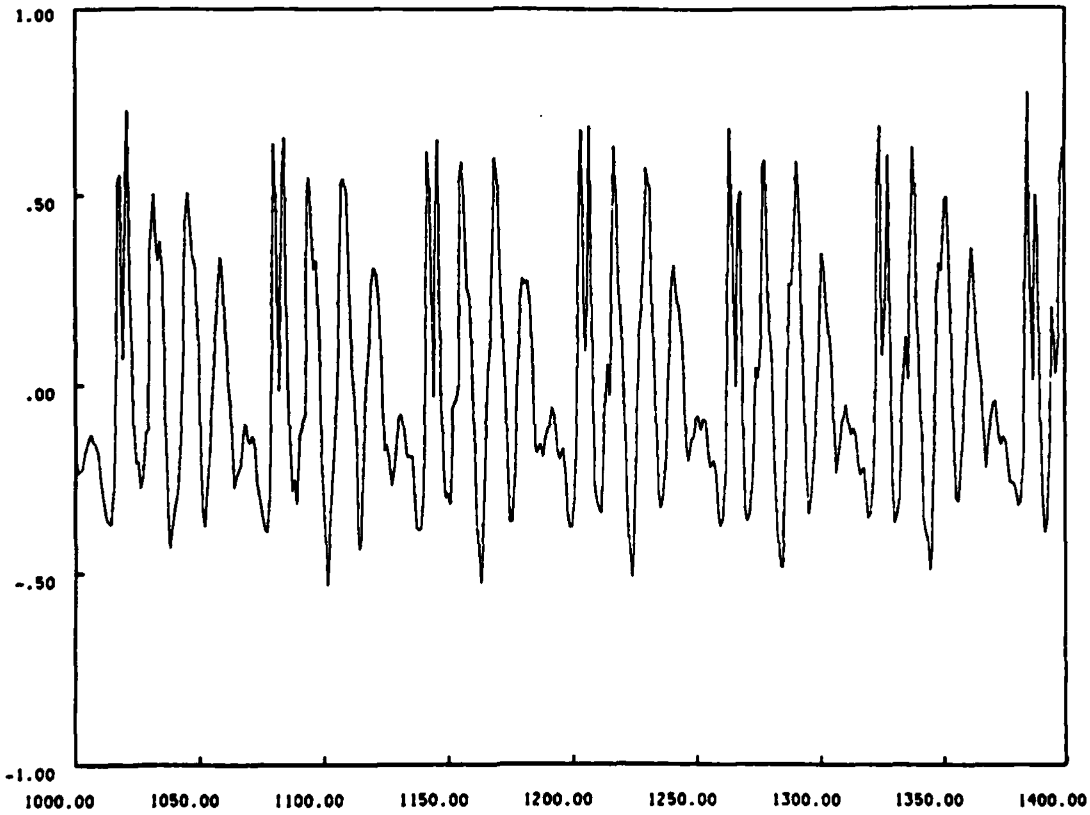


Figure 3.17 POS(dG x dRe x E) of speech data in FILE 1 ;order 10



16-bit digitized speech- FILE 1

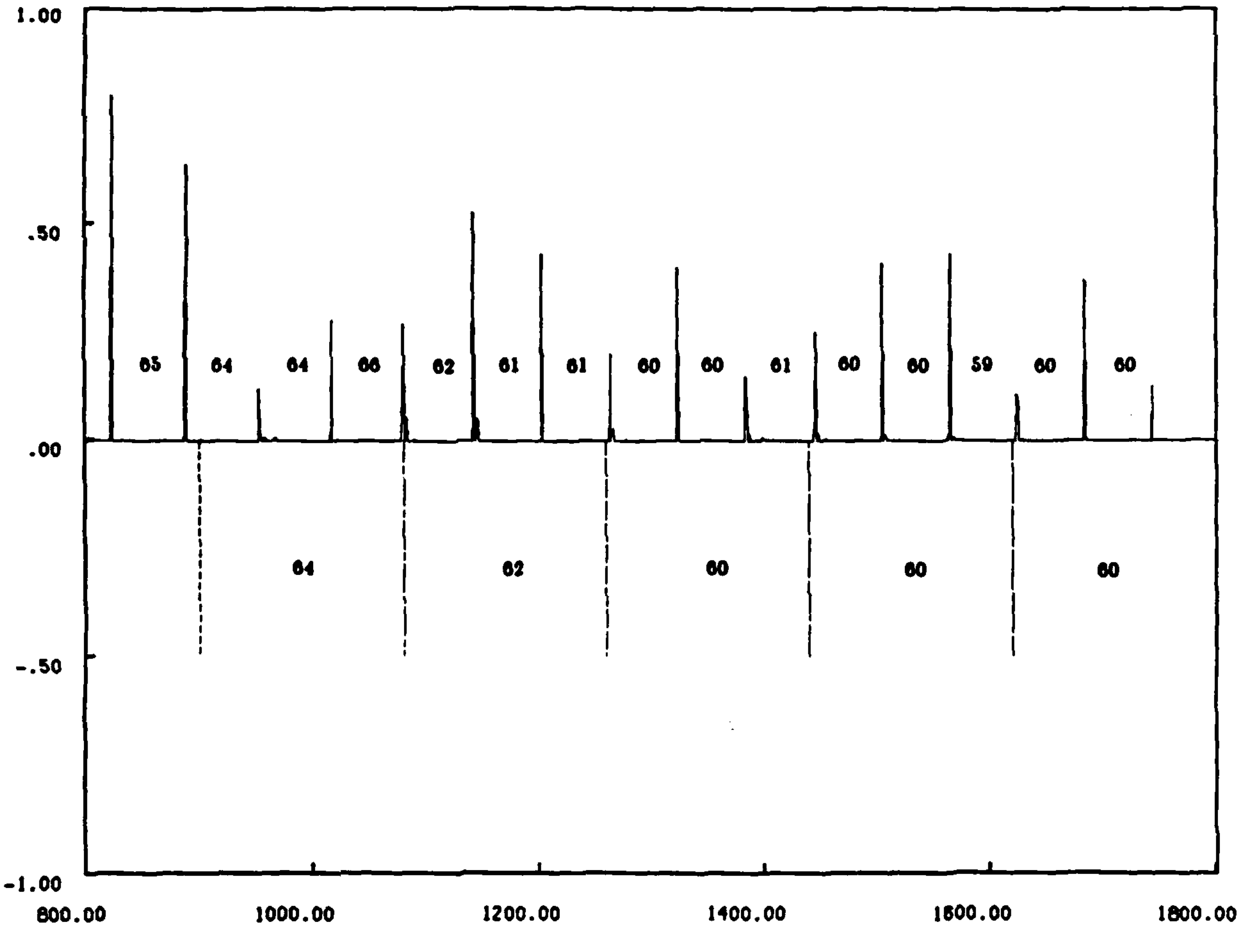


Figure 3.18 POS(dG x dRe x E) and LPC-10, speech data in FILE 1

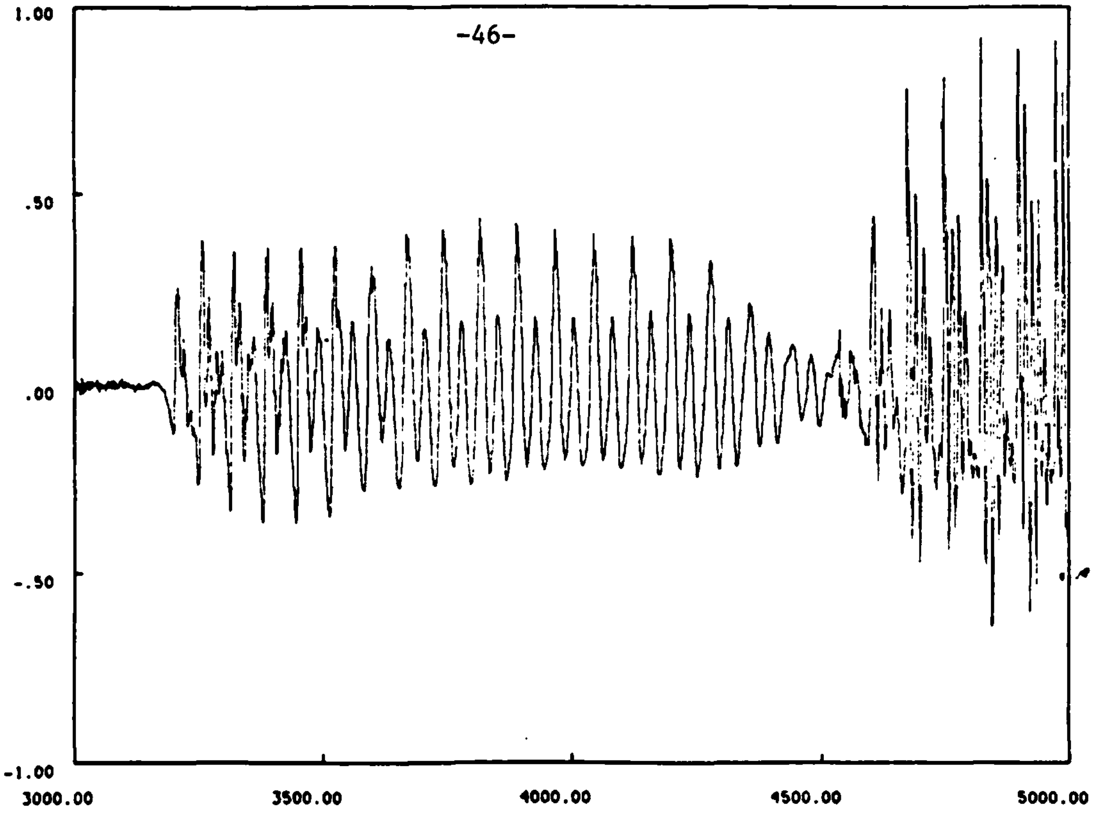


Figure 3.19 16-bit digitized speech- FILE 1

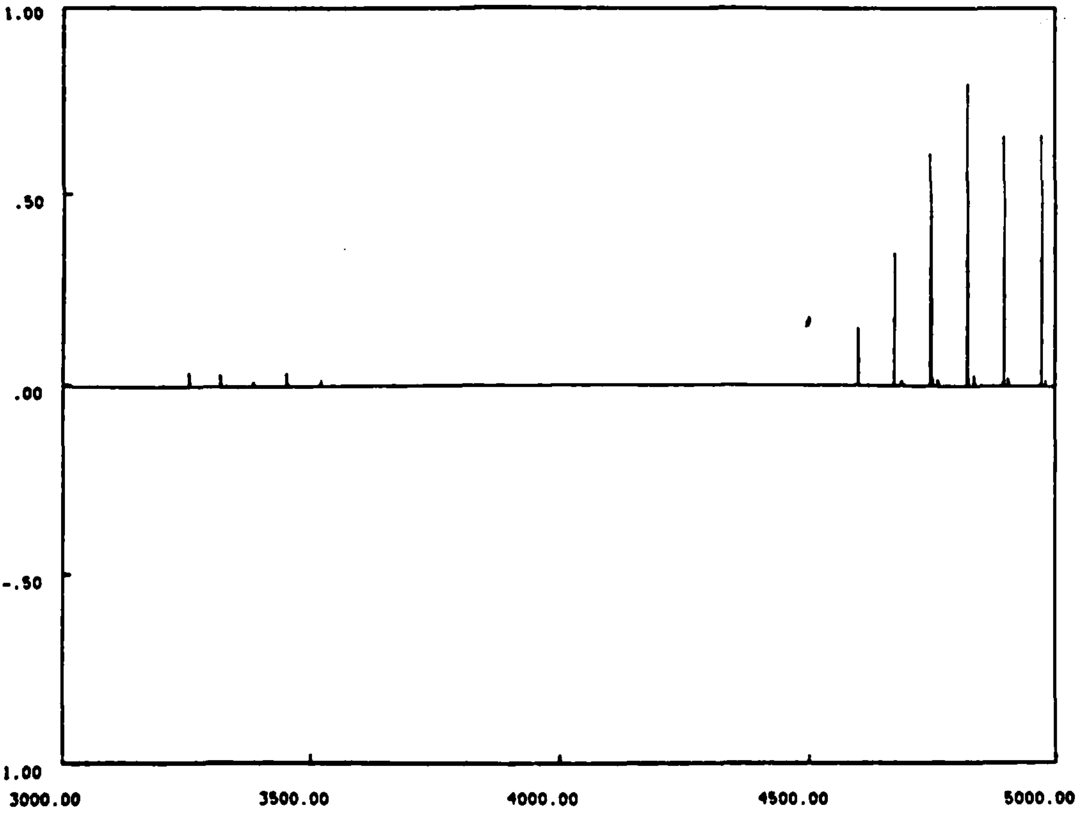


Figure 3.20 POS(dB x dRe x E) of speech data in FILE 1 ;order 10

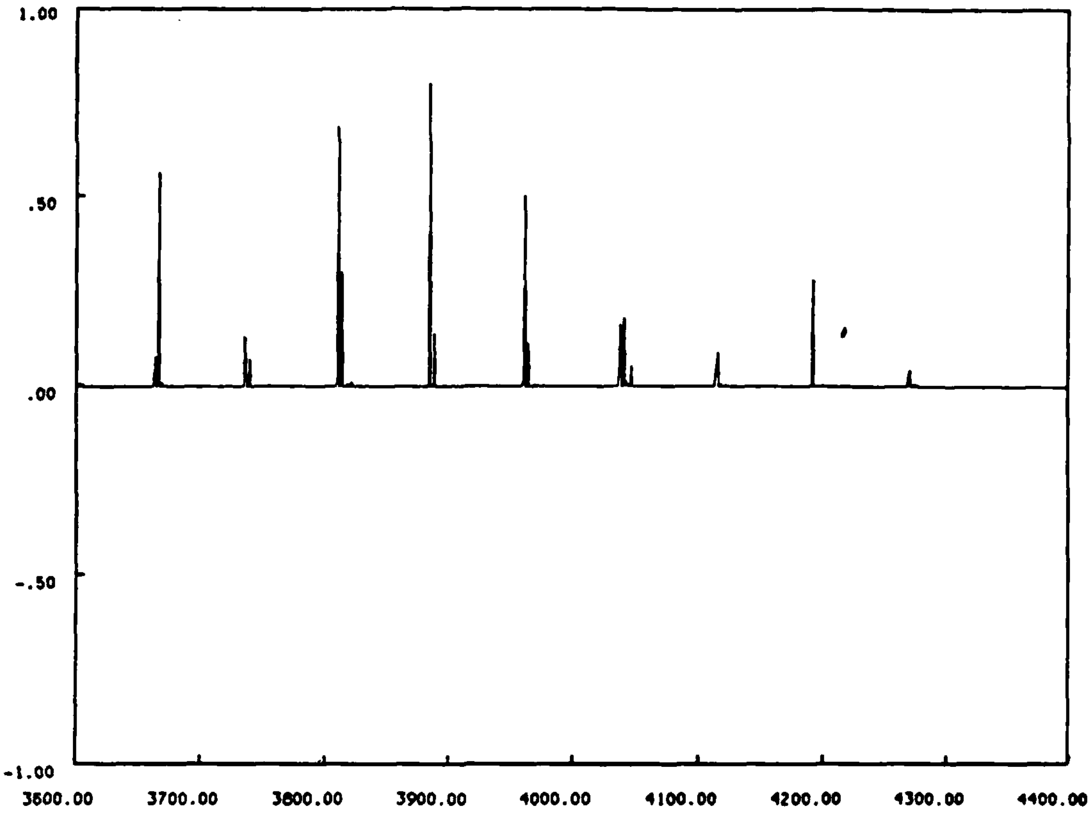
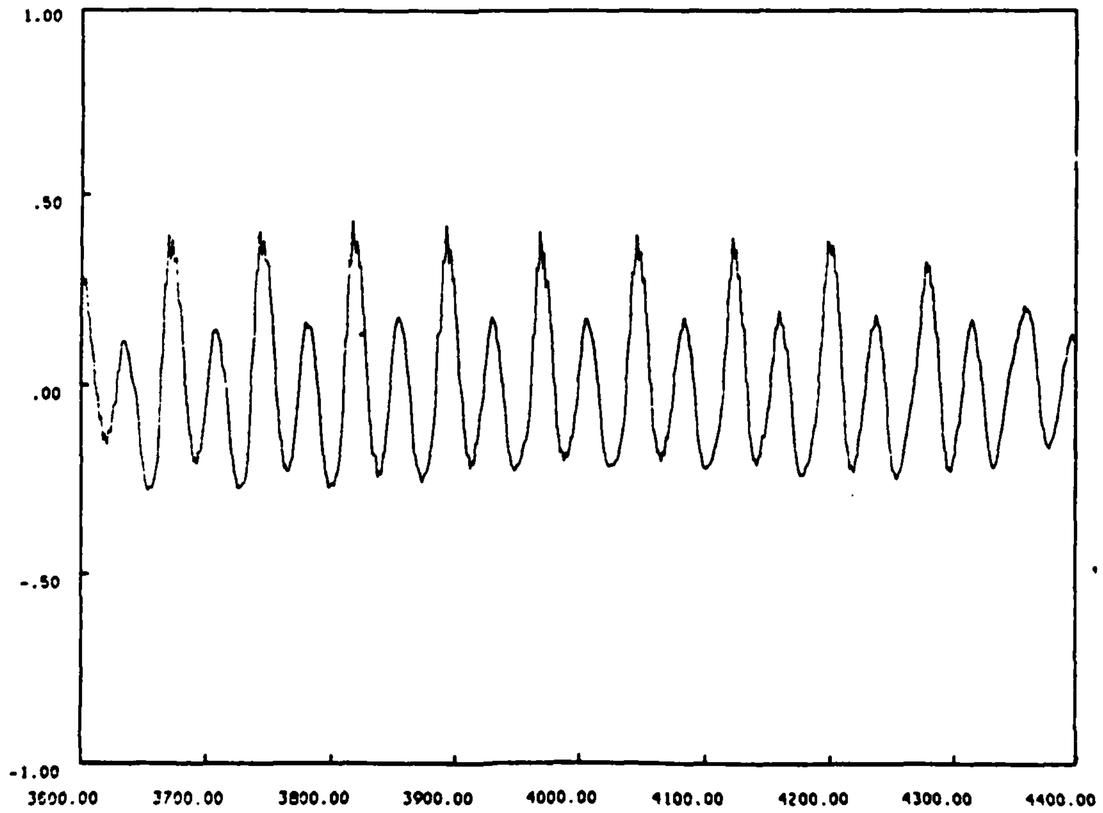


Figure 3.21 POS(dG x dRe x E) of speech data in FILE 1 ,order 10



16-bit digitized speech- FILE 1

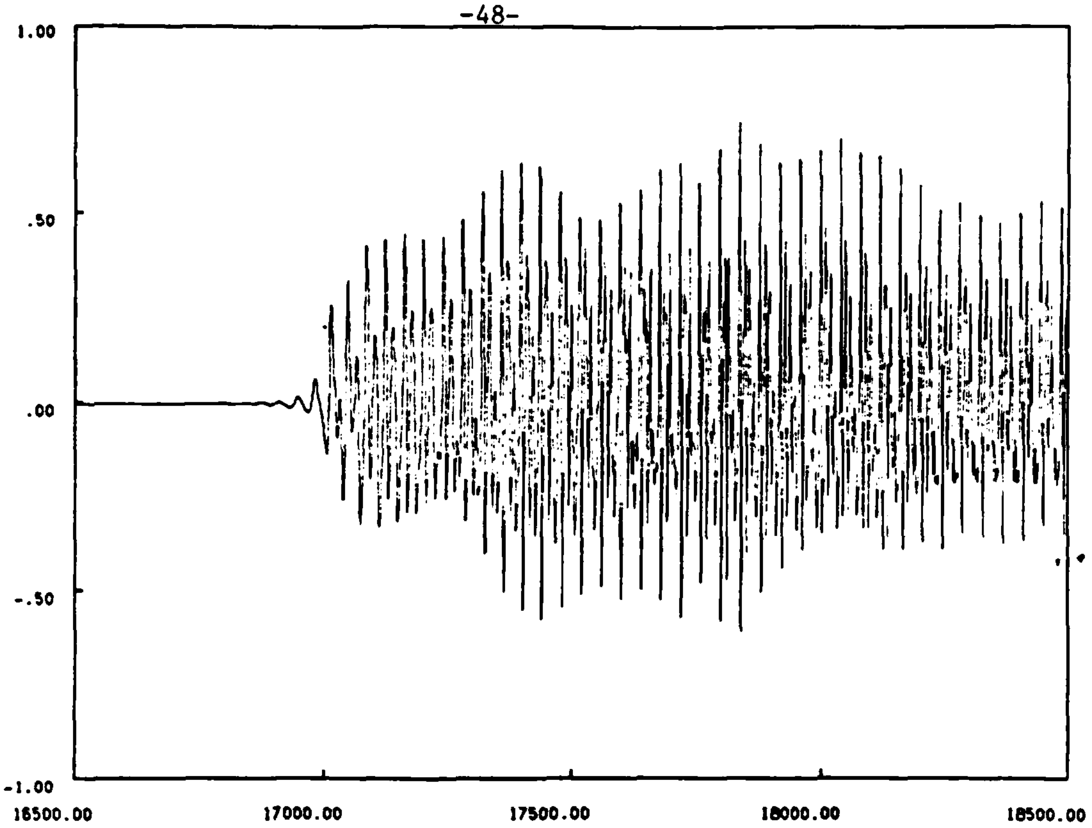


Figure 3.22 16-bit digitized speech- FILE 2

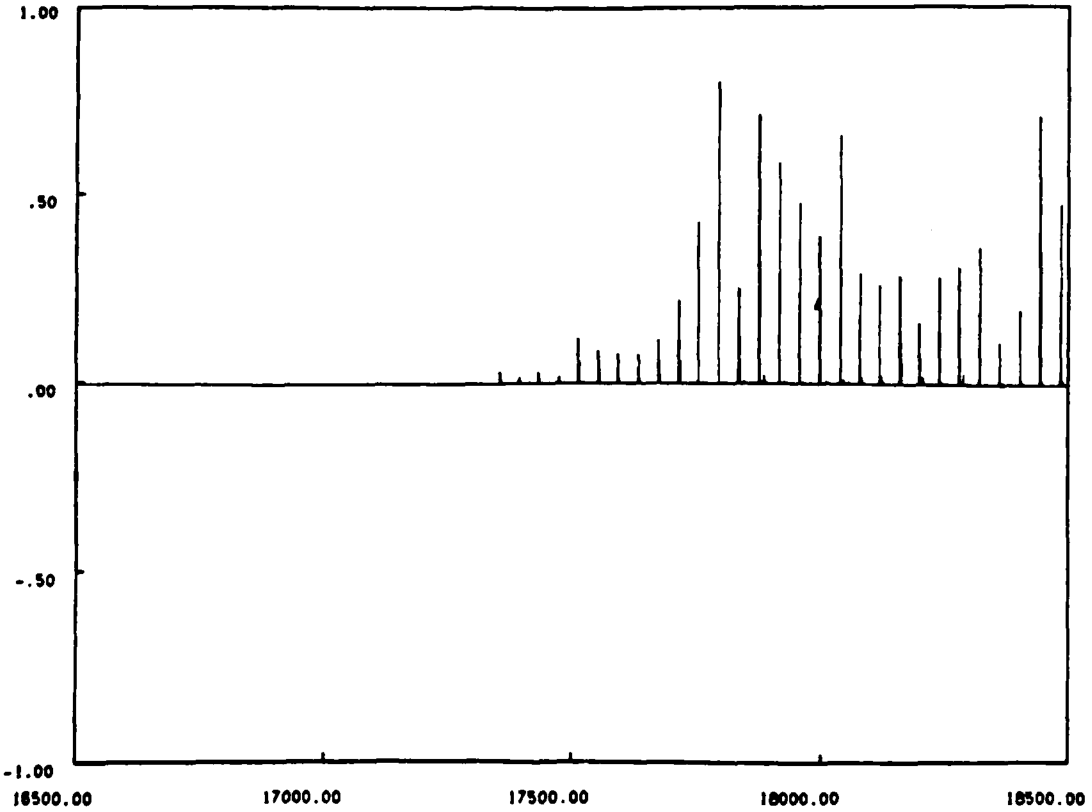


Figure 3.23 POS(dB = dRe = E) of speech data in FILE 2 ,order 10

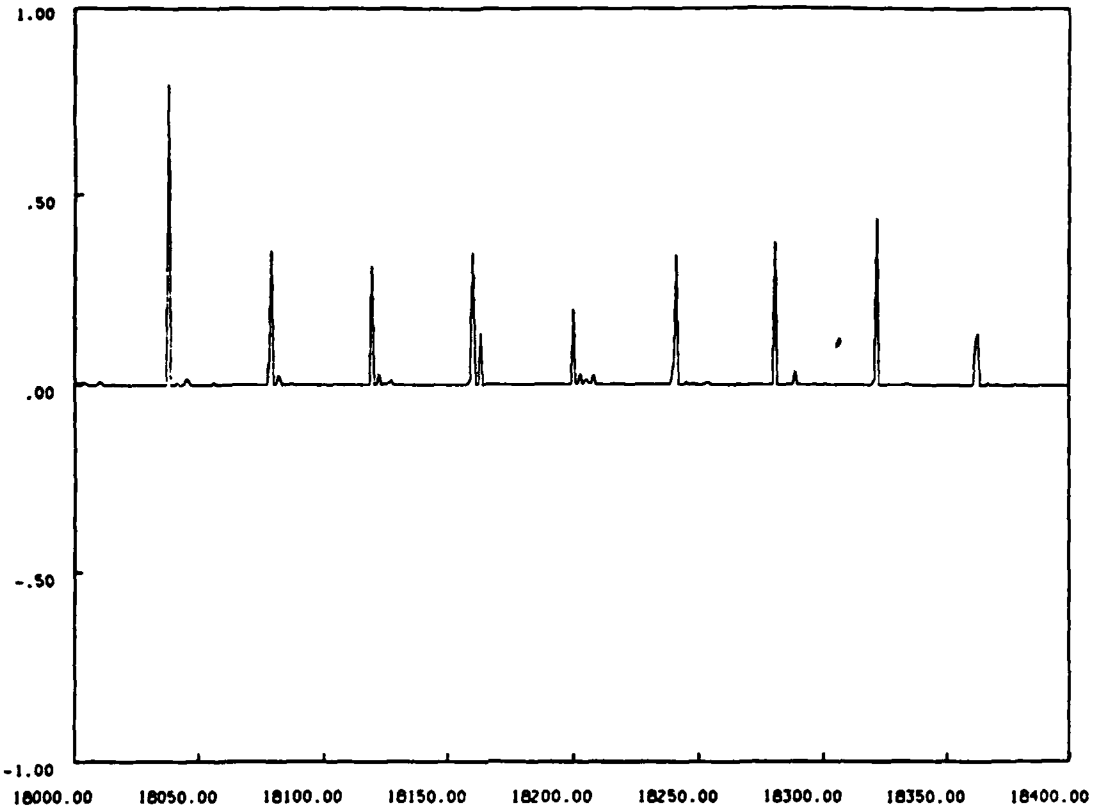
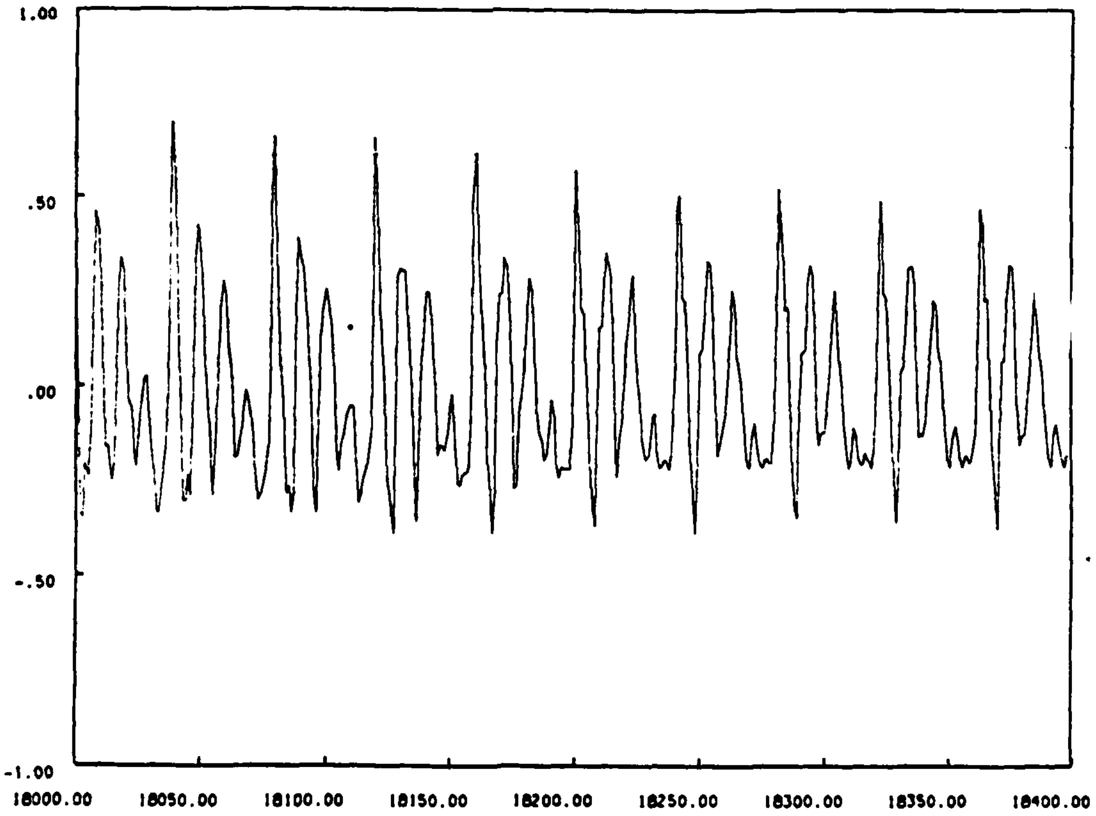


Figure 3.24 POS(dG x dRe x E) of speech data in FILE 2 ;order 10



16-bit digitized speech- FILE 2

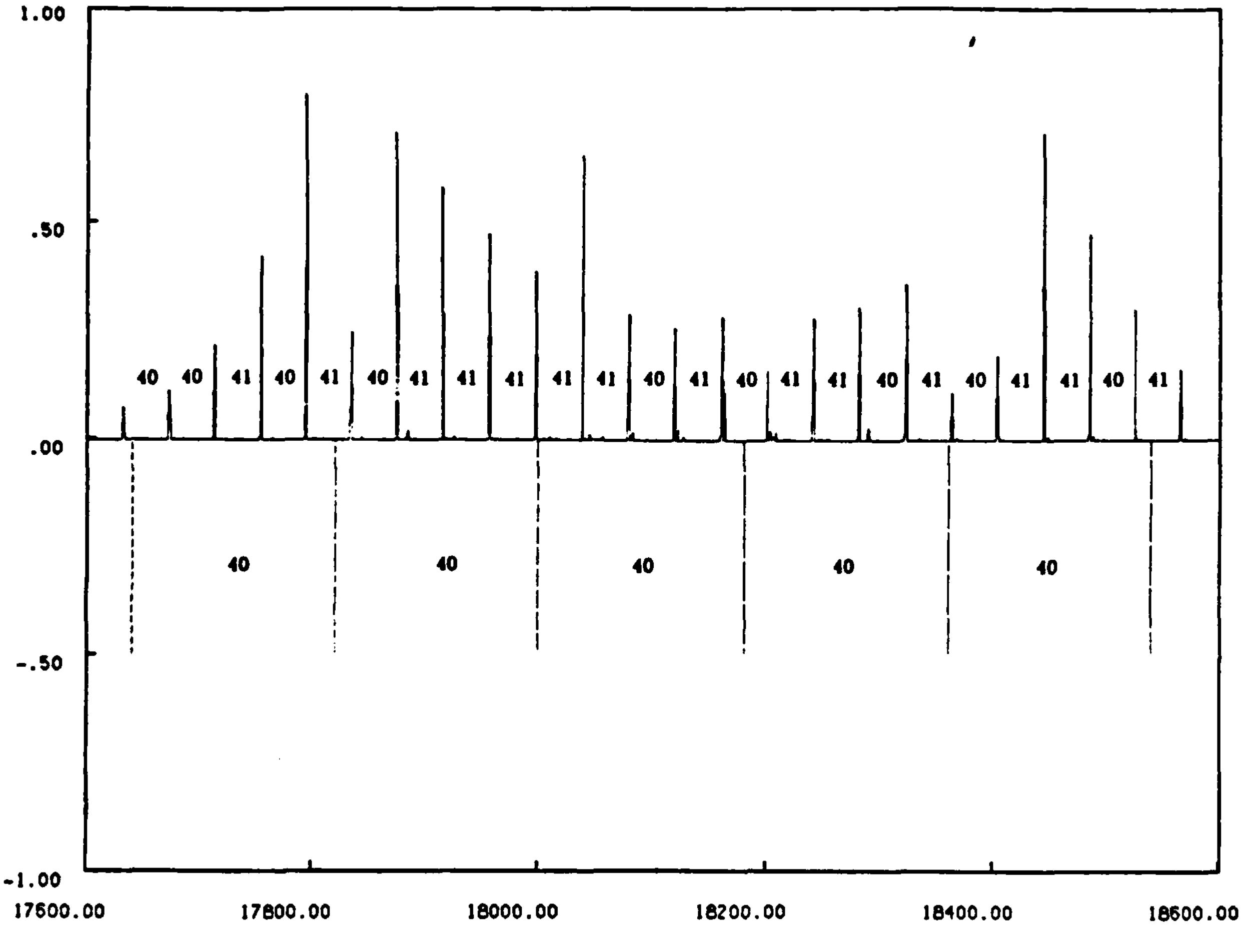


Figure 3.25 POS(dG x dRe x E) and LPC-10, speech data in FILE 2

4. RESEARCH ON RECOGNITION OF STOP CONSONANTS

4.1 INTRODUCTION

Current speech recognition techniques can accurately determine the vowels within a particular word since vowels are of relatively long duration and change character slowly. Within the class of consonants, the stop consonants are hard to distinguish due to their short duration and transient nature. In order to recognize these transitional sounds, an estimation technique that can track the changes is necessary. The approach presented here utilized the recursive exact least square lattice estimation algorithm to determine an autoregressive model (hence a spectral representation) of the speech. This recursive algorithm updates its representation at every speech sample using exponentially weighted past data. Thus it is possible to track the spectral changes in the speech without much time smearing. The experiments performed here on natural speech data were motivated by an attempt to better characterize the fast transitions that occur in stop consonants. A representation based on trajectories of appropriate speech parameters was developed and analyzed.

The region of first and second formant (spectral peak) where each vowel typically occurs was determined by Peterson and Barney [PB]. Diphthongs follow a trajectory within a known region between the vowels. The current understanding of speech perception has not clearly identified whether spectral characteristics are sufficient to distinguish transient consonants such as stops. If the parameterization for stops was dependent on the following (preceding) vowel, the task of automatic speech recognition would be more difficult. The results of Stevens and Blumstein [BS] indicate that the place of articulation for stop consonants is cued by spectral properties in the 10 to 20 milliseconds period initiated by the burst onset. Their studies indicate that the spectral properties of this short time interval appear to be invariant of the following vowel. This burst onset of the stops lasts less than 160 speech samples at an 8 kHz. sampling rate. Once the formant transition has started, the transition is dependent on the following vowel but can be used as a context dependent cue for determination of the consonant, if the voicing condition is already known. Fast estimation techniques are necessary to determine the speech spectra over such a short time interval.

The fast recursive exact least square lattice algorithm developed by Morf et al. [LM] estimates the signal spectral by fitting an autoregressive model. By determining a new estimate for every speech sample using an exponential weighting of past data allows the estimates to keep up with the short time signal characteristics. Section 4.2 describes the recursive lattice estimation algorithm. This algorithm was applied to natural speech words from a set of Diagnostic Rhyme Test word. The voiced stops /b/, /d/ and /g/ followed by various vowels, spoken by a single male speaker were examined in detail.

The process of clustering observations should be insensitive to a transformation of variables provided the distance metric is appropriately changed. Thus a clustering in the space of reflection coefficients with a suitable metric is equivalent to frequency domain clustering. The technique called Vector Quantization (VQ) was used in this study to perform the clustering of parameters. The standard VQ algorithm is presented in Section 4.3. Experiments applying the technique of vector quantization, appropriately modified, have determined a suitable parameterization for distinguishing the stop consonants. These modifications to the standard VQ technique are discussed in Sections 4.6 and 4.7.

Section 4.4 discusses the results of applying the standard VQ method to consonant-vowel words. Section 4.5 looks at the differences in the same vowel spoken in different words. Section 4.6 introduces an augmented parameterization that includes information about reflection coefficient trajectories that can assist in classifying stop consonants. Section 4.7 presents a new Classified Vector Quantization method and its application to consonant-vowel words. Section 4.8 summarizes the results of our procedure to recognize the voiced stops, /b/, /d/, /g/. A summary and discussion of future research is in Section 4.9.

4.2 RECURSIVE LATTICE ESTIMATION ALGORITHM

An alternative parameterization of an autoregressive model is in terms of reflection coefficients $\{\rho_i\}$, in the lattice filter structure. The lattice structure can be related to the transfer function of an acoustical tube formed from connected cylinders of differing diameters. The propagation of acoustic waves down the tube experiences reflections and transmissions at each discontinuity. The reflection coefficients of the lattice filter structure can be related to the signal propagation across a discontinuity in the acoustic tube model. Furthermore, the reflection coefficients can be interpreted as correlation coefficients between the signals in the two paths of the lattice structure. Thus the process of estimating reflection coefficients is similar to orthogonalizing the observed signal with respect to its delayed version. This is one reason why spectral estimation by reflection coefficients has been shown to adapt quickly. These techniques have been used successfully in speech analysis and synthesis, fast adaptive equalization and spectral estimation.

Recently developed techniques by Morf et al. [LM] recursively update reflection coefficient estimates as new data samples are observed with exponential decay of past data. This algorithm solves for the exact least squares fit to the observed data. The square root normalized algorithm, (4.1), has a very compact notation and normalizes all signals to unit variance at each stage. The response of this algorithm to synthetic signals with time varying characteristics and to speech phrases was first studies in [ML].

$$\rho_{n+1,T} = \sqrt{1-\nu_{n,T}^2}\sqrt{1-\eta_{n,T-1}^2}\,\rho_{n+1,T-1} + \nu_{n,T}\eta_{n,T-1}$$

$$\nu_{n+1,T} = \frac{\nu_{n,T} - \rho_{n+1,T}\eta_{n,T-1}}{\sqrt{1-\rho_{n+1,T}^2}\sqrt{1-\eta_{n,T-1}^2}} \tag{4.1}$$

$$\eta_{n+1,T} = \frac{\eta_{n,T-1} - \rho_{n+1,T}\nu_{n,T}}{\sqrt{1-\rho_{n+1,T}^2}\sqrt{1-\nu_{n,T}^2}}$$

The tracking ability of the algorithm can be seen from the first four reflection coefficients computed from the first 40 ms. of 'did' and 'bid', see Fig. 4.1. The burst of the /d/ or /b/ and the transition to the steady vowel /i/ is seen in the time waveform. The pitch pulses cause momentary fluctuations in the coefficient values. The initial trajectories of the reflection coefficients are seen to be different, particularly at higher than first order. Yet they all converge to similar values after the onset burst as the vowel sound stabilizes. Note that the vowel oscillation commences at about the same time in both words. The rise of the first reflection coefficient is different during the burst onset. The second reflection coefficient is more steady in 'b' and changes suddenly at the beginning of voicing oscillation. The third and fourth coefficients have different values for the different stop consonants. Certain similarities were noted in the reflection coefficient trajectories for all the trial words starting with 'b', and likewise for 'd'. The reflection coefficients determine a spectral representation so the formants (spectral peaks) can be estimated. The second formant illustrates a rising trend for 'b' with less of a change for 'd'. The acoustic models for the stop consonants differentiate each by the slope of the second formant.

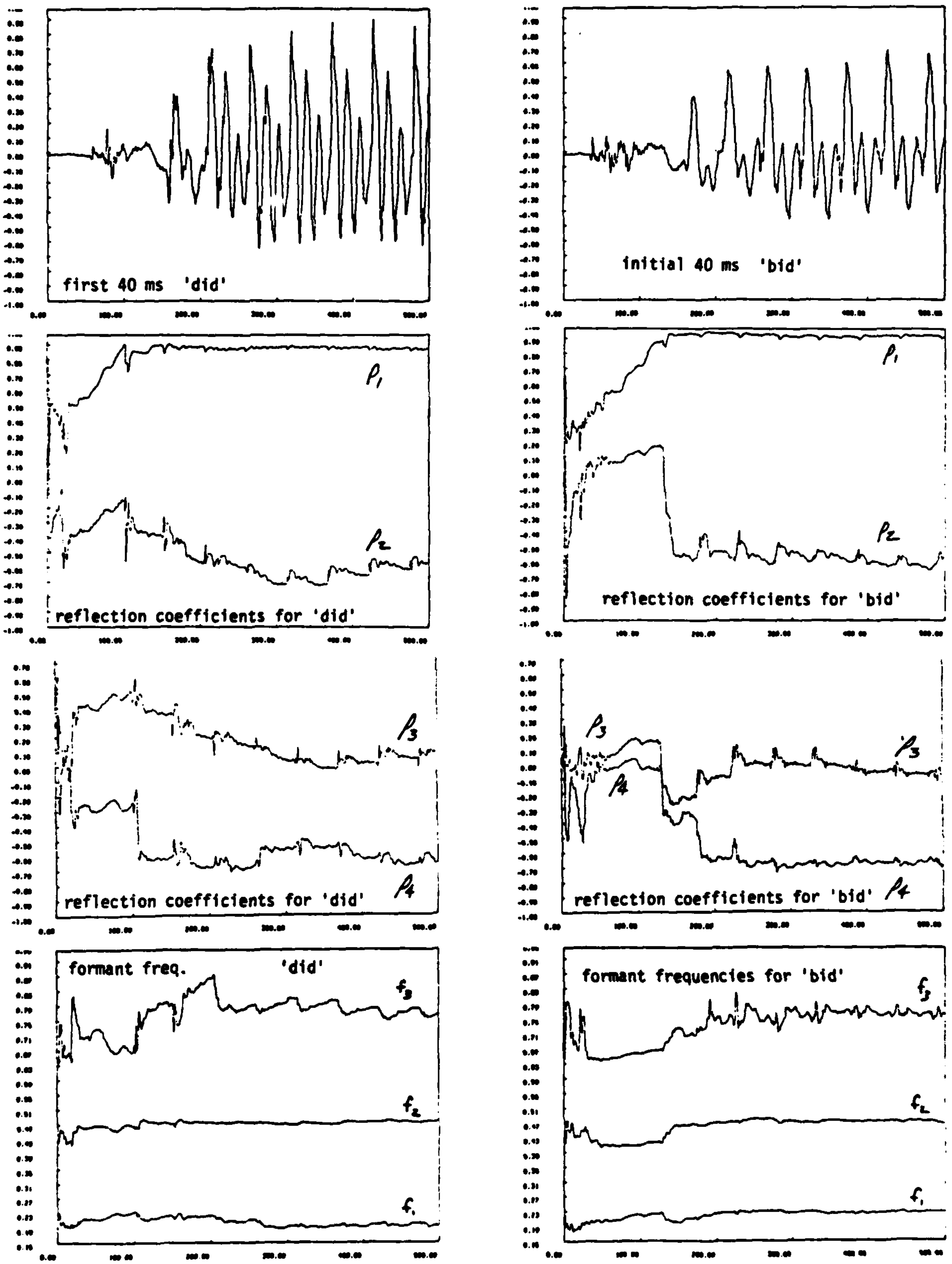

first 40 ms 'did'
initial 40 ms 'bid'
ρ_1
ρ_2
reflection coefficients for 'did'
reflection coefficients for 'bid'
ρ_3
ρ_4
reflection coefficients for 'did'
reflection coefficients for 'bid'
formant freq. 'did'
formant frequencies for 'bid'
f_3
f_2
f_1

Figure 4.1

4.3 VECTOR QUANTIZATION

Vector quantizers have been used for waveform and voice coding systems. Our application of the vector quantization technique is to perform a clustering of speech sounds into categories that can be identified with vowels and consonants. First the general framework of Vector Quantization is presented. A vector quantizer maps input vectors drawn from the M-dimensional Euclidean space $\mathbf{R}^M$ into a finite set (codebook) of reproduction vectors (codewords) contained in the space $\mathbf{R}^K$. A vector quantizer (VQ) is described by the input vector dimension (M), the reproduction vector dimension (K), the number of reproduction vectors (N), the set of reproduction vectors $C = \{\hat{\mathbf{x}}_i, i = 1,2, \cdots N\}$, and the mapping of the input space into the set of reproduction vectors $\hat{\mathbf{q}}(\mathbf{x})$. In our studies the reproduction vector is of the same dimension as the input vector, $M = K$.

A VQ used to compression speech for transmission requires two functional blocks: an encoder, which views the input vector $\mathbf{x}$ and generates the index of the reproduction vector specified by $\hat{\mathbf{q}}(\mathbf{x})$; and a decoder, which uses this index to generate the reproduction vector $\hat{\mathbf{x}}_i$. A VQ can be used to communicate over a digital channel by placing the encoder at the transmitter and the decoder at the receiver and sending the index of the codeword across the channel. For speech compression, each input LPC vector is mapped into a codeword of $\log_2 N$ bits per vector. The bit rate is $\log_2 N$ bits times the rate of generation of LPC vectors. As the bits per vector increases, the codebook size grows exponentially requiring a similar increase in computational effort and storage at both the encoder and decoder. The decoder stores the codebook and performs the simple task of looking up the reproduction vector indexed by the encoder. The encoder has the more complicated task of partitioning the input space into a collection of bins according to $\hat{\mathbf{q}}(\mathbf{x})$, one bin for each reproduction vector in the codebook, and determining in which bin an input vector is contained.

If we define a distortion measure $\mathrm{d}(\mathbf{x};\hat{\mathbf{x}})$ which represents the penalty or cost associated with reproducing a vector $\mathbf{x}$ by $\hat{\mathbf{x}}$, then the best mapping $\hat{\mathbf{q}}(\mathbf{x})$ is the one which selects as the reproduction vector for $\mathbf{x}$ the codeword $\hat{\mathbf{x}}_i$ that minimizes $\mathrm{d}(\mathbf{x};\hat{\mathbf{x}}_i)$. With such a minimum distortion or nearest neighbor mapping, the encoder operates by computing $d(\mathbf{x};\hat{\mathbf{x}}_i)$ for $i = 1,2, \cdots N$, and then selecting the value of i (by a full search) for which $d(\mathbf{x};\hat{\mathbf{x}}_i)$ is minimized. This implies that the bin associated with a particular codeword $\hat{\mathbf{x}}_i$ is the set of input vectors for which $\hat{\mathbf{x}}_i$ is the minimum distortion codeword.

Vector quantization applied to LPC voice coders is used to encode and decode the autoregressive model generated by an LPC analysis of a speech frame. (The coding of the excitation parameters is not considered here.) The LPC speech model is shown in (4.2).

$$\sigma/(1 + a_1 z^{-1} + a_2 z^{-2} \cdots + a_p z^{-p}) = \sigma/A(z) \tag{4.2}$$

The order p used here is 10. Once the model parameters $\{\sigma, a_1, a_2, \cdots a_p\}$ have been obtained, they are coded by means of vector quantization. The input vector $\mathbf{x}$ to the VQ is the vector $[\sigma\ a_1\ a_2\ \cdots\ a_p]^T$ of model parameters. Each codeword is a vector $[\hat{\sigma}_i\ \hat{a}_{i,1}\ \hat{a}_{i,2}\ \cdots\ \hat{a}_{i,p}]^T$ that represents a reproduction model (4.3).

$$\hat{\sigma}_i/(1 + \hat{a}_{i,1} z^{-1} + \hat{a}_{i,2} z^{-2} \cdots + \hat{a}_{i,p} z^{-p}) = \hat{\sigma}_i/\hat{A}_i(z) \tag{4.3}$$

The distortion measure chosen for LPC vocoding is the modified Itakura-Saito distortion. It can be regarded as a measure of the dissimilarity between the power spectrum $|\sigma/A(e^{j\theta})|^2$ of the input model and the power spectrum $|\hat{\sigma}_i/\hat{A}_i(e^{j\theta})|^2$ of the reproduction model. For this case of autoregressive models, the distortion can be expressed as

$$d(\mathbf{x};\hat{\mathbf{x}}_i) = \frac{\hat{\mathbf{a}}_i^T \mathbf{R}(\mathbf{x}) \hat{\mathbf{a}}_i}{\hat{\sigma}_i^2} + \ln\hat{\sigma}_i^2 - \ln\sigma^2 - 1 \ , \tag{4.4}$$

where $\hat{\mathbf{a}}_i$ is the vector $[1\ \hat{a}_{i,1}\ \hat{a}_{i,2}\ \cdots\ \hat{a}_{i,p}]^T$ and $\mathbf{R}(\mathbf{x})$ is a $p+1$ by $p+1$ Toeplitz correlation matrix with elements $\{r_x(k-j),\ k,j = 0, 1, \cdots p\}$.

$$r_x(k) = \int_{-\pi}^{\pi} |\sigma/A(e^{j\theta})|^2 e^{jk\theta} \frac{d\theta}{2\pi} \tag{4.5}$$

Since the last two terms in (4.4) do not depend on $\hat{\mathbf{x}}_i$, they can be ignored when finding the nearest neighbor of an input vector. Thus the encoding can be performed by computing $\hat{\mathbf{a}}_i^T \mathbf{R}(\mathbf{x}) \hat{\mathbf{a}}_i / \hat{\sigma}_i^2 + \ln\hat{\sigma}_i^2$ for each $i = 1,2, \cdots N$ and picking the codeword that minimizes this quantity. This quantity can be efficiently computed in the following manner.

$$[r_x(0)r_{\hat{a}_i}(0) + 2\sum_{m=1}^{p} r_x(m)r_{\hat{a}_i}(m)]/\hat{\sigma}_i^2 + \ln\hat{\sigma}_i^2 \tag{4.6}$$

$$r_{\hat{a}_i}(k) = \sum_{m=0}^{p-k} \hat{a}_{i,m}\hat{a}_{i,m+k}$$

Since the computation of distortion between an input vector and each reproduction vector, $d(\mathbf{x};\hat{\mathbf{x}}_i)$ must be calculated often, (4.6) is used to speed up the computations. Thus the codewords are stored as the following $p+2$ scalar quantities.

$$r_{\hat{a}_i}(0)/\hat{\sigma}_i^2\ ,\ 2r_{\hat{a}_i}(1)/\hat{\sigma}_i^2\ ,\ 2r_{\hat{a}_i}(2)/\hat{\sigma}_i^2\ ,\ \ldots\ ,\ 2r_{\hat{a}_i}(p)/\hat{\sigma}_i^2\ ,\ \ln\hat{\sigma}_i^2 \tag{4.7}$$

The standard VQ algorithm proceeds by performing the following operations for every vector in the training sequence, see Fig. 4.2. First, find the codeword that is closest to each input vector and compute the average IS distortion for all of the data. Second, for all the input vectors that are encoded into a particular codeword, compute the centroid of the region and define it as the new codeword. If the decrease in distortion is above a threshold, repeat the process again on all of the training sequence. Otherwise, if the size of the codebook is below the desired number, then generate additional codewords as perturbed versions of existing codewords.

APPLICATION TO SPEECH RECOGNITION

For the speech recognition task, the VQ technique is used to cluster the LPC speech models into a few characteristic types. The use of the Itakura-Saito distortion measure provides a means to cluster observed LPC models based on the distance between their spectra. After establishing the VQ codebook on a training set, an unknown observation can be encoded so its closeness (distortion) to each codeword can be determined.

The LPC models used in this study are parameterized by reflection coefficients rather that predictor coefficients as in (4.2). The recursive lattice estimation technique was used to determine a new LPC model for every speech sample rather than the common approach of once every 128 to 256 samples. The efficient computation of the IS distortion measure (4.6) uses the speech correlation function. The reflection coefficients can be transformed into a normalized correlation sequence, instead of using (4.5).

The process of encoding a speech sequence once codewords have been established proceeds in the following manner.

(i) Recursive lattice algorithm is applied to the speech sequence. Each speech sample generates a set of reflection coefficients, $\{k_1,k_2,...,k_{10}\}$.

(ii) The reflection coefficients are transformed into the normalized correlations $\{r_s(1),...,r_s(10)\}$.

(iii) Calculate $\sigma^2 = \prod_{i=1}^{10}(1-k_i^2)$.

(iv) The input vector $\mathbf{x}$ becomes $\{\ln\sigma^2,1,r_s(1),...,r_s(10)\}$.

(v) For each codeword, the distortion (4.6) is computed using $\mathbf{x}$ and the codeword description (4.7).

(vi) The codeword with the lowest distortion is the reproduction vector $\hat{\mathbf{x}}_i$ and is associated with that speech sample.

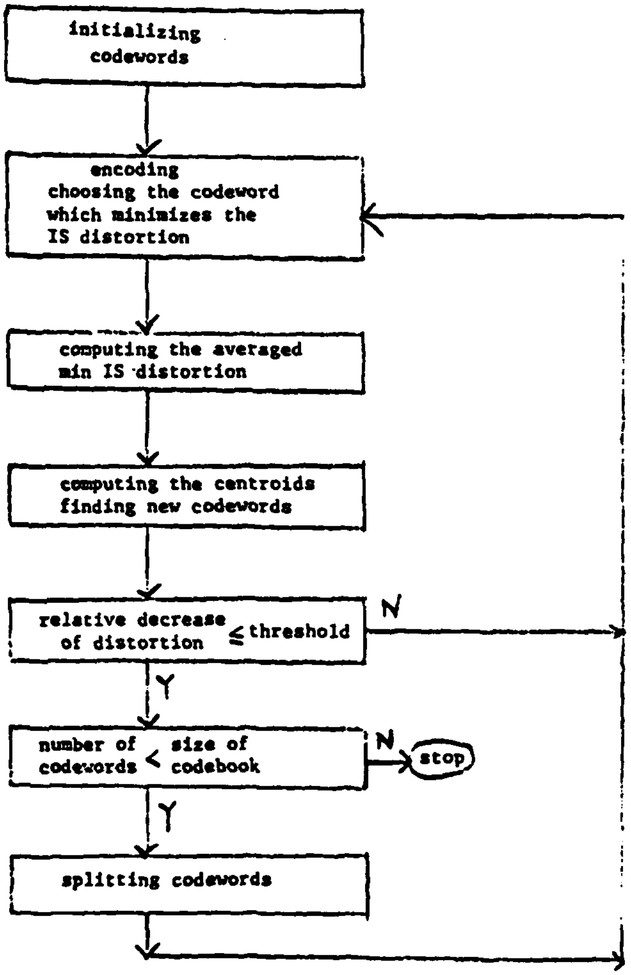

initializing codewords
encoding choosing the codeword which minimizes the IS distortion
computing the averaged min IS distortion
computing the centroids finding new codewords
relative decrease of distortion ≤ threshold
N
Y
number of codewords < size of codebook
N
stop
Y
splitting codewords

Figure 4.2

4.4 ANALYSIS OF ENTIRE WORDS

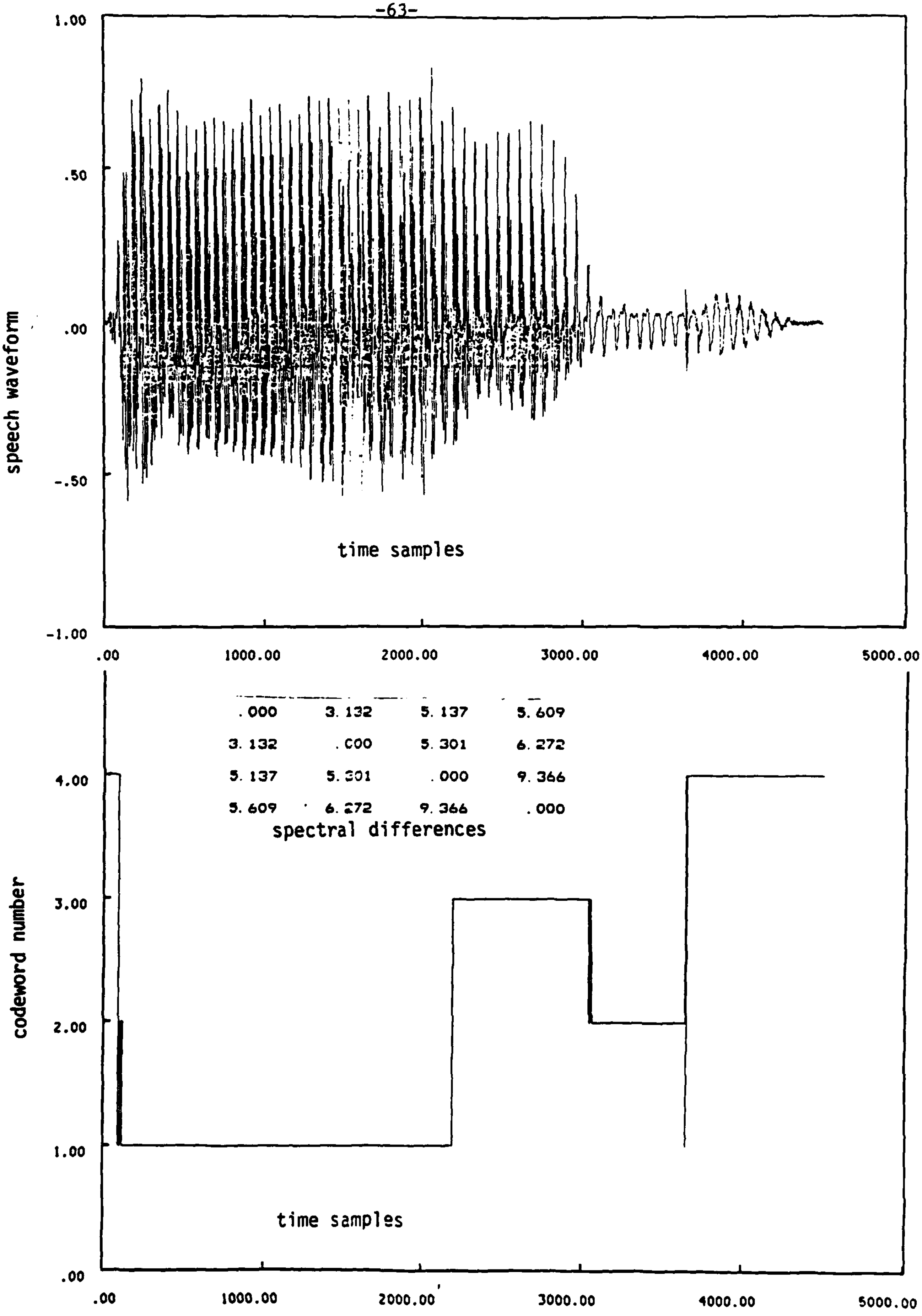
In order to better understand the effect of the lattice VQ on spoken words, our studies began by examining entire words. The square root normalized recursive least square lattice algorithm was applied to the speech signal. A short time constant, λ=159/160 was used to track the fast variations in the speech waveform, particularly during the stop consonant portion. A set of ten reflection coefficients were determined for every speech sample. The reflection coefficients were transformed into normalized correlation coefficients of order ten so that the standard VQ algorithm could be used to obtain the codewords. The results of studying the two words 'bad' and 'bat' are presented in this section. Each word was sampled at 8 Khz. and converted to a 12 bit integer. The duration of each word was more than 4500 samples so that more than 4500 vectors were used in the determination of the codebook. This is contrary to the standard LPC method of determining a single speech model vector for blocks of 128 to 256 speech samples.

The standard VQ approach uses the Itakura-Saito distortion measure to indicate how well the codewords fit the input data. Another distortion measure was used to compare codewords. The difference between the log of the spectra associated with the codewords was computed, called the spectral difference measure. The limit of perceptual difference in two (autoregressive) spectra was determined for subjective studies to be 2 db spectral difference.

The standard VQ algorithm was used to find codebooks of size four and eight for the entire word, 'bad' and 'bat'. When four codewords were used, the word 'bad' was encoded into these codewords as shown in Fig. 4.3. This figure shows which codeword was chosen (vertical axis) for each time sample (horizontal axis). From Fig. 4.3 and 4.4, the speech waveform and the VQ partition can be compared. Generally, there were two codewords for the vowel /a/, the other two representing the other parts of the words. No codeword was determined that represented the stop consonant /b/. Here the codewords could not be used to distinguish the silence, the first stop consonant, the vowel, or the final consonant. For these four codewords, the Itakura-Saito distortions were .165 and .170, respectively and the difference between codewords are all greater than 3 db, so the four codewords are distinct.

When eight codewords were determined, the vowel part was more accurately determined but again a clear identification of the silence and the consonant were not make. The distributions of codewords, Fig. 4.5 and 4.6 show that three codewords represented the different stages of the vowel. In Fig. 4.5 for the words 'bad', codewords one and five are used alternatively during the vowel. This happens because these codewords are only 2.1 db spectral difference apart and hence not perceptually distinguishable entities. Similarly codewords three and seven are 2.3 db apart. Thus although the IS distortion for 'bad' has dropped to .099 for eight codewords from .165, the additional codewords try to refined the specification of the vowel rather than distinguish other parts of the words. The spectral differences between the codewords is given in Table 1.

From the above experiments, we could not determine the codewords for the various parts of the words. Therefore, the steady state vowel part was extracted from each word and studied separately.

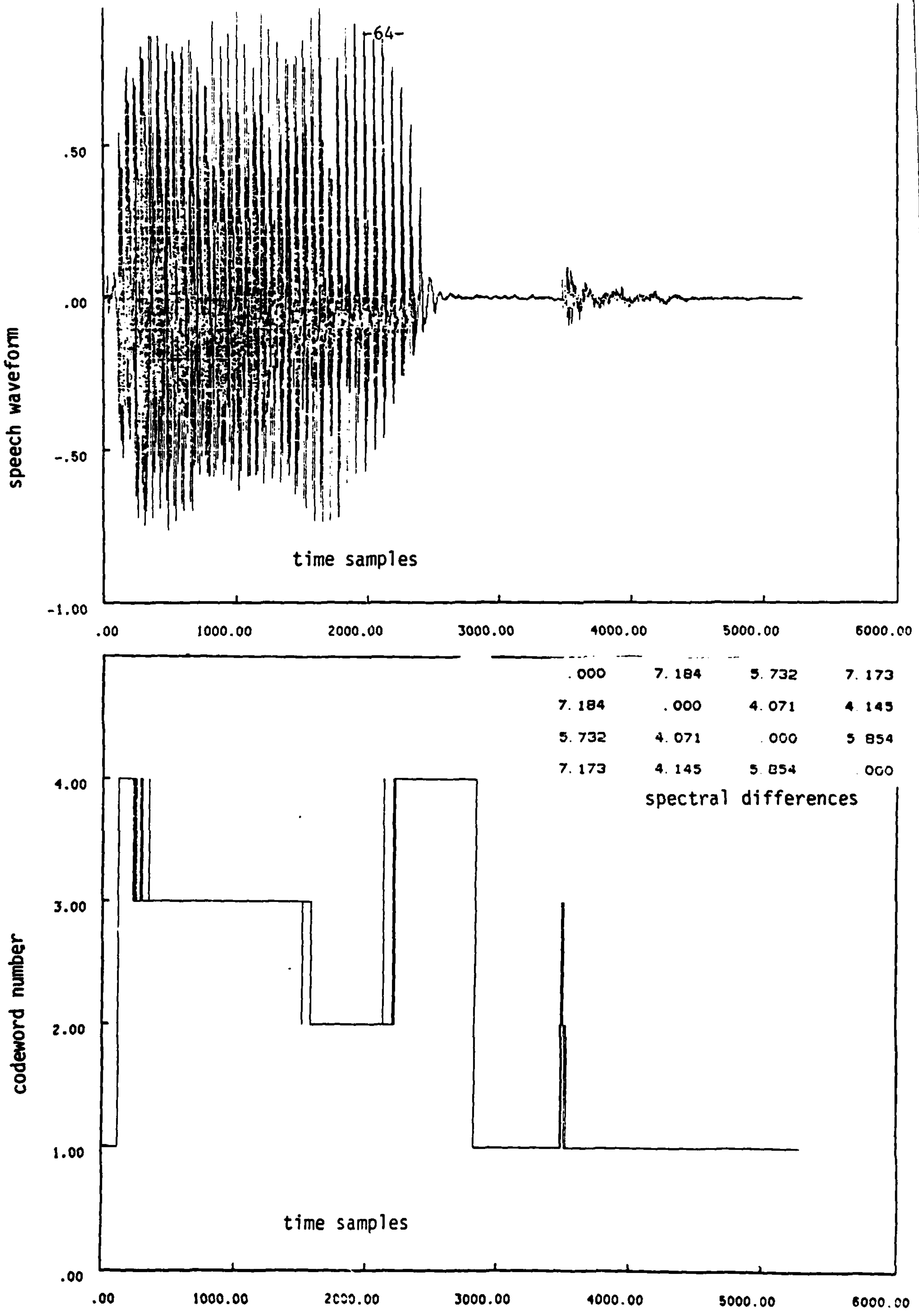


.000	3.132	5.137	5.609
3.132	.C00	5.301	6.272
5.137	5.301	.000	9.366
5.609	6.272	9.366	.000

spectral differences

standard VQ codewords for 'bad' Figure 4.3

IS = .1647

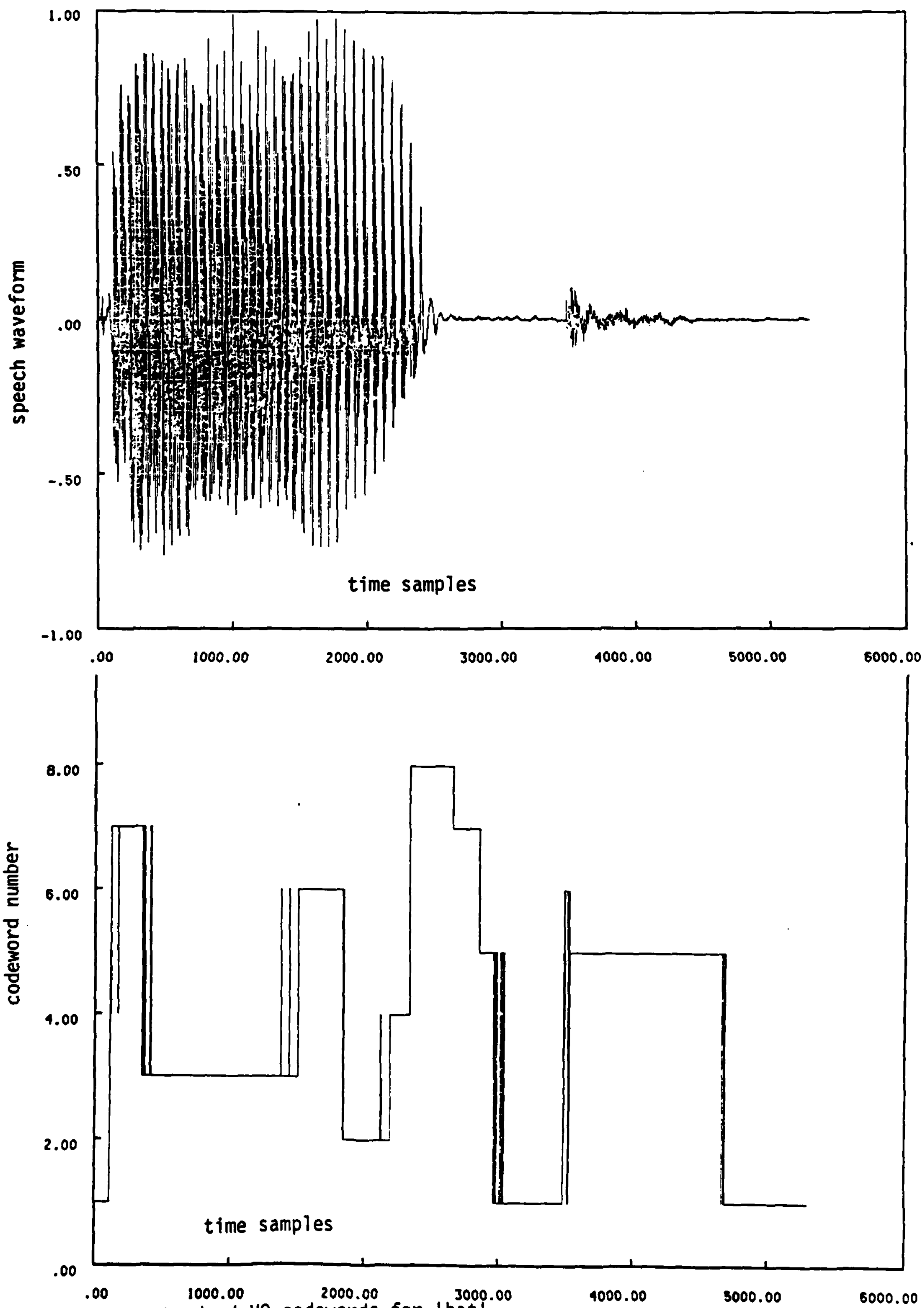


.000	7.184	5.732	7.173
7.184	.000	4.071	4.145
5.732	4.071	.000	5.854
7.173	4.145	5.854	.000

spectral differences

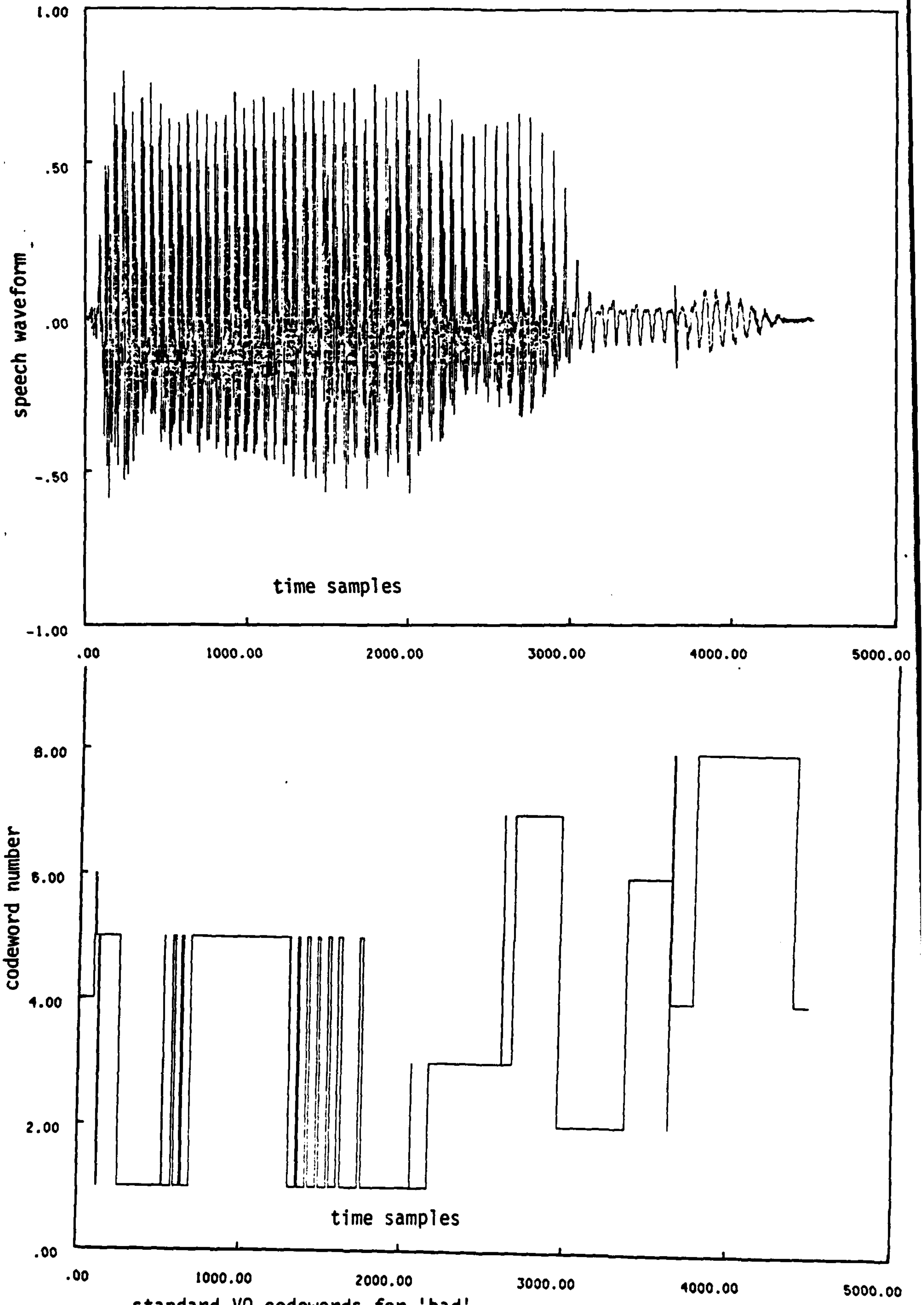
standard VQ codewords for 'bat'
IS = .1698

Figure 4.4



standard VQ codewords for 'bat'
IS = .1024

Figure 4.5



standard VQ codewords for 'bad'
IS = .0994

Figure 4.6

.000	3.977	4.385	6.770	2.153	3.880	5.347	5.500
3.977	.000	3.649	8.047	3.645	5.261	3.932	6.891
4.385	3.649	.000	9.563	5.322	7.423	2.353	8.331
6.770	8.047	9.563	.000	6.015	6.559	10.319	3.376
2.153	3.645	5.322	6.015	.000	3.453	5.823	5.403
3.880	5.261	7.423	6.559	3.453	.000	7.954	4.806
5.347	3.932	2.353	10.319	5.823	7.954	.000	9.277
5.500	6.891	8.331	3.376	5.403	4.806	9.277	.000

spectral differences (db) of 8 codewords for /bad/

.000	8.381	5.954	9.669	3.591	7.109	6.439	10.039
8.381	.000	5.196	2.609	7.264	3.008	4.150	4.317
5.954	5.196	.000	6.804	5.908	3.142	4.018	7.722
9.669	2.609	6.804	.000	7.773	4.933	4.357	1.966
3.591	7.264	5.908	7.773	.000	6.740	4.700	7.661
7.109	3.008	3.142	4.933	6.740	.000	4.356	6.252
6.439	4.150	4.018	4.357	4.700	4.356	.000	4.533
10.039	4.317	7.722	1.966	7.661	6.252	4.533	.000

spectral differences (db) of 8 codewords for /bat/

Table 1

4.5 ANALYSIS OF VOWELS

Since the vowels dominated the previous experiments, the steady state vowel parts of several different words were studied to find the general codewords representing the vowels. From the words, 'bad', 'bat' and 'gat' the steady state vowel portions were extracted for a training sequence to generate a codebook for the vowel /a/. Similarly the steady state parts in the words 'bid', 'bit', 'did', 'dip' was used for the vowel /i/ and 'boast', 'bowl', 'dole', and 'ghost' was used for the vowel /o/. When only one codewords was determined for each vowel, the codewords were surprisingly similar. Table 2 shows that the codewords for different vowels differ between 3 and 4.4 db. For words containing the same vowel, the codewords for the same vowel sometimes differed as much as the difference between /a/, /o/ and /i/ in Table 2.

TABLE 2: Spectral difference between vowels

Codeword	/a/	/o/	/i/
/a/	0.00	4.39	3.29
/o/	4.39	0.00	3.00
/i/	3.29	3.00	0.00

When four codewords were used for the steady state part of the vowels, the codewords for the same vowel in different words were often different, see Fig. 4.7, 4.8 and 4.9. Often a vowel was split into two codewords, one for the beginning and another for the end. For /i/, the beginning of the vowel is represented by codewords 3 and 4, and the end of the vowel is codewords 1 and 2. Some of the four codewords were quite similar, for example in /o/ codewords 1 and 2 and codewords 2 and 4 are less than 2 db apart, see Fig. 4.9. When eight codewords were used for the /a/ vowel, many of them were very similar, see Table 3, therefore it is appropriate to use four codewords to represent different stages of the same vowel.

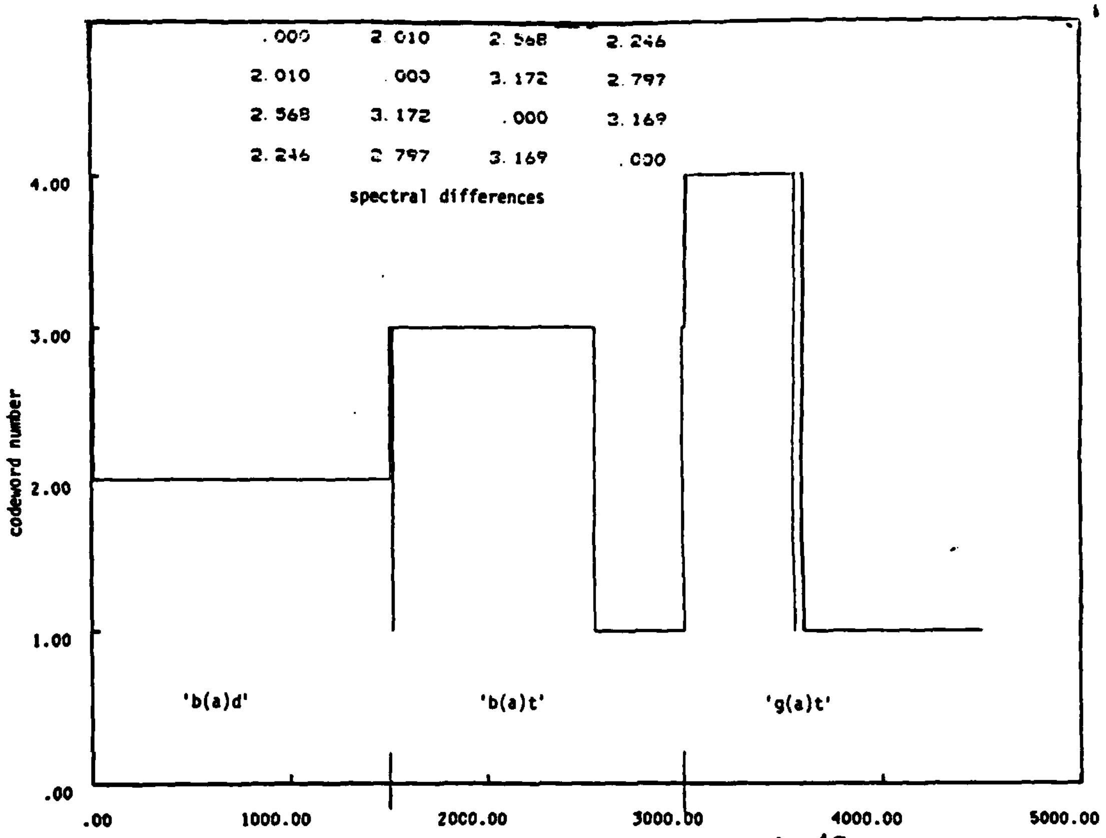


Fig.4.7

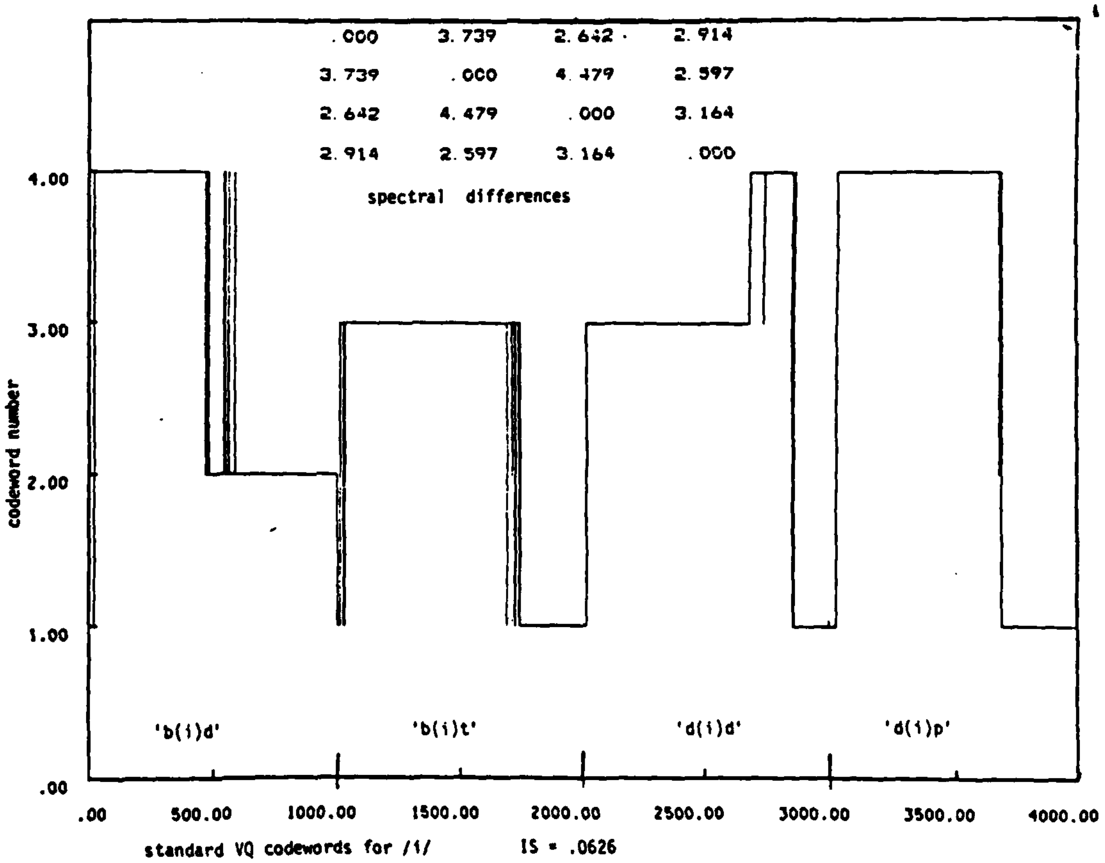
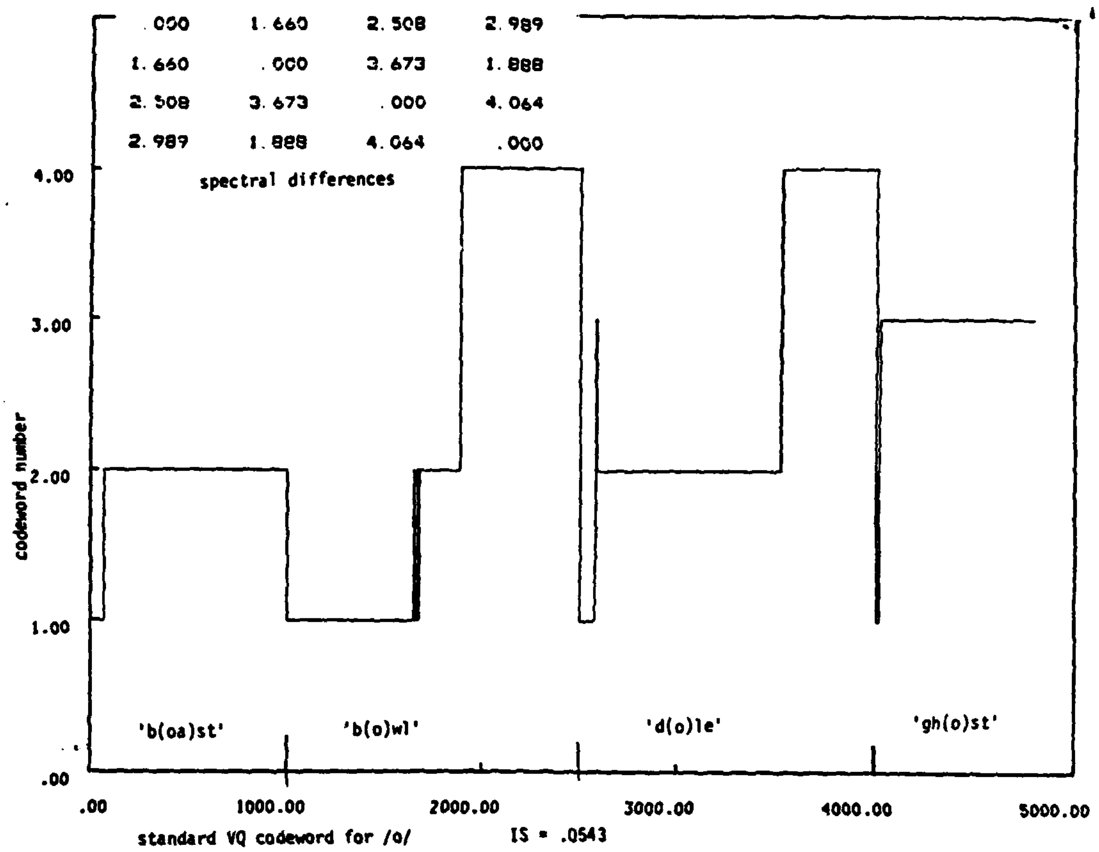


Figure 4.8



.000	1.660	2.508	2.989
1.660	.000	3.673	1.888
2.508	3.673	.000	4.064
2.989	1.888	4.064	.000

Figure 4.9

.000	2.369	2.853	2.468	1.718	2.078	2.638	3.424
2.369	.000	3.743	3.458	2.795	1.607	3.583	3.565
2.853	3.743	.000	3.310	2.557	3.085	1.039	3.450
2.468	3.458	3.310	.000	1.610	2.643	3.178	2.783
1.718	2.795	2.557	1.610	.000	2.304	2.707	2.902
2.078	1.607	3.035	2.643	2.304	.000	2.839	3.138
2.638	3.583	1.039	3.178	2.707	2.839	.000	3.585
3.424	3.565	3.450	2.783	2.902	3.138	3.585	.000

spectral differences (db) of 8 codewords for /a/

Table 3

These codebooks for the respective vowels were tested to see if they could distinguish the correct vowel. When the training sequences of the vowels was encode by each codebooks, a IS distortion was determined, see Table 4. The IS distortion for a vowel codebook on the wrong vowel was at least four times higher than for the correct vowel. Therefore, it is not very difficult to distinguish the vowel in each word using the standard VQ technique.

TABLE 4: Itakura-Saito distortion between vowels

Vowel	/a/	/o/	/i/
codebook /a/	.058	.460	.340
codebook /o/	.418	.062	.490
codebook /i/	.365	.447	.082

4.6 MODIFIED VQ WITH TRAJECTORY INFORMATION

From the previous results, the vowels are not hard to distinguish because they are relatively stationary and of long duration. However, the stop consonants (like 'b', 'd' and 'g') are transient in nature and are of short duration, typically less than 20 ms. (160 samples). Using a LPC based VQ system that determines speech model parameters every 128 (to 256) samples would not yield enough information to identify these very short sounds. From studies in acoustic phonetics, it is known that the formant trajectories of these consonants follow different paths. If the parameterization used to represent speech sounds included information about formant trajectories, these transitional sounds would be easier to identify. As seen in Section 3, the trajectories of the reflection coefficients were different for the beginnings of the words 'did' and 'bid'. A steep change occurred during the initial consonant while during the steady state vowel, very slowly changing coefficients resulted. By incorporating this trajectory information in the speech parameterization, recognition of transitional sounds should be improved.

The trajectory of a reflection coefficient was determined as a smoothed derivative. During the vowels, the reflection coefficients had a ripple due to the pitch period. This ripple in the otherwise steady reflection coefficient values had an undesirable influence in the modified VQ approach. Thus a linear approximation over 15 sample points to the derivative of the reflection coefficients was used for the trajectory information. The fluctuations due to the influence of the pitch were smoothed out. The trajectories of the first and second order reflection coefficients, denoted Δk_i appeared to be the most indicative of changing signal characteristics so they were included in the modified VQ technique. The standard VQ algorithm of Section 3 was modified so that the codewords consist of two parts; the original correlation coefficients and the trajectory of the reflection coefficients. The distortion measure used for the spectrum part in the modified VQ was still the IS distortion. The Euclidean norm was used as the distortion measure for the two reflection coefficient trajectories. The total distortion was the weighted sum of the IS distortion and the Euclidean norm of the trajectories. The centroid (dk_i) was calculated as the averages of the reflection coefficient trajectories. A weighting factor for the Euclidean norm was used to balance the two distortion measures. This factor is the ratio of the minimum IS distortion (M) to twice the variance of the reflection coefficient trajectories (D).

MODIFIED VQ ALGORITHM

N = *number of the input samples*

$\Delta k_i(n)$ = *reflection coefficient trajectory at sample n*

dk_i = *codeword* for *reflection coefficient trajectories*

IS = *Itakura-Saito distortion*

M = *minimum IS distortion*

D_i = *variance of reflection coefficient trajectories*

INITIALIZING: $dk_1 = 0 \quad dk_2 = 0 \quad M = 0$

$$D_i = \frac{1}{N}\sum_{n=1}^{N}\Delta k_i^2(n) \quad i=1,2$$

ENCODING: *choose the codeword that minimizes the total distortion*

$$\textit{total distortion} = IS + \frac{M}{2D_1}(\Delta k_1 - dk_1)^2 + \frac{M}{2D_2}(\Delta k_2 - dk_2)^2$$

UPDATING:

$$D_i = \frac{1}{N}\sum(\Delta k_i^2 - dk_i^2)$$

M = min *IS distortion*

$$\textit{avgdist}(dk_i) = \frac{M}{D_i}\frac{1}{N}\sum(\Delta k_i - dk_i)^2$$

$$\textit{total distortion} = \text{min } IS + \frac{1}{2}\textit{avgdist}(dk_1) + \frac{1}{2}\textit{avgdist}(dk_2)$$

NEW CODEWORDS: *compute the centroids of the standard VQ parameters* and Δk_i

TESTING: if *relative decrease of distortion* $\leq$ *threshold* : *go to splitting*
else : *go to encoding*

SPLITTING: if *number of codewords* = *size of codebook* : *stop*
else : *split codewords*
go to encoding

The modified VQ approach was first applied to simulated data that represent the ideal acoustical models of the stop consonants. The simulation of the sound 'ba' had two poles (750 Hz and 1600 Hz) for the steady state vowel while the first formant went from 200 Hz to 750 Hz and the second formant went from 1400 Hz to 1600 Hz in the first 20 ms (160 samples) of the transitional part. The reflection coefficient trajectories were approximately constant in the transition region and zero in the steady state region. The reflection coefficients of fourth order were computed from the simulated data. When the size of the codebook was two, the result turned out to be perfect, the two partitions were exactly the transitional part and the steady state part. Next the considerably more difficult problem of real speech data was studied using this modified VQ approach.

In order to find the codeword for the consonant 'b', the beginnings of three words which start with 'ba' ('bad', 'bat' and 'bank') was cascaded and used to generate the codebooks of size eight of both modified and standard VQ (Table 5 and 6). The IS distortions were very close in these two codebooks. The distributions of the modified codewords are in Fig. 4.10. Basically, one codeword was used for the silence (codeword 3) and one for the transition (codeword 8). The other six codewords represented the vowel. A consistent pattern of change from the codeword for silence (3) to the same codeword (8) occurred at the transition time in all of these three words. This effect was not seen in the standard VQ (Fig. 4.11). Instead, at the beginning of each word, several codewords were used before reaching the vowel. It appeared that there were too many codewords for the steady state parts, so the size of both codebooks was reduced to four. Surprisingly, the difference of the vowel /a/ in different words was so important that three different codewords were used for the same vowel in three different words. The other one represented the transitional parts while the leading silence was encoded as a vowel.

Going through exactly the same procedures but using 'gab', 'gaff' and 'gat' for 'ga', different types of problems were encountered. In the case of eight modified codewords, there was one for the silence and still too many for the vowel, but it was very 'unstable' at transient time. This did not happen in the standard VQ. But the standard VQ mapped most of the beginning of 'b' into the silence. If four modified codewords were used, there was one for silence (codeword 1) and the same two codewords (2,4) representing two stages of the vowel in three different words as in Fig. 4.12. This was better than that in 'ba'. But, it still alternated between two codewords (1,3) at transient time again. The unstability persisted in the modified VQ but not in the standard VQ (Fig. 4.13). Comparing Fig. 4.12 and 4.13, if the effects of the reflection coefficient trajectories is included, the transition can be detected earlier at the beginning of each word. However, the standard VQ assigned all the samples of 'b' to the codeword for silence (1).

The differences of the same vowel in different words were very large so that many codewords were used to represent the same vowel. To find the typical codeword for the stop consonants, the strong influence of the following vowel had to be diminished. This lead to a classified VQ algorithm where the number of codewords used for vowels was restricted so more codewords would be determined for the transitional parts.

.686177	-.262159	.279971	-.378734
-.123740	-.177142	.446814	.433686
.309272e-01	-.247705		
-.425619r-04	.250907e-03		
.825386	-.635448	-.152013	-.322850
-.173931	-.124693	.501873	.378517e-01
.642003e-02	.134824e-01		
-.665252e-04	-.983339e-04		
.549098	.205957	.128860	.185963e-01
-.615557e-01	-.715450e-01	-.422913e-01	-.303090e-01
.337069e-01	.369361e-01		
.783153e-02	.155118e-02		
.865827	-.626488	-.694115e-01	-.363710
-.280562e-01	-.295255	.182878	.330512
.873702e-01	-.123212		
-.198653e-03	.334887e-03		
.745936	-.389708	.157856	-.375582
-.143735	-.144050	.442576	.357337
.532621e-01	-.276019		
-.208266e-03	.372548e-03		
.881377	-.783747	-.224615	-.185121
.135996	.299462e-01	.357597	.100329
-.591172e-01	-.452891e-02		
-.342635e-04	.696324e-04		
.798454	-.434789	-.258210	-.345911
-.200801	-.276570	.196048	.324956
.152234	.164580e-02		
-.143626e-02	.124287e-02		
.919844	-.745508	-.120656	-.946805e-01
.139044	.829787e-01	.106712	.111964
.153276	-.637693e-01		
.495182e-03	-.602024e-02		

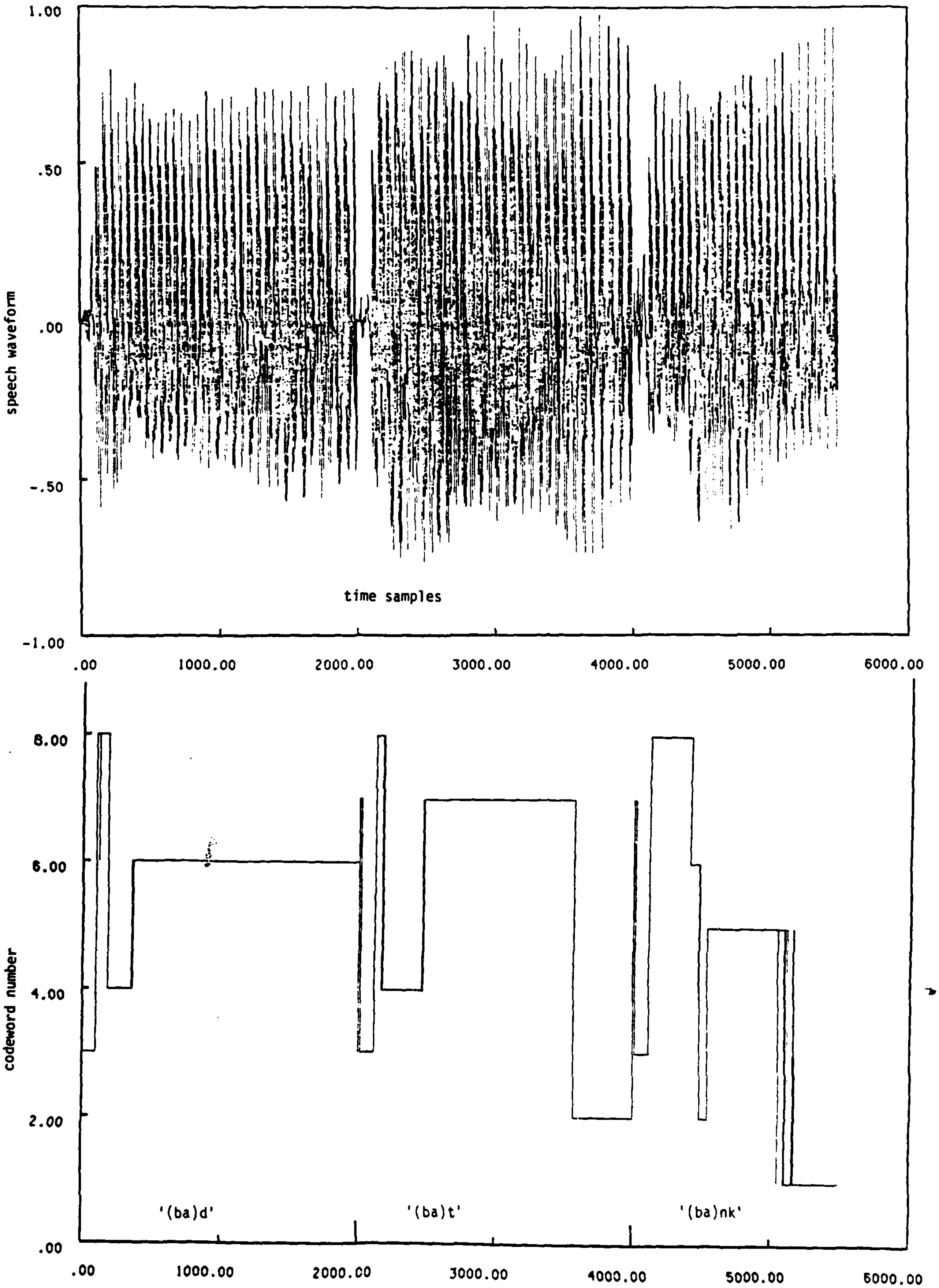
Modified VQ codewords for /ba/
(last two entries are trajectories of reflection coefficients)

Table 5

.302182	.154637	.119021	.671577e-01
.235479e-01	-.610938e-02	-.876120e-02	-.140060e-01
-.249855e-02	-.395959e-02		
.825190	-.634524	-.149558	-.323153
-.173634	-.123710	.502010	.385303e-01
.766131e-02	.106940e-01		
.799472	-.437694	-.258345	-.348869
-.198813	-.270308	.210387	.319895
.154158	.101171e-03		
.873909	-.656325	-.648970e-01	-.357912
.212173e-02	-.274094	.187845	.333131
.765464e-01	-.133481		
.723796	-.335170	.202954	-.385472
-.133175	-.154734	.441685	.385229
.460555e-01	-.249998		
.881331	-.783175	-.224739	-.184928
.136005	.300092e-01	.356547	.100334
-.591125e-01	-.386784e-02		
.834675	-.417306	-.793471e-01	-.287117
-.104307	-.182997	.166795	.348844
.187232	-.339953e-01		
.920311	-.768690	-.930438e-01	-.923088e-01
.155437	.677066e-01	.105970	.117992
.159892	-.822916e-01		

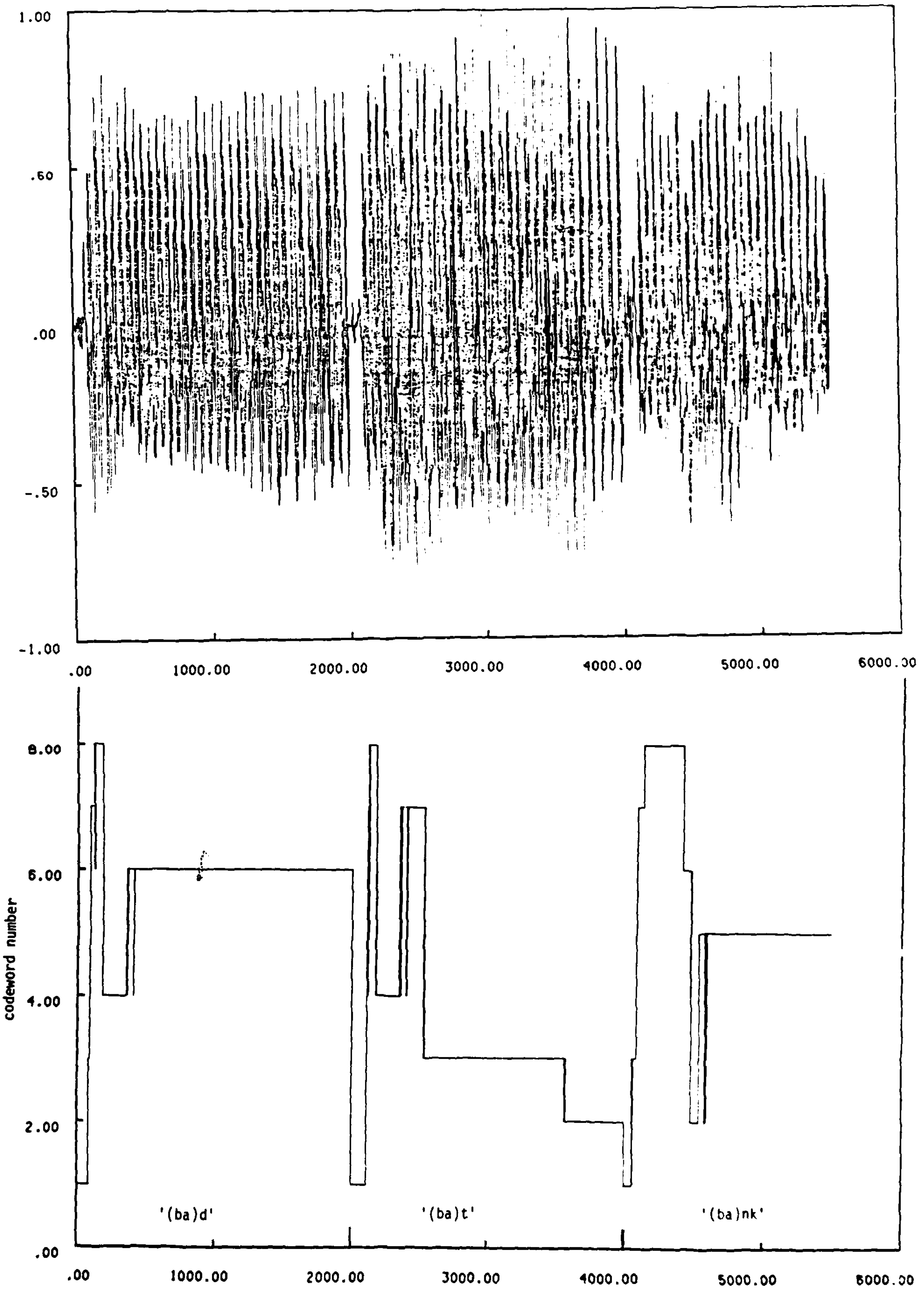
Standard VQ codewords for /ba/

Table 6



Modified VQ codewords for /ba/ IS = .0532

Figure 4.10



Standard VQ codewords for /ba/ IS = .0519 Figure 4.11

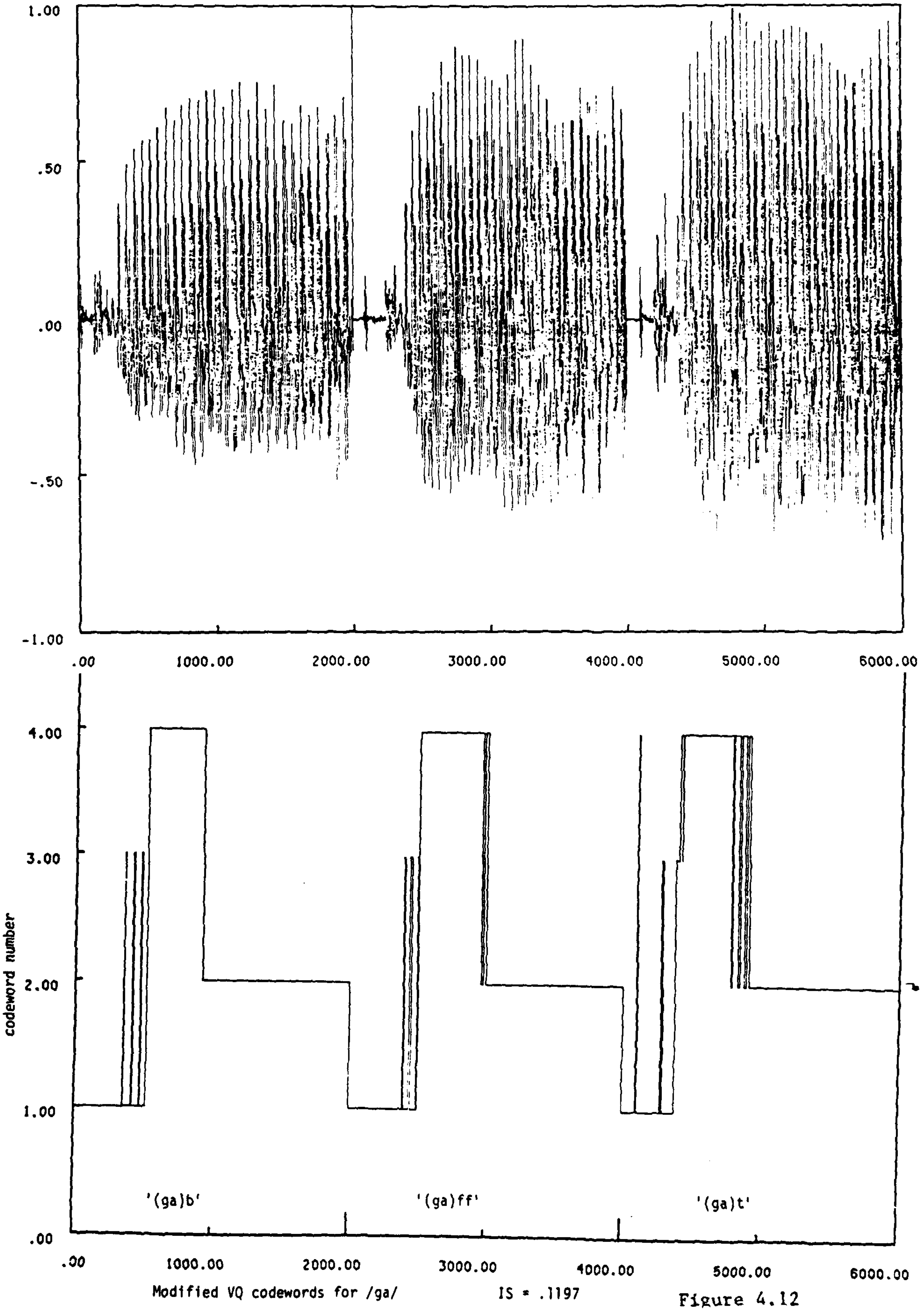

1.00
.50
.00
-.50
-1.00
.00
1000.00
2000.00
3000.00
4000.00
5000.00
6000.00
4.00
3.00
2.00
1.00
.00
codeword number
'(ga)b'
'(ga)ff'
'(ga)t'
.00
1000.00
2000.00
3000.00
4000.00
5000.00
6000.00
Modified VQ codewords for /ga/
IS = .1197

Figure 4.12

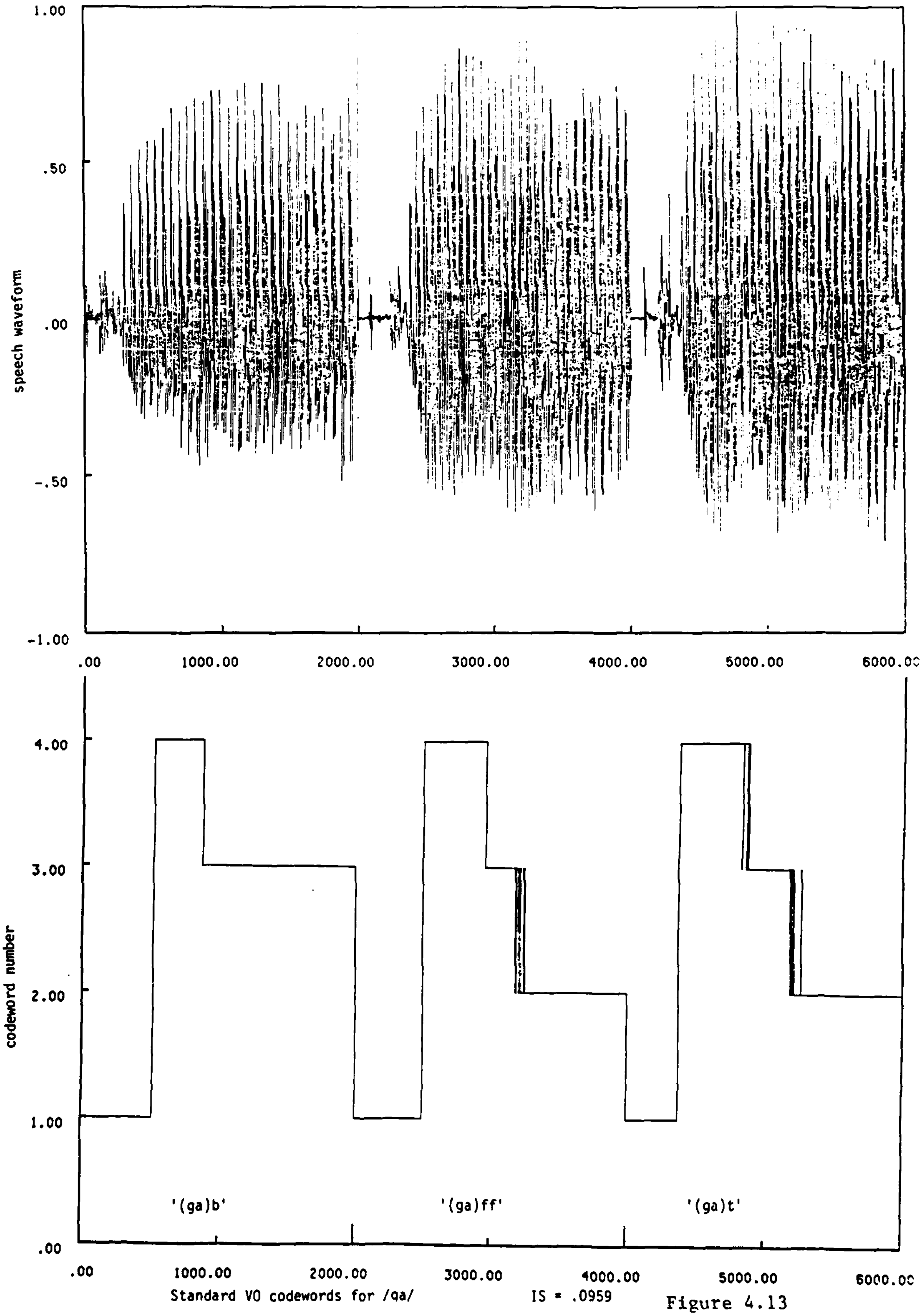

1.00
.50
.00
-.50
-1.00
speech waveform
.00
1000.00
2000.00
3000.00
4000.00
5000.00
6000.00
4.00
3.00
2.00
1.00
.00
codeword number
'(ga)b'
'(ga)ff'
'(ga)t'
.00
1000.00
2000.00
3000.00
4000.00
5000.00
6000.00
Standard VO codewords for /ga/
IS = .0959

Figure 4.13

4.7 CLASSIFIED VQ

A classified VQ design allows a time varying signals to be divided into different components where each component is quantized to a desired accuracy. When VQ is applied to a spoken word, the codewords represent primarily the vowel sounds since they are the longest and most stationary sounds, see Section 4.4. When only vowels are quantized, there is a difference between repetitions of the same vowel in difference words. This difference can be similar to the difference between nonvowel sounds and vowels. Since the stop consonants are short transitional sounds, codewords must be explicitly allocated to represent them if they are to be identified. The classified VQ approach separates consonant-vowel words into a few codewords for the vowel and a few codewords for the silence, consonant and vowel transition.

The classification procedure uses codewords determined for a steady state vowel to separate a word into a 'vowel' part and a 'transitional' part. This 'transitional' part is used to defined a codebook that can identify the stop consonant, see Fig. 4.14. Four codewords were determined for the steady state part of a vowel (using different words) as in Section 4.5. Then, the training sequences of similar consonant-vowel words were encoded by that vowel codebook to find the best codeword for each speech sample. If the distortion was below a certain threshold, the sample was assigned to that codeword. If the distortion was above the threshold, the sample was put in a 'transitional' group. After this classifying procedure, the 'transitional' group contained the samples for silence, stop consonants and the beginnings of the vowel. A few sample points of the steady state vowel part were occasionally included. A codebook for the transitional part was designed so that four codewords could be forced for these transient sounds.

The steady state vowel parts of six words ('bad', 'bat', 'dab', 'gab', 'gaff' and 'gat') were combined as the training sequence to design a codebook of size four for the vowel /a/ using the standard VQ algorithm. The threshold for accepting each codeword was twice the average distortion of that codeword. The training sequence of 'ba' was classified in this way where those samples assigned to the fifth codeword are in the 'transitional' group, see Fig. 4.15. For 'bad' and 'bat' only, this group consisted of the first 500 samples from each word and very few of the vowel.

But almost all the samples in 'bank' belonged to this 'transitional' group. The nasal consonant 'n' affects the vowel so that the steady state part of 'bank' was quite different from that in all the other words, i.e. 'an' is different from 'a'. Therefore, only the beginning of the words 'bad' and 'bat' were used in the following study. Four codewords were designed for the transitional parts of these two words using the standard and modified VQ algorithm, Fig. 4.16. Using the standard VQ, there is one codeword for the vowel /a/ (2), one for the silence (1), one for the transition (3) and the other one was between transition and steady state (4). The codeword for the vowel (2) in this codebook was similar to one of the codewords in the codebook of /a/. The distributions of those codewords using the modified VQ still has one for the silence (1), one for the transient part (4) and one for the vowel (2). Codeword 3 represents a very few samples between the silent part and transient part. In the very beginning of these words, the codeword for silence (1) and codeword (3) alternate. This is natural occurrence since there is no definitive boundary between silence and the stop consonant. Comparing the standard and modified VQ (Fig. 4.16), at the beginning of each word the first sample not encoded into silence occurs earlier in the modified VQ method. The modified algorithm can detect the transition from silence to consonant earlier than the standard VQ.

A similar approach was applied to the training sequence of 'ga' (from 'gab', 'gaff' and 'gat'). There were about 3000 samples in the 'transitional' group as in Fig. 4.17. Besides the beginnings of those three words, some samples from the steady state part of 'gab' were included. These samples were used to design a codebook of size four. For the standard VQ , there was a codeword for silence, for the transition, for the vowel, and one for these samples from the middle of the vowel part of 'gab'. Codewords for the vowel were quite similar to those in the codebook for the steady state /a/. For the modified VQ, the beginning of 'gat' was very different from those of 'gab' and 'gaff'. The codewords represent the silence, the transition, the vowel, and a codeword for the beginning of 'gat'.

When the beginnings of 'dab' and 'dan' were encoded by the vowel codebook, the effect of the nasalized vowel was seen again, see Fig. 4.18. The final part of the vowel in 'dan' was influence by the following nasal consonant 'n' such that the distortion was more than twice the expected distortion for that codeword. Therefore the end of the vowel in 'dan' was not included in the 'transitional' group. In this case, the distributions of codewords in both the standard and modified VQ were exactly the same. The smoothed derivatives of reflection coefficients did not make any differences in this case.

The same experiments were repeated for different vowels; 'bo' (from 'boast', 'bone' and 'bowl'), 'do' (from 'dole', 'dough' and 'doze') and 'go' (from 'ghost', 'goat' and 'go'). The distributions of the codewords for 'd' and 'g' in the standard and modified VQ were very similar. But for 'b', they were significantly different at the very beginnings of each word.

In general, the classified VQ enabled us to have more codewords for the transitional parts. Also, in some cases, the codebooks including the reflection coefficient trajectories could detect the transient parts better than using the standard VQ only. Since these results were quite promising, a test of recognizing stop consonants could be performed.

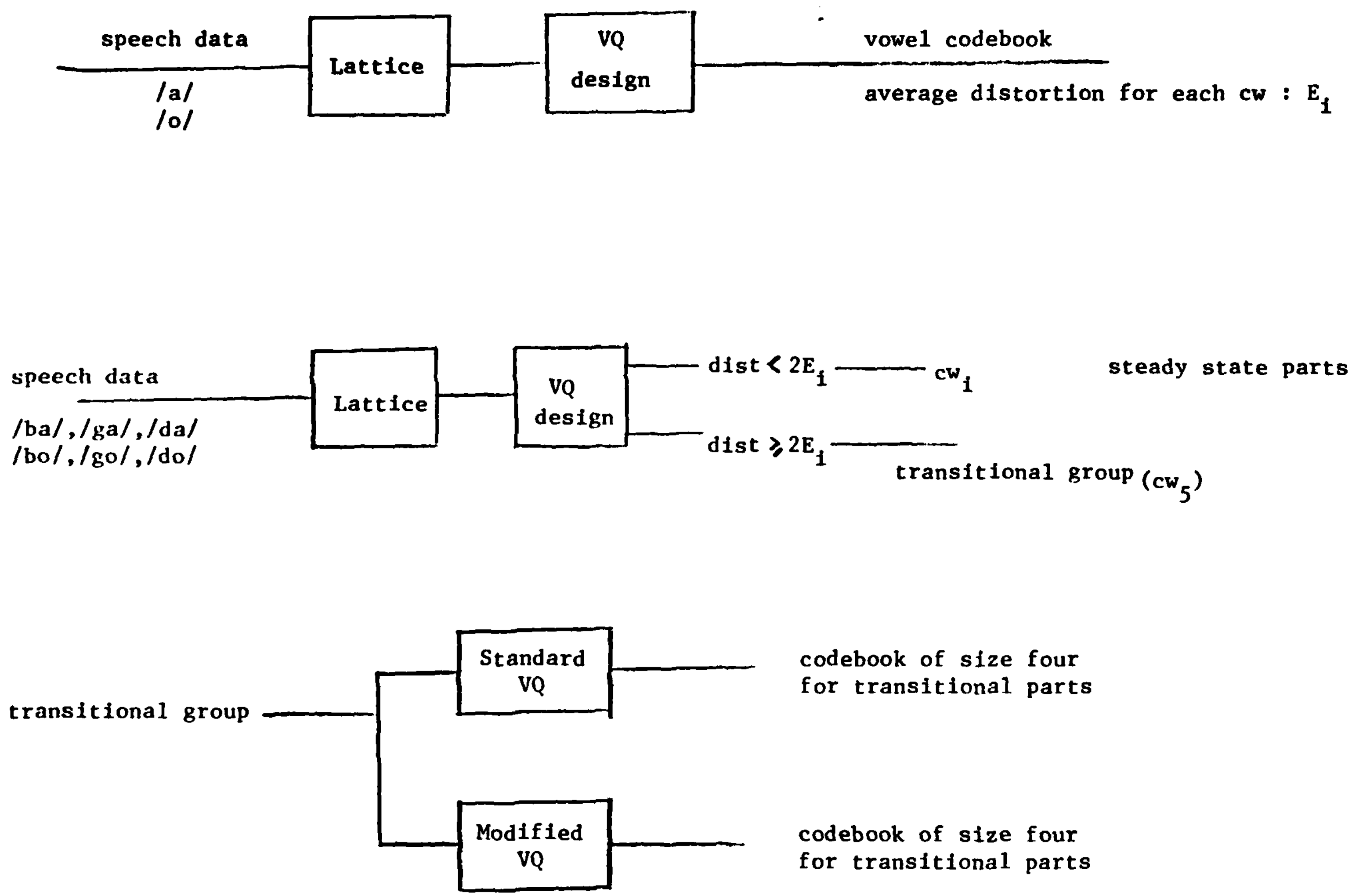
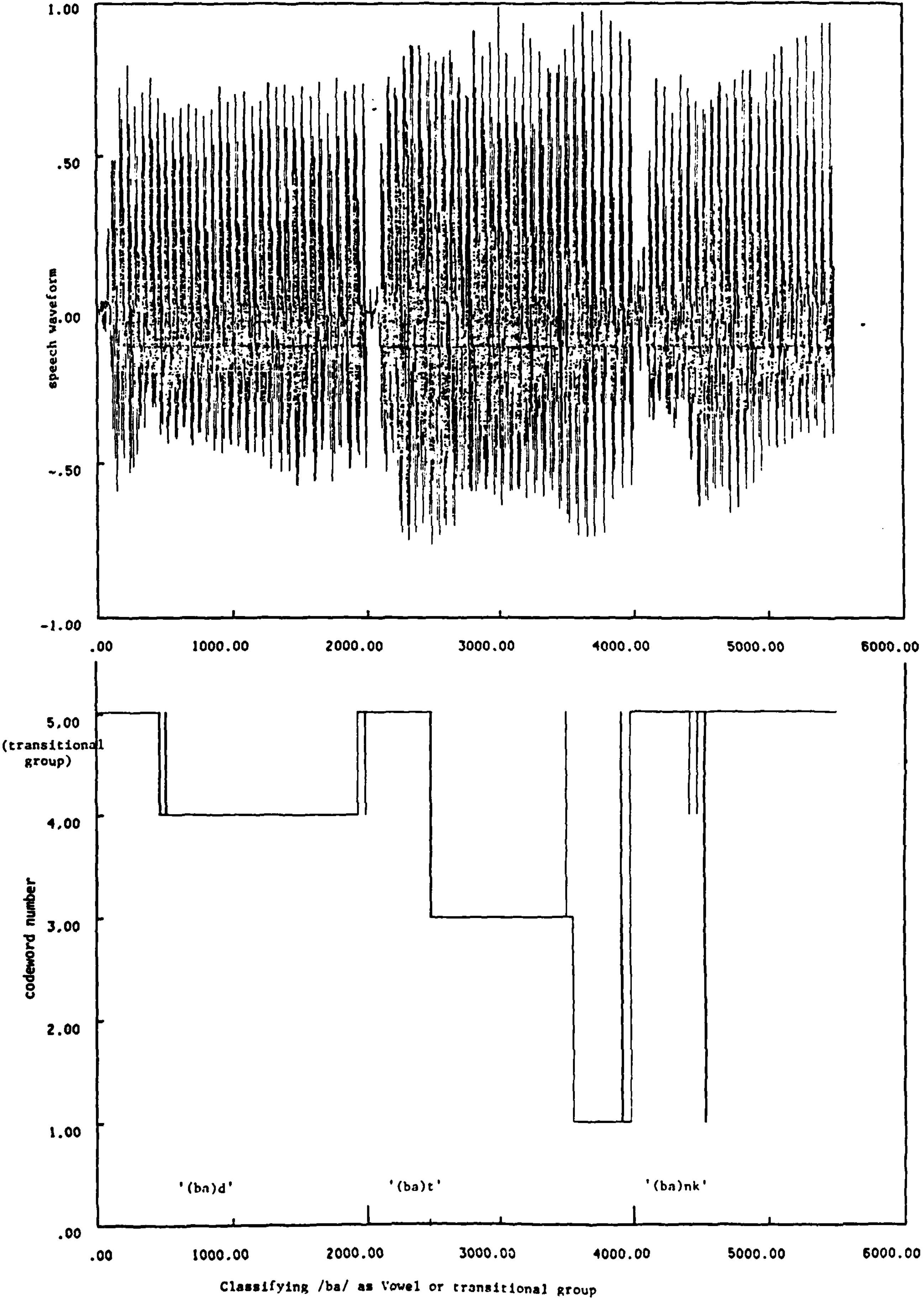
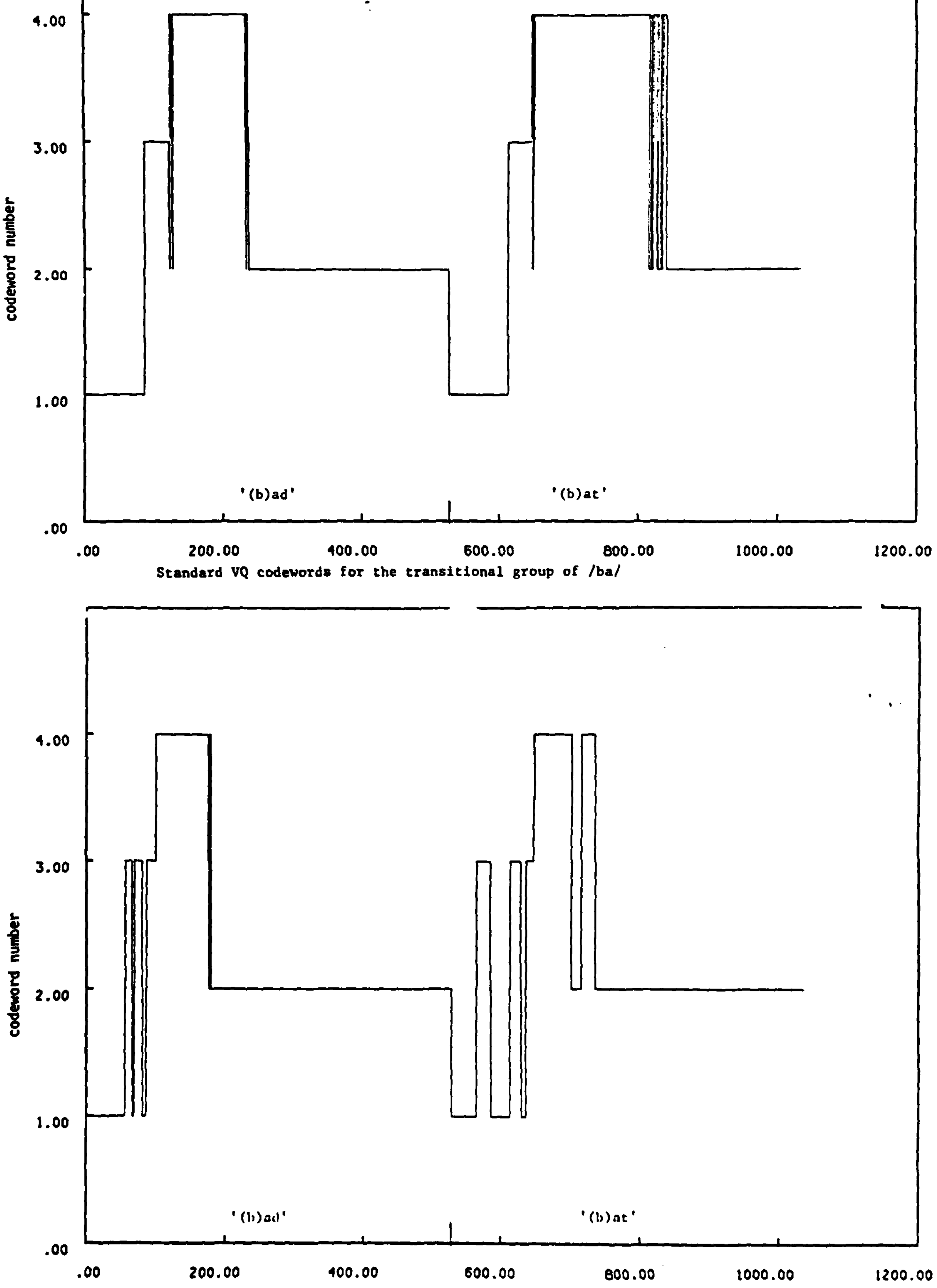

speech data
/a/
/o/
Lattice
VQ
design
vowel codebook
average distortion for each cw : E_i
speech data
/ba/,/ga/,/da/
/bo/,/go/,/do/
Lattice
VQ
design
dist $< 2E_i$
cw_i
steady state parts
dist $\geqslant 2E_i$
transitional group (cw_5)
transitional group
Standard
VQ
Modified
VQ
codebook of size four
for transitional parts
codebook of size four
for transitional parts

Figure 4.14



Classifying /ba/ as Vowel or transitional group

Figure 4.15



Standard VQ codewords for the transitional group of /ba/

Modified VQ codewords for the transitional group of /ba/

Figure 4.16

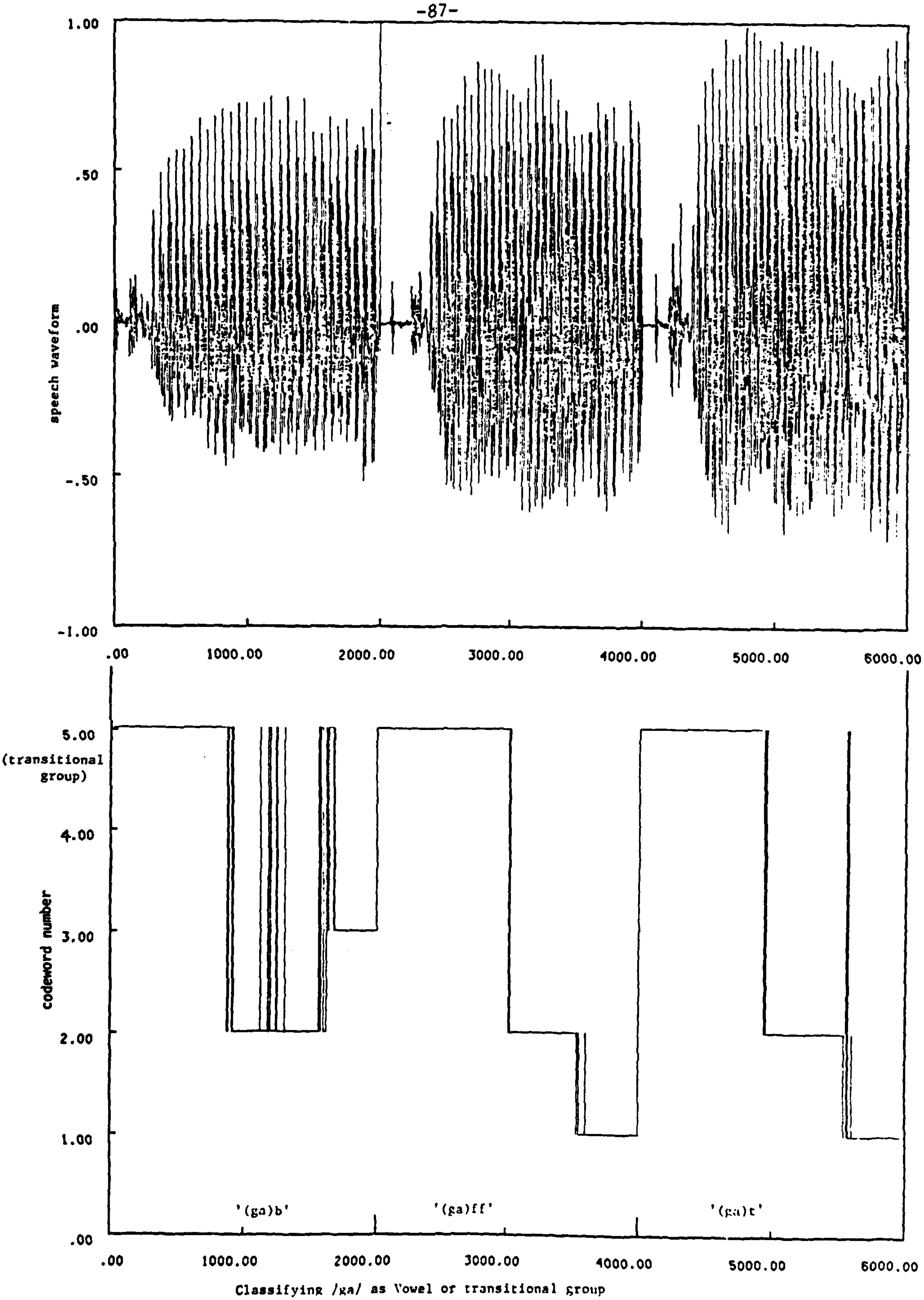

1.00
.50
.00
-.50
-1.00
speech waveform
.00
1000.00
2000.00
3000.00
4000.00
5000.00
6000.00
5.00
(transitional
group)
4.00
3.00
2.00
1.00
.00
codeword number
'(ga)b'
'(ga)ff'
'(ga)t'
.00
1000.00
2000.00
3000.00
4000.00
5000.00
6000.00
Classifying /ga/ as Vowel or transitional group

Figure 4.17

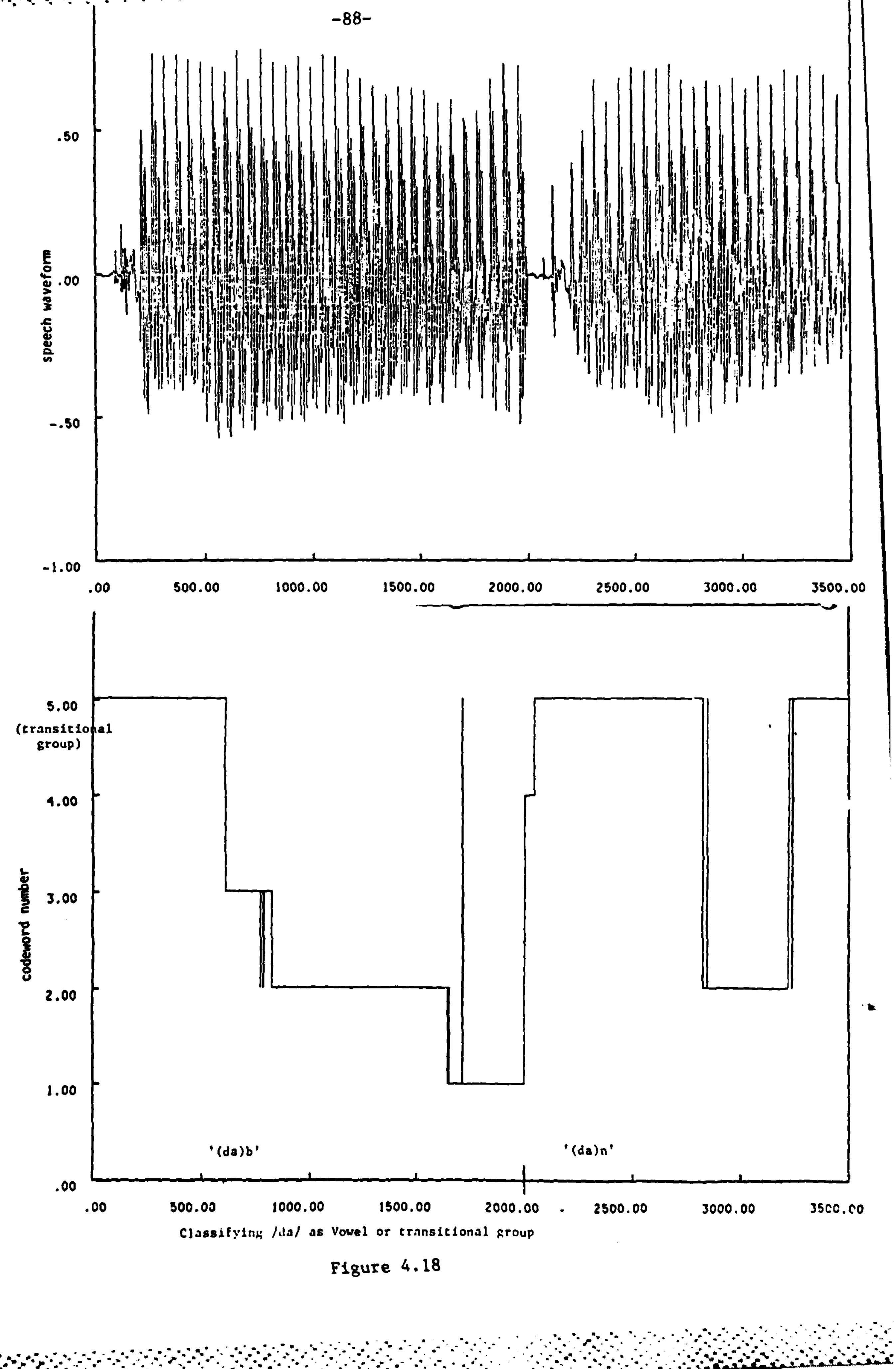

.50
.00
-.50
-1.00
speech waveform
.00
500.00
1000.00
1500.00
2000.00
2500.00
3000.00
3500.00
5.00
(transitional
group)
4.00
3.00
2.00
1.00
.00
codeword number
'(da)b'
'(da)n'
.00
500.00
1000.00
1500.00
2000.00
2500.00
3000.00
3500.00
Classifying /da/ as Vowel or transitional group

Figure 4.18

4.8 RECOGNITION OF VOICED STOP CONSONANTS

The classified VQ approach was tested as a means of recognizing the voiced stop consonants, /b/, /d/, and /g/. The codebooks were computed as in the previous section. The distortion was computed for each 'transitional' codebook applied to the beginning of a test word. The test assumes that the corect vowel has been identified since the 'transitional' codebooks for the various stop consonants depend on the following vowel.

For each word beginning with 'ba', 'ga', or 'da', the transitional part was determined. The standard and modified VQ codebooks for 'ba', 'ga' and 'da', were applied to each word to compute the distortion, Table 7. The IS distortion using the wrong codebooks was at least twice that using the right codebook. By choosing the codebook with the minimum IS distortion, the correct stop consonant was always determined. Using the modified VQ, the correct consonant was chosen but the contribution to the distortion from the reflection coefficient trajectories was not always consistent. Most of the time, the IS component of the distortion dominated the distortion due to the trajectory. This suggested that the weighting factor in this encoding process might need to be changed.

The same experiments were repeated on those words with vowel /o/ and /i/ and these results are in Table 8 and 9. The standard VQ approach always chose the correct stop consonant. Using the modified VQ, the results were correct except for the word 'boast' where the codebooks for 'b' and 'd' were confused. and for the word 'gilt' where the total distortion for the codebooks for 'b' was slightly less than for 'g'. In both cases, the IS component of the modified VQ distortion indicated the correct consonant. However the distortion due to the trajectories was lower on the wrong codebook. The total distortion was only slightly lower for the wrong codebook than for the corect codebook. This points to a problem with the weighting factor that is used to combine the two distortions.

Under the conditions of our study where the same word was used to train the VQ and test for recognition, the correct consonant was identifiable once the following vowel was known. Most of the silence prior to the spoken word was removed from the training sequence. However there

	boast	bone	bowl	ghost	goat	go-90-	dole	dough	doze	
										modified VQ
'b'	.2661	.2640	.2085	.8164	.6889	.7310	.4602	.3767	.4335	total
	.1546	.1425	.1354	.7019	.5820	.6219	.3233	.2343	.3019	IS
	.1149	.1143	.0915	.1597	.1514	.1293	.1304	.1355	.1343	dk1
	.1082	.1289	.0549	.0694	.0625	.0888	.1433	.1494	.1290	dk2
'g'	1.2220	1.4526	.8148	.1718	.2423	.2032	1.1981	1.1766	.6278	
	1.0646	1.2499	.7214	.0757	.1290	.1111	1.0673	1.0267	.5230	
	.2065	.2934	.0747	.0864	.1208	.0848	.1483	.1901	.0976	
	.1083	.1121	.1121	.1058	.1057	.0993	.1134	.1097	.1121	
'd'	.2309	.6452	.2833	.6297	.5329	.4601	.2171	.2005	.2170	
	.1410	.5471	.2115	.5266	.4333	.3615	.1176	.1011	.1317	
	.0756	.0845	.0624	.1079	.1071	.1052	.1025	.0950	.1027	
	.1043	.1117	.0812	.0983	.0922	.0920	.0964	.1038	.0679	
										standard VQ
'b'	.0951	.0900	.1182	.6090	.5361	.5348	.2273	.1641	.2670	IS
'g'	1.0259	1.1914	.6717	.0708	.1218	.1073	1.0545	.9934	.5215	
'd'	.1387	.5447	.2087	.5244	.4355	.3603	.1139	.0992	.1273	

Table 8 Recognition results of stop consonants with vowel /o/

	bid	bit	gill	gilt	did	dip	
							modified VQ
'b'	.19115	.16165	.40434	.35715	.35072	.31618	total
	.12342	.08343	.31994	.26941	.26580	.24083	IS
	.06166	.06748	.07797	.08698	.06890	.07170	dk1
	.07380	.08896	.09083	.08850	.10093	.07900	dk2
'g'	.91977	.39164	.38263	.36857	1.05777	.63908	
	.72942	.30031	.26439	.23271	.89151	.49977	
	.17145	.08522	.06536	.09456	.09846	.09863	
	.20925	.09745	.17113	.17715	.23406	.18000	
'd'	.29257	.62178	.61987	.38270	.19346	.19155	
	.19427	.49728	.50658	.27590	.09713	.09969	
	.09342	.11707	.12942	.12069	.09438	.09432	
	.10318	.13196	.09718	.09292	.09827	.08942	
							standard VQ
'b'	.10242	.08324	.32107	.27138	.26571	.24203	IS
'g'	.60865	.21629	.10809	.11140	.49456	.44875	
'd'	.19532	.49107	.49734	.26494	.08861	.10056	

Table 9 Recognition results of stop consonants with vowel /i/

was a short varying amount of silence in each word which often was represented by a codeword and thus added to the distortion. When the distortions contributed by the samples encoded by the codewords for silence were excluded from the average distortions, almost all the samples in the stop consonant portions were thrown away if the wrong codebooks were used because most of the transitional parts were mapped into the codewords for silence. Thus, all the information from the consonants was lost. If an appropriate way to exclude silence was possible, it would help in the recognition the stop consonants.

	bad	bat	gab	gaff	dab	dan	
							modified VQ
	.14755	.16939	.40074	.40506	.30927	.39356	total
'b'	.10520	.11550	.28296	.31468	.20882	.32280	IS
	.03046	.02204	.11367	.06941	.08200	.03104	dk1
	.05425	.08573	.12189	.11134	.11891	.11049	dk2
	.51074	.45589	.05414	.04417	.06483	.08720	
'g'	.48234	.43019	.03491	.02834	.05054	.06337	
	.02756	.02129	.01621	.01369	.01136	.02313	
	.02923	.03012	.02226	.01797	.01722	.02453	
	.55131	.48383	.07982	.09300	.02672	.04928	
'd'	.53016	.46423	.06391	.08152	.01234	.03168	
	.02037	.01633	.01405	.01042	.01400	.01596	
	.02193	.02288	.01777	.01254	.01477	.01925	
							standard VQ
'b'	.07782	.10744	.14382	.15048	.09005	.18869	IS
'g'	.53383	.43642	.03461	.02368	.05351	.07491	
'd'	.52963	.46376	.06092	.07958	.01207	.03073	

Table 7 Recognition results of stop consonants with vowel /a/

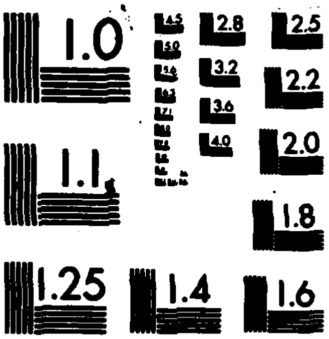
4.9 SUMMARY AND FUTURE WORK

This research effort has lead to an understanding of the combination of recursive estimation with vector quantization. The ability to track quickly changing signal characteristics and classify them into a small number of signal types, provides a powerful signal processing tool. The additional information provided by trajectories of coefficients was useful to separate steady state and transitional signal segments. The classified VQ approach allows different signal segments to be quantized (or clustered) into a varying number of levels. For speech recognition, particularly for phoneme based approaches where quickly changing consonants must be identified, this method appears very useful. A method of recognizing the voiced stop consonants, /b/, /d/, and /g/ was developed and tested using this approach. For the limited data base of words, the method accurately determined the consonant for various following vowels.

Future research activities would include an investigation of the combined recursive estimation and vector quantization for speech transmission, an extended look at the recognition problem to reduce the effect of the following vowel, and a recognition test using a larger data base. There is considerable potential for theoretical developments in combined recursive estimation and quantization, use of parameter trajectories for signal classification and 'adaptive' vector quantization using the classification approach.

AD-A137 569   FAST ALGORITHMS FOR IMPROVED SPEECH CODING AND   2/2
RECOGNITION(U) STANFORD UNIV CA INFORMATION SYSTEMS LAB
J M TURNER ET AL. 31 DEC 83 ISL-M736-3 N00014-82-K-0492
UNCLASSIFIED   F/G 12/1   NL

END
FILMED
3-84
DTIC


1.0
1.1
1.25
1.4
1.6
1.8
2.0
2.2
2.5
2.8
3.2
3.6
4.0
4.5
5.0
5.6
6.3
7.1

4.10 REFERENCES

[AMLA]H.M. Ahmed, M. Morf, D.T. Lee and P.H. Ang, "A VLSI Speech Analysis Chip Set Based on Square-Root Normalized Ladder Forms," *Proc. 1981 IEEE ICASSP,* Atlanta, GA, pp. 648-653, March 30-April 1, 1981.

[BS] Blumstein.,S. and Stevens, K., "Perceptual invariance and onset spectra for stop consonants in different vowel environments" J. Acoust. Soc. Am., vol 67, no 2, pp 648-662,Feb. 1980.

[LM] Lee, D.T.L. and M. Morf, "Recursive Square-Root Estimation Algorithms," *Proc. IEEE ICASSP,* Denver, CO, pp. 1005-1017, April 1980.

[ML] Morf,M. and Lee,D.,'Fast Algorithms for Speech Modeling', Tech. Report M308-1, Information System Laboratory, Stanford University, Dec 1978.

[PB] Peterson,G. and Barney,H., "Control Methods Used in a Study of the Vowels", J. Acoust. Soc. Am., Vol 24, No. 2, pp 175-184, March 1952.

5. SUMMARY

During the course of this research contract, estimation techniques for processes that contain Gaussian noise and jump components, and classification methods for transitional signals by using recursive estimation with vector quantization were studied. New theoretical techniques were developed and practical application considered. Experience was gained in recursive estimation and vector quantization techniques and an investigation of their combined use was begun.

Three technical reports were issued during this project. The first report, M736-1 presented a detailed discussion of "Simultaneous Jump Excitation Modeling and System Parameter Estimation". The second report, M736-2 presented an overview of recursive least squares estimation and lattice filters. This final technical report is the third report and focused on pitch estimation and stop consonant recognition. Here in this last report, the combination of recursive estimation and vector quantization is studied for the first time.

It is our intent to continue studying signal processing techniques that utilize the fast tracking nature of recursive estimation and the efficient classification features of vector quantization. Hopefully, future contracts will allow us to continue this research.

6. TECHNICAL PUBLICATIONS

Turner,J. "Application of Recursive Exact Least Square Ladder Estimation Algorithm for Speech Recognition", Int. Conf. Acoustics,Speech and Signal Processing, May 1982, pp 543-555.

Stirling,W. and Turner,J. "Joint Estimation of Excitation and Vocal Tract Response", Int. Conf. Acoustics,Speech and Signal Processing, April 1983.

Turner,J., "Recursive Least Squares Estimation and Lattice Filters", a chapter in *Adaptive Filters,* Cowan,N. and Grant,P. (editors) , Prentice Hall, to be published 1984.

Huang. S-S. and Turner,J. "Speech Recognition using Recursive Estimation" in preparation.

ONR REPORTS

Stirling,W. and Turner,J. "Simultaneous Jump Excitation Modeling and System Parameter Estimation", technical report M736-1 for Office of Naval Research, Contract N00014-82-K-0492, Feb. 1983.

Turner,J., "An Overview of Recursive Least Squares Estimation and Lattice Filters", technical report M736-2 for Office of Naval Research, Contract N00014-82-K-0492, Jan. 1984.

Turner,J. et al., "Fast Algorithms for Improved Speech Coding and Recognition", final technical report M736-3 for Office of Naval Research, Contract N00014-82-K-0492, Dec. 1983.

7. ACKNOWLEDGEMENT

The assistance of Prof. R. Gray and his students at Stanford University on Vector Quantization techniques was appreciated.

Computer facilities were partially supported by the Defense Advanced Research Projects Agency under Contract MDA903-82-K-0382.

The work of W. Stirling was partially supported by ESL Inc., Sunnyvale, CA, 94086.

END

FILMED

3-84

DTIC