Design and implementation of 4800 BPS CELP Coder (U.S. federal standard 4800 BPS voice coder)

Author:
Chen, Zhao

Publication Date:
1991

DOI:
https://doi.org/10.26190/unsworks/10706

DESIGN AND IMPLEMENTATION OF 4800 BPS CELP CODER

(U.S FEDERAL STANDARD 4800 BPS VOICE CODER)

Chen Zhao

University of New South Wales, Communications Department,

Speech Group

A thesis submitted in partial fulfilment of the requirements for the degree of Master of Engineering Science in the School of Electrical Engineering at the University of New South Wales.

April, 1991

ACKNOWLEDGEMENT

I would like to thank my supervisor, Associate Professor W. Harvey Holmes and commercial collaborator, Dr Stan. K. Baker for their support and encouragement throughout the course of this research.

I would also like to thank Mr Dip Sen, Mr Mark Prandollini, Mr Noel Gordon for their knowledge and discussions on the subject of this thesis.

My special thanks to Mr Noel C. Kaarsberg and Mr Dip Sen for their assistance during the course of this research and for their language help during the writing up of this thesis.

Finally, I am grateful to my wife Yao Nan for her continuous support and encouragement which have contributed immensely to this work.

This thesis is dedicated to my parents and my wife.

ABSTRACT

The objectives of this project are to implement and study the U.S. Government Department of Defense (DoD) code excited linear predictive (CELP) speech coder which operates at 4.8 kbps and which has been reported as providing "very good" intelligibility and "excellent" quality with less computational complexity than alternative.

The implementation procedure in this project follows the principles of the DoD CELP algorithm. The coder system is divided into two main parts: (i) finding the parameters of the vocal tract filter using linear predictive analysis techniques and (ii) finding the parameters of the excitation function using analysis-by-synthesis techniques.

In finding the vocal tract parameters, an efficient coding technique using a fast Line Spectrum Pairs search table for the quantization of the LPC parameters is presented. In the procedure of the synthesized speech, an adaptive code book is developed for long term prediction of the speech and a stochastic ternary overlapped sparse code book is presented for new code excitation. Both of these show a revolutionary change in the class of CELP coders.

The results of subjective testing are given together with further improvement in the algorithm.

Contents

Chapter 1

Introduction and Design Guidelines

1.1 INTRODUCTION

Speech is the simplest and most natural means for one person to convey information to another. Many modern communications systems are devoted to the transmission and, to a lesser extent, storage of voice signals. With the increase of digital communications, processing of speech signals with small, compact computers and digital devices is required.

Speech coding deals with analysis and synthesis techniques for reducing the bandwidth needed to transmit speech over a communications channel. Bandwidth is usually expressed as the bit rate, in bits per second (bps). The purpose of speech coding is to transmit digital speech at low bit rates without distorting voice quality. The amount of information that is contained in a speech signal is not precisely known; the analog speech signal is a continuous signal whose information content,

in theory, could be large. The primary problem in reducing the bit rate of a speech signal is that some of the information must be discarded in the process of speech coding. The signal processing capabilities of the human auditory channel are however limited and the information which least affects the perceived quality of the speech can be discarded. In general, there is also a trade off between a lower bit rate and increased computational complexity of the algorithms for reducing the bit rate.

The primary research goal of this project was to study and develop techniques for near toll quality speech coders at 4.8 kbps. The techniques of interest were thus limited to those which could achieve reasonable quality at 4.8 kbps, which could operate acceptably in a real-time environment, and which could be realized using reasonable computational resources. That is, the coders of interest had to achieve a compromise between quality, robustness, delay, and complexity operating at the 4.8 kbps limit.

The U.S. Government Department of Defense (DoD) 4.8 kbps Code Excited Linear Predictive (CELP) version 3.1 coder [7] is extremely attractive because it out-performs all other coder standards operating at rates below 16 kbps, with a performance comparable to that of 32 kbps continuously variable slope delta modulation (CVSD). A study and implementation of DoD CELP version 3.1 are carried out in this thesis.

Since this DoD standard falls into a general class of Linear Predictive (LP) speech coders, the thesis starts from the formulation of the basic LP and CELP coders, and then describes the DoD CELP algorithm and the software designs.

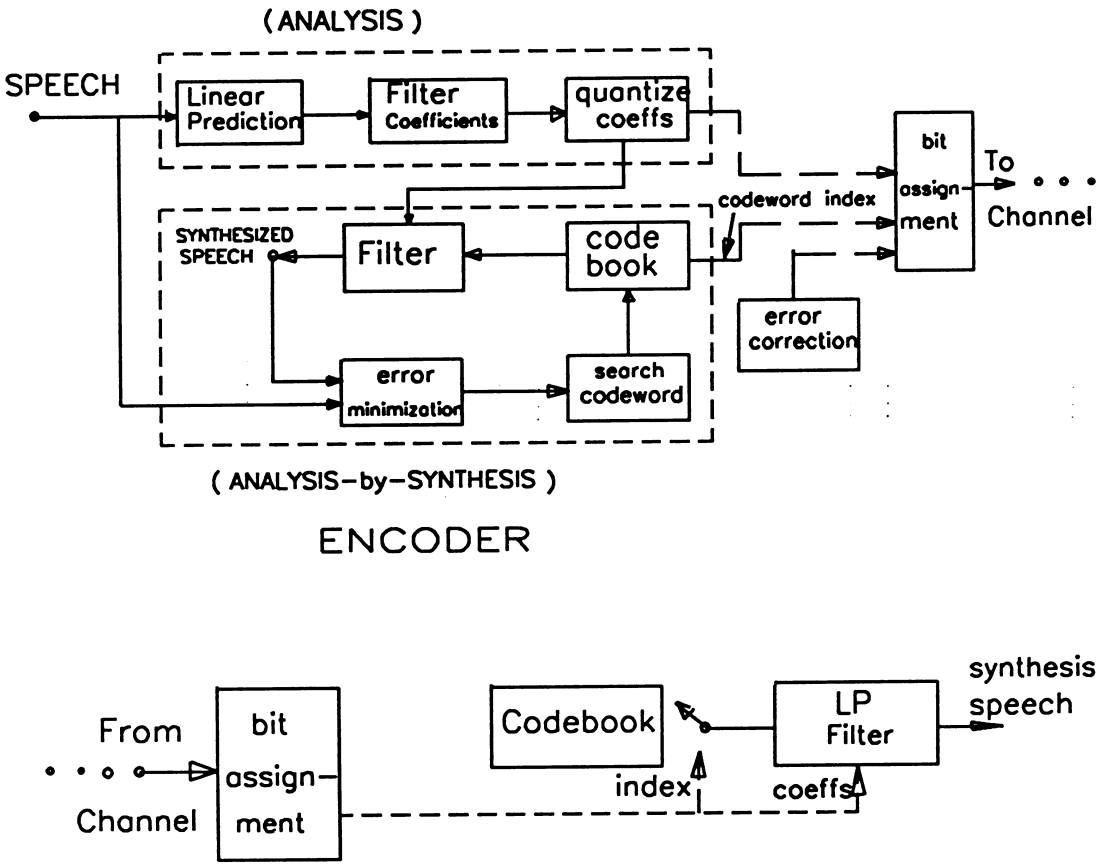


Figure 1.1 Linear Predictive Coder

As shown in Fig 1.1, all of the LP coders model speech as a short-time stationary, time-varying vocal tract filter excited by a parametrically generated excitation signal. This class of coders finds the coefficients of their vocal tract filters using a Linear Predictive ***analysis*** technique and the parameters of their excitation functions using an ***analysis-by-synthesis*** technique. The coefficients of the filter are quantized so that the same filter can be constructed at both the transmit and receive ends of the channel. A set of candidate excitation sequences is stored in a codebook, and synthetic speech is generated using each of these sequences. The index of the sequence producing the most ***accurate*** speech is then transmitted to the receiver.

Many potential applications for low bit rate speech coding algorithms demand good speech at a reasonable cost. The original CELP algorithm is shown in Figure 1.2. The speech synthesis model is composed of three separate components: a short-term predictor, a long-term predictor, and an excitation signal. In such systems, the function of the short-term predictor is to model the slowly varying spectral envelope of the speech signal. The long-term predictor is primarily intended to model the pitch redundancy in voiced sound. Finally, the function of the excitation signal is to excite the system and to model all perceptually important features of the speech signal which are not well modelled by the short-term and long-term predictor. For all speech coders in this class, the excitation signal is chosen (from a fixed ensemble of possible excitation signals) by minimizing the

energy in the weighted difference between the original speech signal and the coded speech signal. This CELP coder was found to provide good speech quality at intermediate bit rates (4.8-9.6 kb/s), [1]. However, the speech quality was obtained at the expense of very high computational complexity and huge memory, making real-time implementation on low-cost hardware (with one or two general purpose digital signal processing (DSP) devices) impossible.

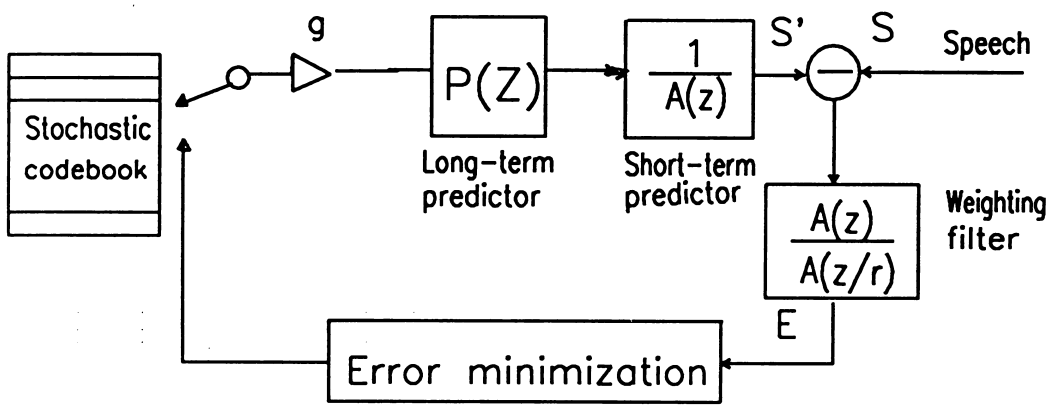


Figure 1.2. ORIGINAL CELP

CELP's major computational requirements are dominated by the code book and pitch searches. The computational complexity and speech quality of the coder depend upon the search sizes and the structure of the code book. Recently, many new CELP-type algorithms have been implemented to enhance CELP performance with less computation and memory. The U.S. DoD 4.8 kbps CELP algorithm is

the culmination of already developed CELP-type coders [4] [5] [6]. The major difference from the original CELP is in the structure of the excitation. In this system, long-term signal periodicity is modelled by a vector quantizer (VQ) adaptive code book. The adaptive codebook becomes one part of the excitation. The second excitation is a VQ fixed stochastic code book. The use of a special stochastic code book also benefits coder performance.

In this thesis, the DoD coder is emphasized because of its good quality/complexity ratio and because of the implications of its good performance for linear predictive analysis-by-synthesis coders in general.

1.2 ORGANIZATION OF THE THESIS

This section summarizes the contents of each of the following five chapters.

An LP analysis technique for evaluating and coding the linear predictive coding (LPC) coefficients is presented in Chapter Two. It begins with a derivation of the LPC equations and is followed by the Levision-Durbin algorithm used to compute efficiently the LPC coefficients from the LPC equations. The Line Spectrum Pairs (LSP) technique is used for LPC coefficient quantization. A mathematical treatment of LSP and a software design procedure is presented in the last section of this chapter.

Chapter Three discusses the background of the DoD 4.8 kbps CELP coder. The characteristics of analysis-by-synthesis technique are listed to provide basic information for the work described in later chapters.

Chapter Four presents an original code search algorithm that is the most important point of the DoD coder. A code search model and the principles of the adaptive code book and of the stochastic ternary code book are given in this chapter. The advantages of this method of code search algorithm , such as efficient computation and performance enhancement, are pointed out during the disscussion.

Experimental results of the system implemented are presented in Chapter Five. This chapter also includes an outline of the whole coding process from the original speech to the synthesised speech. A simple waveform comparison and pairwise comparison of informal listening tests are used for subjective quality estimation.

Finally, Chapter Six presents the conclusions that can be drawn from this thesis. Possible future areas of research that may improve the performance of the DoD CELP coder are also suggested.

Chapter 2

Linear Prediction

2.1 INTRODUCTION

In a LPC system, the vocal tract is modelled as a filter that shapes the spectrum of the speech. The spectral envelope represented by an LPC filter, [28], in turn, preserves information that is important to the perceived quality of the speech (e.g. resonances or formants in the frequency responses of the filters). The filters, which model the vocal tract as it changes from one instant to another, are also a very compact representation. The coefficients of the filter are quantized so that the same filter can be constructed at both the transmit and receive ends of the channel. Although the LPC filter coefficients are a compact representation of the vocal tract, it is possible to further reduce the number of bits through coding and quantization techniques. Efficient encoding of LPC parameters is also beneficial for the CELP algorithm, since more bits can then be available for the excitation parameters. The Line Spectrum Pairs (LSP) technique is used for coding the LPC coefficients.

This chapter starts with a derivation of the LPC Equations (or normal equations). The LP analysis, which is based on autocorrelation procedure, produces matrix forms (from an expandsion of the normal equations) which are both symmetric and Toeplitz. The Levision-Durbin algorithm which most efficiently solves the matrix is used in section 3 to evaluate the LPC coefficients. Finally, a fast LSP table search technique is used to quantize the LPC coefficients. This is described in the last section.

2.2 LINEAR PREDICTIVE CODING EQUATION

Linear Prediction is a modelling and spectral estimation technique [18][29]. The speech production system is modelled as a linear system, H(z), which is excited by a signal, E(z). The excitation signal is passed through a filter which models the glottal pulse shaping, the vocal tract, and the lip radiation impedance. The speech analysis problem is then to determine the *parameters* of the underlying model for a given short segment of speech. The linear prediction technique is used to obtain an *estimate* of these parameters based on the observed waveform. The Department of Defence coder refers to this as either "the frame rates and parameter coding methods " or " spectrum analysis". The term frame here corresponds to a short segment of speech.

According to the linear model of speech production, the speech signal results from the excitation signal passing through the linear system:

$$S(z) = E(z)H(z) \tag{2.1}$$

For this application, the system transfer function, H(z), can be assumed to be an all pole filter since the prediction is based on the spectral envelope involving short delays [26] [27] [28], and the most important perceptual feature are those caused by poles in the transfer function. It is also better to use an all pole filter since the estimation of all-pole parameters involves only the solution of linear equations. The all pole transfer function can be expressed as:

$$H(z) = \frac{1}{A(z)}$$

where A(z) is an m^{th} order polynomial of the form:

$$A(z) = 1 + \sum_{i=1}^{m} a_i z^{-i}$$

The zeros of A(z) become the poles of H(z). To ensure the stability of the synthesis filter H(z), it is required that the zeros of A(z) be inside the unit circle (both initially and when the coefficients are quantized for transmission or storage).

Given the linear speech production system in equation (2.1), the analysis problem can be reinterpreted as finding the inverse filter which produces the minimum energy output signal for an observed speech waveform, i.e., A(z) is the inverse of the speech production filter H(z). Equation (2.1) can be rewritten by replacing H(z) with 1/A(z):

$$E(z) = S(z)A(z) \tag{2.2}$$

If the inverse filter is an exact match of the system H(z), the signal e(n) will be *exactly* the excitation signal. If the match is imperfect, as in the case for actual speech, e(n) will be a combination of the excitation and error signals. The error signal represented by equation (2.2) can be interpreted as the difference between the input signal s(n), and an estimate $\hat{s}(n)$, based on a linear combination of the past samples:

$$e(n) = s(n) - \hat{s}(n) \tag{2.3}$$

where $\hat{s}(n)$ is given by:

$$\hat{s}(n) = -\sum_{i=1}^{m} a_i s(n-i) \tag{2.4}$$

The coefficients, $\{a_i\}$, are called the linear prediction coefficients and are chosen by minimizing some error measure. A commonly used measure is the Mean Square Error (MSE), denoted as

$$\varepsilon(n) = E\{e^2(n)\} \tag{2.5}$$

where $E\{\cdot\}$ is the Expectation operator. The MSE is attractive because it leads to efficient mathematical solutions for the prediction coefficients, [29]. Since voiced speech is considered a quasi-stationary source over a few voice pitch periods [28],the error in (2.5) can be defined over N samples (e.g. N=240) of the input sequence:

$$E = \sum_{n=0}^{N-1} e^2(n) \tag{2.6}$$

Substituting eqs.(2.3) and (2.4) in (2.6), then taking the partial derivative of E with respect to each a_k, and setting the derivatives equal to zero, we obtain a set of simultaneous linear equations for the filter coefficients a_i:

$$\sum_{i=1}^{m} a_i c_{ik} = -c_{0k} \tag{2.7}$$

for k = 1,2...,m. The quantities c_{ij} are defined as:

$$c_{ij} = \sum_{n=n_0}^{n_1} s(n-i)s(n-j) \tag{2.8}$$

for i=1,2,...,m and j = 1,2,...,m. The equation (2.7) is often called the set of *normal* equations. If the limits in eqn.(2.8) are minus and plus infinity the solution process is classified as the *autocorrelation* method. A finite length window function w(n) in turn, causes the coefficients c_{ij}, to be equivalent to the short-time autocorrelation sequence:

$$R_k = \sum_{n=0}^{N-1-k} s(n)w(n)s(n+k)w(n+k) \tag{2.9}$$

for k =0,1,..,m. Then eqn. (2.7) becomes :

$$\sum_{k=1}^{m} a_k R_{i-k} = -R_i \tag{2.10}$$

If eqn.(2.10) is expanded into full matrix form, the matrix will be both symmetric and Toeplitz. The solution for the coefficients, $\{a_i\}$ in eqn.(2.10), requires the solution of the Toeplitz system of equations (2.9). Levinson-Durbin developed a fast algorithm (described in the next section) to evaluate the coefficients, $\{a_i\}$.

A typical window function used in eqn.(2.9) is the Hamming window. There are two methods that can be used for the Hamming window function in the software design:

(1) Save only half the data points of the Hamming window since it is symmetric. For example, 'win' is stored in the lower 120 points of the window data. The upper 120 points can be obtained by reading 'win' backwards from tail to head.

(2) A recursive routine can also be used to compute the Hamming window data.

The Hamming window is specified by the following equation.

$$w(n)=0.54-0.46\cos\left(\frac{2\pi n}{N-1}\right) \quad \text{for } 0\le n \le N-1 \ (N=240)$$

where $\cos\left(\frac{2\pi n}{N-1}\right)$ is iteratively computed by:

$$\cos\left(\frac{2\pi(n-1)}{N-1}\right)\cdot\cos\left(\frac{2\pi}{N-1}\right)-\sin\left(\frac{2\pi(n-1)}{N-1}\right)\cdot\sin\left(\frac{2\pi}{N-1}\right)$$

The values of $\cos\left(\frac{2\pi}{N-1}\right)$ and $\sin\left(\frac{2\pi}{N-1}\right)$ are stored in a lookup table in the routine.

Comparing these two methods, the implementation of the second window method is more conservative in its use at memory (about 120 floating point memory locations).

2.3 LEVINSION-DURBIN ALGORITHM

As stated before, the Levinsion-Durbin algorithm is a procedure for computing the prediction coefficients for an m^{th} order predictor based of the first m+1 values of the short-time sample autocorrelation sequence. It results in an efficient recursive procedure that solves the Normal equations using the autocorrelation method [18]. The coefficients are computed for increasing predictor order until the desired order is reached. The algorithm is as follows:

(1) Initiallization:

$$a_0^{(1)} = 1$$

$$a_1^{(1)} = -R_1/R_0$$

$$k_1 = -a_1^{(1)}$$

$$E_1 = R_0(1 - k_1^2)$$

where R_I is the short-time autocorrelation sequence.

(2) Compute a new set of coefficients for i = 1,2,...,m

(2a) compute the i^{th} coefficient

$$a_i^{(i)} = R_i - \sum_{j=1}^{i-1} a_j^{(i-1)} R_{i-j}$$

$$k_i = -a_i^{(i)}$$

(2b) compute the remaining coefficients for j=1,2,...,i-1.

$$a_j^{(j)} = a_j^{(j-1)} - k_i a_{i-j}^{(i-1)}$$

(2c) compute the error for the i^{th} order predictor

$$E_i = E_{i-1}(1 - k_i^2)$$

The superscripts of the a_i's represent the iteration number. Hence, the coefficients, $\{a_j^{(i)}\}$, for the i^{th} iteration are computed based on those calculated in the previous iteration. The quantity, E_i, is the mean squared error between the original speech and the synthesised speech of the i^{th} order predictor, and will always decrease or remain the same as the predictor order increases.

The coefficients, $\{k_i\}$, are called PARCOR (PARtial CORrelation) or reflection coefficients depending on the sign convention used. They possess an interesting and desirable property in that the stability of the speech synthesis filter is guaranteed (i.e., all of the root of A(z) lie inside the unit circle) if the magnitude of each k_i is less than one. It is obvious that the reflection coefficients k_i are equal

to 1 in some procedure step i. Equivalently, this can only happen if the Minimum error energy E_i in the routine process is zero . Therefore, E_i may be checked at the end of each step in the recursion. If E_i is negative or zero, then an error has been made and the process should terminate.

The software model can be found in references [18] and [19]. It is important to note that the algorithm must be tested for it to work properly. Markel's test program in 3.3.6 of ref.[18] can be used for this purpose.

2.4 LINE SPECTRUM PAIRS

2.4.1 The LPC Quantization

As mentioned at the beginning of this chapter, a suitable quantization technique may further reduce the number of bits needed to transmit or store the coefficients. Indeed, in the context of speech compression, it is well known that the LPC coefficients are inappropriate for quantization. This is primarily due to their wide dynamic range and the problems of instability of the synthesis filter. LSP and PARCOR quantization are commonly used alternative representation of the filter parameters.

The advantage of using the LSP frequencies for the coding techniques has been reported in many papers such as ref.[20] and [21]. With the use of LSPs, the

perceived quality of the synthetic speech is at a higher level than that of comparable coders not using LSPs, but this is at the cost of the bit rate . The bit rate reduction is achieved by using very few bits to code formant bandwidth information, while the more perceptually important formant location information is coded more precisely. The bit rate of the fixed frame rate LSP coders is 25 to 35 percent less than similar PARCOR coders [21], but the quality of the synthetic speech is rated higher in the Diagnostic Acceptability Measure (DAM, defined in ref.[8]).

However, Bishnu S.Atal [22] in 1989 reached the new conclusion that the performance of non-uniform scalar quantizers using three different LPC representations (the arcsines of reflection coefficients, log area ratios and line spectral frequencies) were very similar. This implies that there is no inherent advantage in using Line Spectral frequency based quantization when compared to refection coefficient based quantization.

Nevertheless, DoD CELP still uses LSP quantizers. The reason could be that the LSP speech analysis-synthesis method is known as one of the most efficient vocoders, and encoding of the LSP parameters in a certain ordered relationship can ensure the stability of the synthesis filter [23][24][21]. LSP fixed encoding/decoding levels are employed by DoD for a fast search, (see section 3 below).

2.4.2 The Aspects of LSP

A brief description of the LSP speech analysis-synthesis procedure are now given, including implications and the quantization algorithm.

For a given order m, LPC analysis results in an inverse filter

$$A_m(z) = 1 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_m z^{-m} \tag{2.11}$$

which minimizes the residual energy. The parameters $\{a_i\}_{i=1,2,\ldots,m}$, are commonly called the LPC coefficients.

It is easy to verify that the polynomial $A_m(z)$, associated with an m^{TH}-order LPC analysis, satisfies the following recurrence relationship:

$$A_{n+1}(z) = A_n(z) - K_{n+1} z^{-(n+1)} A_n(z^{-1}) \tag{2.12}$$

$$n = 0, 1, 2, \ldots, m,$$

where $A_0(z) = 1$. The quantities $\{k_i\}_{i=1,2,\ldots,m}$, are the PARCOR coefficients. They may also interpreted as the *reflection coefficients* at the boundaries of the acoustic tube model for the vocal tract.

Eqn.(2.12) may be considered at two extreme artificial boundary conditions, namely, $k_{m+1} = 1$ and $k_{m+1} = -1$. These conditions correspond, respectively, to a complete closure and a complete opening of the glottis in the acoustic tube model. Under these conditions, the polynomial $A_{m+1}(z)$ coincides with the polynomials

$$P(z) = A_m(z) - z^{-(m+1)}A_m(z^{-1}) \tag{2.13a}$$
$$= 1 + (a_1 - a_m)z^{-1} + (a_2 - a_{m-1})z^{-2} + \cdots$$
$$-(a_2 - a_{m-1})z^{-(m-1)} - (a_1 - a_m)z^{-m} - z^{-(m+1)}$$

for $k_{m+1} = 1$, or

$$Q(z) = A_m(z) + z^{-(m+1)}A_m(z^{-1}) \tag{2.13b}$$
$$= 1 + (a_1 + a_m)z^{-1} + (a_2 + a_{m-1})z^{-2} + \cdots$$
$$+(a_{m-1} + a_2)z^{-(m-1)} + (a_m + a_1)z^{-m} + z^{-(m+1)}$$

for $k_{m+1} = -1$.

It is obvious that P(z) is an anti-symmetric polynomial and Q(z) is a symmetric polynomial, with

$$A_m(z) = [P(z) + Q(z)]/2 \tag{2.14}$$

Each polynomial P(z) and Q(z) is of order m+1, but P(z) always has a root at z = +1, while Q(z) has a root at z = -1. Hence, the two can be factored into m^{th} order polynomials:

$$\overline{P}(z) = P(z)/(1-z^{-1}) \tag{2.15a}$$
$$= 1 + p_1 z^{-1} + \cdots + p_1 z^{-(m-1)} + z^{-m}$$

$$\overline{Q}(z) = Q(z)/(1+z^{-1}) \tag{2.15b}$$
$$= 1 + q_1 z^{-1} + \cdots + q_1 z^{-(m-1)} + z^{-m}$$

When m is an even integer, it is easily shown that the polynomials P(z) and Q(z) can be expressed as

$$P(z) = (1-z^{-1}) \prod_{i=1,2,\cdots,m/2} (1 - 2z^{-1}\cos w_{pi} + z^{-2}) \tag{2.16a}$$

and

$$Q(z) = (1+z^{-1}) \prod_{i=1,2,\cdots,m/2} (1 - 2z^{-1}\cos w_{qi} + z^{-2}) \tag{2.16b}$$

where it is assumed that $w_{p1} < w_{p2} < \cdots < w_{p\frac{m}{2}}$ and $w_{q1} < w_{q2} < \cdots < w_{q\frac{m}{2}}$.

The polynomial P(z) and Q(z) possess some very interesting and important properties [25] summarized as follows:

(1) All zeros of P(z) and Q(z) are on the unit circle;

(2) The zeros of P(z) and Q(z) are interlaced with each other;

(3) The minimum phase property of $A_m(z)$ is easily preserved after quantization of the zeros of P(z) and Q(z).

These properties are basic for the quantization technique. The quantization procedure using a fast table search method will be presented in detail in the following section.

2.4.3 Coding algorithm with fast table search LSP

Since the zeros of P(z) and Q(z) are on the unit circle , they can be expressed as e^{jw}, where the w's are called the LSP frequencies.

$$P(w) = (1 - e^{-jw}) \prod_{i=1,2,\cdots,m/2} (1 - 2\cos w_{pi} e^{-jw} + e^{-2jw}) \qquad (2.17a)$$

$$Q(w) = (1 + e^{-jw}) \prod_{i=1,2,\cdots,m/2} (1 - 2\cos w_{qi} e^{-jw} + e^{-2jw}) \qquad (2.17b)$$

Considering property 2 , P(z) always has a zero at w=0 and Q(z) always has zero at w = π. Hence, we can assume that

$$w_{pi} = w_{2i}$$

and

$$w_{qi} = w_{2i-1}$$

More specifically, the following relationship is always satisfied:

$$0 = w_0 < w_1 < w_2 < \cdots < w_{m-1} < w_m < w_{m+1} = \pi. \tag{2.18}$$

From now on, we refer to the above relationship as the *ordering property* of the LSP parameters. It has been shown in ref. [21] that the ordered LSPs will enable the synthesis filter $1/A_m(z)$ to be stable, since ordered LSPs result in $A_m(z)$ with minimum phase.

Referring to equation (2.15), we need only evaluate the polynomials $\overline{P}(z)$ and $\overline{Q}(z)$ on the unit circle; in particular:

$$\overline{P}(w) = 2e^{-j\frac{m}{2}w} P'(w) \tag{2.19a}$$

and

$$\overline{Q}(w) = 2e^{-j\frac{m}{2}w} Q'(w) \tag{2.19b}$$

where

$$P'(w) = \cos\frac{mw}{2} + p_1 \cos\frac{(m-1)w}{2} + \cdots + \frac{1}{2}p_{m/2} \tag{2.20a}$$

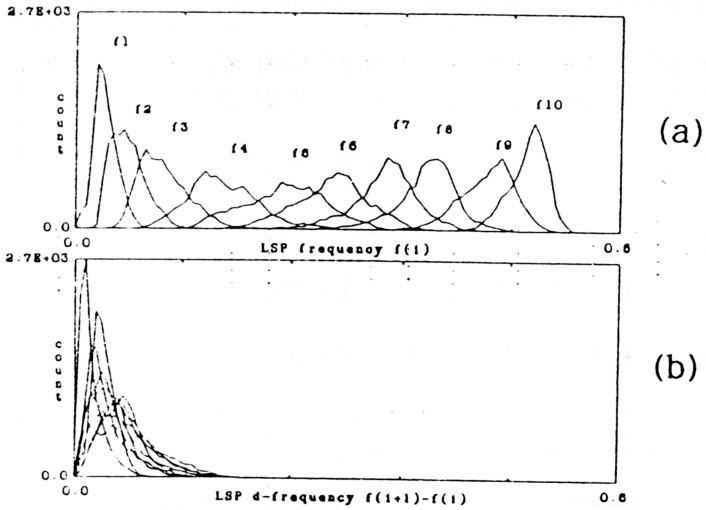
$$Q'(w) = \cos\frac{mw}{2} + q_1 \cos\frac{(m-1)w}{2} + \cdots + \frac{1}{2}q_{m/2} \tag{2.20b}$$

The CELP algorithm uses a 10th order LPC. The resulting P'(w) and Q'(w) are

$$P'(w) = \cos 5w + p_1 \cos 4w + p_2 \cos 3w + p_3 \cos 2w + p_4 cosw + \frac{1}{2}p_5 \tag{2.21a}$$

$$Q'(w) = \cos 5w + q_1 \cos 4w + q_2 \cos 3w + q_3 \cos 2w + q_4 cosw + \frac{1}{2} q_5 \quad (2.21b)$$

All the LSPs can then be evaluated from equation (2.21). However, more efficient algorithms are available. Since the LSP frequencies are distributed in an ordered manner along the frequency axis and the range of distributions for each frequency is highly limited (Figure 2), a reasonable initial frequency quantized level (table 1) is used by the DoD CELP coder.



(a)

(b)

(a) Histograms of LSP frequencies.
(b) Histograms of LSP frequency differences.

Figure 2: LSP Histograms (from Sugamura 1988)

LSP	Bits	Output Levels (Hz)
1	3	100, 170, 225, 250, 280, 340, 420, 500
2	4	210, 235, 265, 295, 325, 360, 400, 440, 480, 520, 560, 610, 670, 740, 810, 880
3	4	420, 460, 500, 540, 585, 640, 705, 775, 850, 950, 1050, 1150, 1250, 1350, 1450, 1550
4	4	620, 660, 720, 795, 880, 970, 1080, 1170, 1270, 1370, 1470, 1570, 1670, 1770, 1870, 1970
5	4	1000, 1050, 1130, 1210, 1285, 1350, 1430, 1510, 1590, 1670, 1750, 1850, 1950, 2050, 2150, 2250
6	3	1470, 1570, 1690, 1830, 2000, 2200, 2400, 2600
7	3	1800, 1880, 1960, 2100, 2300, 2480, 2700, 2900
8	3	2225, 2400, 2525, 2650, 2800, 2950, 3150, 3350
9	3	2760, 2880, 3000, 3100, 3200, 3310, 3430, 3550
10	3	3190, 3270, 3350, 3420, 3490, 3590, 3710, 3830

Table 1: Spectrum Encoding/Decoding frequencies

In practice, the quantization routine (1) adjusts the quantization if LSPs are nonmonotonically quantized , and(2) finds the minimum quantization error adjustment without evaluating the actual LSPs (or zeros of P'(w) and Q'(w))in equation (2.21). The details are listed below:

Let

$$\hat{w}_{i,j_i} = 2\pi \hat{f}_{i,j_i} T \tag{2.22}$$

where i=1,2,3,$\cdots$,10 are subscripts representing10 quantized LSP frequencies, j_i=0,1,$\cdots$, $2^{u[i]} - 1$ are the level indexes of each quantized LSP frequency and u = {3,4,4,4,4,3,3,3,3,3}.

Note that P'(0) = Q'(0) = 1 > 0. Ten quantized LSP frequencies are determined orderly (2.18) by the following procedure.

The first quantized LSP frequency $\hat{w}_{1,j_1}\left(= \hat{w}_{q1}\right)$ is found by finding the first index j, such that $Q'(\frac{\hat{w}_{1,j_1} + \hat{w}_{1(j_1+1)}}{2}) < 0$.

The second quantized LSP frequency $\hat{w}_{2,j_2}\left(= \hat{w}_{p1}\right)$ starts from $\hat{w}_{2,j_2} > \hat{w}_{q1}$ and is evaluated when $P'(\frac{\hat{w}_{2,j_2} + \hat{w}_{2(j_2+1)}}{2}) < 0$.

The third quantized LSP frequency $\hat{w}_{3,j_3}\left(= \hat{w}_{q2}\right)$ is searched from $\hat{w}_{3,j_3} > \hat{w}_{p1}$ and is chosen when $Q'(\frac{\hat{w}_{3,j_3} + \hat{w}_{3(j_3+1)}}{2}) > 0$.

The fourth quantized LSP frequency $\hat{w}_{4,j_4}\left(= \hat{w}_{p2}\right)$ starts from $\hat{w}_{4,j_4} > \hat{w}_{q2}$ and is

determined when $P'(\frac{\hat{w}_{4,j_4} + \hat{w}_{4(j_4+1)}}{2}) > 0$. The procedure is repeated further till all the

LSP frequencies have been found.

Mark Prandlini, (a Ph D student at the University of N.S.W), has developed this routine efficiently by pre-storing $\hat{w}_{i,\chi}$ and $\cos\left(\frac{\hat{w}_{i,\chi} + \hat{w}_{i,\chi+1}}{2}\right)$ in two tables. However his method uses four times as much memory as table 1 (stored as short integer memory) does.

After quantizing the LPC coefficients, the indexes of each LSP frequencies are then transmitted to the receiver.

2.4.4 Conversion from LSPs to LPC coefficients

Before generating the synthesised speech from the LPC filter excited by codewords, both in the transmitter (encoder) and receiver (decoder), the LSPs have to be converted to the LPC coefficients. In the encoder, these coefficients are assigned to the vocal tract filter for the analysis-by-synthesis procedure to determine the optimum excitation (or codeword). In the decoder, the same is done to generate the synthesised speech.

The procedure of converting LSPs to LPC coefficients is the inverse procedure of quantizing LPC coefficients using LSPs. It is as below:

1) Given the index, the LSP frequencies ($f_{p1},\cdots,f_{p5}$ and $f_{q1},\cdots,f_{q5}$) can be found from the table 1.

2) With w= 2πf and eqn.(2.16), the polynomials P(z) and Q(z) can be reconstructed.

3) The inverse filter A(z) is recovered from eqn.(2.14). In this way, the coefficients of the filter are found.

Chapter 3

The Code Excited Linear Predictive (CELP) Coder

3.1 INTRODUCTION

The code excited linear predictive (CELP) algorithm falls into a general class of linear predictive (LP) coders. This class of coder operates on sampled speech on a frame by frame basis. A filter is used to describe the spectral envelope (or vocal tract) of the speech signal. The coefficients of the filter are obtained using the LP technique (Chapter 2). The excitation for the filter is determined using an analysis-by-synthesis procedure. The primary difference between the many different types of LP coders is the characteristics of their excitation values. These lead to the similarities and differences between the different algorithms.

To understand the exact formulation of the DoD 4.8 kbps CELP coder, it is important to understand the analysis-by-synthesis technique first. This introduction to the background of CELP and its excitation characteristics are necessay to prepare for the work in the next Chapter (Excitation Search).

3.2 ANALYSIS-BY-SYNTHESIS PROCEDURE

The analysis-by-synthesis technique, reported by Bishnu S. Atal in 1982 [30], was first used to determine a Multi-pulse excitation for LPC speech synthesis. The method is shown in Figure 3.1.

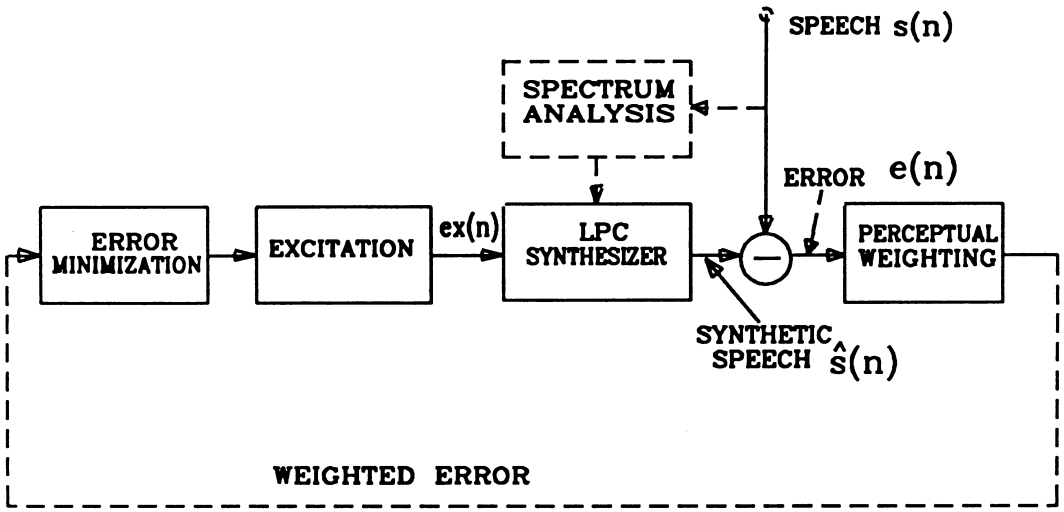


Figure 3.1: Block Diagram of an Analysis-by-synthesis Procedure for Determining the Excitation

The LPC synthesizer produces samples $\hat{s}(n)$ of synthetic speech signal in response to the excitation ex(n). The synthetic speech samples are compared with the corresponding speech samples of the original (natural) speech signal s(n) to produce an error signal e(n). This error is not very meaningful and must be modified to take into account of how the human perception treats the error. For example, if the error criterion of the Mean Square Error (MSE) of eqn. (2.6) is used to determine the excitation which produces a minimum error, high numerical accuracy of the synthetic speech will be the result; however, this is not the error criterion which minimizes the perceived distortion (B. S. Atal 1979 [27]).

To reduce the perceived error, it is advantageous to reduce the error in those frequency regions that contain less energy. This is done by a linear filter. The error is weighted to produce a subjectively meaningful measure of the difference between the signals $\hat{s}(n)$ and s(n). More detail is brought out in the next section.

3.3 GENERAL CELP MODEL

The system described in section 3.2 was later developed to a system called Code Excited-Linear Prediction (CELP) [1] or Vector Excitation Coding (VXC) [2], shown as Figure1.2 in Chapter 1. The speech synthesizer in CELP consists of two time-varying linear recursive filters. The first filter is a long-term predictor that generates the pitch periodicity of voiced speech. The second is a short-term

predictor which is used to model the slowly varying spectral envelope of the speech signal, as discused in Chapter 2. The excitation signal ex(n) was replaced by a stochastic codebook. The function of the excitation signal is to model all perceptually important features of the speech signal that are not well modelled by the short-term and long-term predictors.

3.3.1 Long-term Predictor

The long-term predictor is formed by a low order linear predictor,

$$B(z) = \sum_{k=0}^{p-1} b(k) z^{-(M+k)}$$

where the delay M respresents the pitch period (the integral number estimate of the pitch period) and b(k)'s are the predictor coefficients. The delay of the pitch predictor is of the order of the pitch period of the speech frame (2-20 ms), corresponding to pitch lags M of 16 to 160 samples. A range of 128 lags is typically used in CELP coders. For example, pitch lags vary from a minimum of 16 to a maximum of 143, or 20 to 147.

The order of the predictor p is typically 1 to 3. It is noted that the prediction error is reduced with increasing predictor order. Multiple coefficients can provide

interpolation between the samples, if the pitch delay is represented by a non-integer number of samples rather than by a sample lag. They also may represent a frequency-dependent gain factor, which is useful because most speech signals exhibit less periodicity at the high frequencies compared to the low frequencies. However, more bits are needed for encoding the additional coefficients, compared to the order one filter case in the following[28].

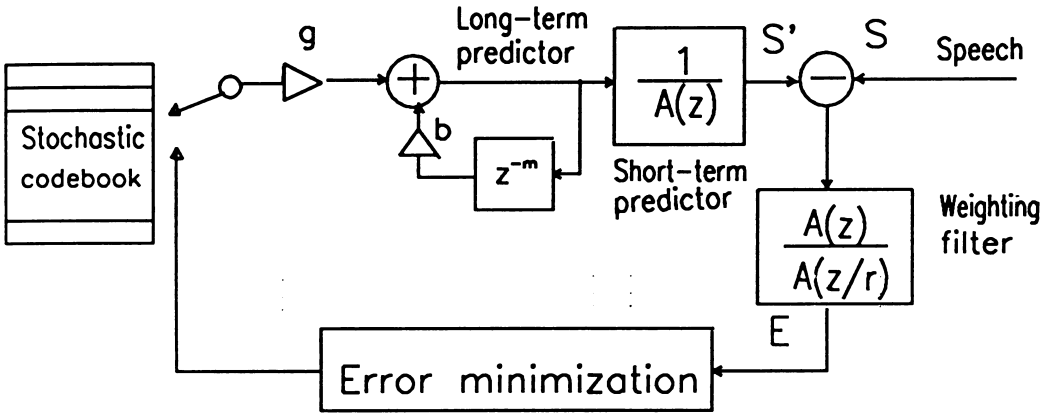


Figure 3.2. Original CELP

Figure 3.2 indicates an order one long-term predictor case. It is of the form

$$B(z) = bz^{-M}$$

where b is named the pitch filter gain factor.

Before 1985 the long-term predictor was mostly determined by analyzing the original speech only (open loop form). Later on the optimal long-term predictor was determined with an analysis-by-synthesis procedure (closed-loop form). Peter Kroon [17] and Richard Rose [32] both reported that there was significant audible improvement obtained by using the closed loop long-term predictor over the open loop predictor for CELP coders.

3.3.2 Excitation

In the CELP algorithm the excitation signals contain an ensemble of white, Gaussian random sequences, $ex_I(n)$, where I=1,...,F; F is the size of the ensemble; n = 0,...,N-1, and N is the excitation length. Each sequence is specified by an ensemble index I. Since speech has a large dynamic range, it is advantageous to scale all sequences by an optimum scaling factor (or a code vector gain) g. The object of the analysis-by-synthesis procedure is to determine the excitation ensemble index (I) and gain g_I for each of the excitation sequences $ex_I(n)$ through the error minimization estimate. The approach involves an exhaustive search of each the excitation ensembles. The search procedure is a sub-optimum, sequential method that finds the parameters (I,g_I) associated with each component excitation sequence.

Figure 3.2 is a block diagram of the analysis procedure. The analysis procedure minimizes the energy in the weighted error signal E through the proper choice of ensemble sequence $ex_I(n)$. The spectral error weighting filter W(z) that is used here was developed by Atal and Schroeder [27] and is dependent on the short-time spectral envelope of the speech. This weighting filter has the effect of concentrating coding noise energy in the formant regions of the spectral envelop. The weighting filter is defined by

$$W(z) = \frac{1 - \sum_{k=1}^{p} a_k z^{-k}}{1 - \sum_{k=1}^{p} a_k \alpha^k z^{-k}} \qquad (3.1)$$

where the a_k are the coefficients of the envelope filter. Values of α in the range $0.6 < \alpha < 0.9$ were found to give similar subjective result in informal listening tests. A value $\alpha = 0.8$ is always used in CELP coders.

Let x(n) be the weighted original speech after removing the memory contribution of the pitch synthesis and weighting filters from previous frames, and h(n) be the impulse response of the filter $(1/A(z))(A(z)/A(z/\alpha)) = 1/A(z/\alpha)$. $y_I(n)$ is the weighted response to the excitation corresponding to that I :

$$y_I(n) = h(n) * ex_I(n).$$

Then the mean squared weighted error (MSWE) between the original and synthesised speech is given by:

$$E = \sum_{n=0}^{N-1} [x(n) - g_I y_I(n)]^2 \tag{3.2}$$

Setting $\partial E/\partial g_I = 0$ leads to the relation:

$$g_I = \frac{\sum_{n=0}^{N-1} y_I(n)x(n)}{\sum_{n=0}^{N-1} y_I(n)^2} \tag{3.3}$$

The corresponding MSWE is then given by

$$E = \sum_{n=0}^{N-1} x(n)^2 - g_I \sum_{n=0}^{N-1} y_I(n)x(n) \tag{3.4}$$

Since this expression is minimized by maximizing the second term on the right hand side of the expression, the optimum I corresponds to that excitation function that produces the synthetic speech y_I (n) that is the most highly correlated with the original weighted speech x(n). Also, the computational complexity is determined by the number of operations needed to evaluate this second term for all the codebook entries.

In Atal and Schroeder's original design [1], the codebook was generated from a zero-mean unit-variance white Gaussian sequence where each codeword (or

sequence) consisted of an independent segment of this sequence. This provides good speech quality at intermediate bit rates, but the vector quantization of the independent excitation signal required an extremely high level of computation. The computation of all $y_I(n)$ sequences in an ensemble requires an order of (N+L)LF operations per source analysis frame, where L is the order of the impulse response h(n). In addition, the pitch predictor was calculated by a brute force calculation using the filter approach that also involved huge computation [13]. The complexity in the original CELP coder was thus far too high for real-time implementation.

3.4 DEVELOPMENT OF CELP CODERS

With contributions from Lin [9], Davidson and Gersho [2], some CELP-type coders started to employ sparse and efficient pseudostochastic block codes for excitation, in which the codeword is centre clipped with a high proportion of the samples being zero and the adjacent codewords in an stochastic codebook are non-independent.

The DoD CELP coder version 2.3 [13], which was originally written by Peter Kroon of AT&T [17], and later developed at the U.S. Department of Defence, is a development of the CELP coder. It has the same flowchart as shown in Fig. 3.2, but the excitation in this system comes from a special form of code book containing samples of a zero-mean, unit-variance, white Gaussian sequence which

is centre clipped at level 1.2, resulting in approximately 75% sparsity (zero values). Each code word contains one new sample and all but ***one*** sample of the previous codeword. This sparse, overlapped code book is used to compute fast convolutions and energies by exploiting the recursive **end-point correction** algorithm (defined in ref.[13] and stated in Chapter 4).

As shown in [13], this code book reduces computation by an order of magnitude from 66 to 6 million instructions (multiplies and adds) per second (6 MIPS) for a codebook with 256 codeword. The long-term prediction is achieved using closed-loop analysis. The pitch calculation using the filter approach can be replaced by an end-correction algorithm since the pitch filter memory also shows an overlapped property. The computation for pitch calculation is reduced from 33 to 3 MIPS. When using overlapped versus independent codebooks, the difference in synthesised speech is virtually unnoticeable, and the reduction in segment signal-to-noise ratio is less than a fraction of a decibel.

Many other new CELP-type algorithms have also been implemented to enhance performance with less computation and memory. Some well-known coders like the Stochastically Excited LPC (or SELP),[5] [12], Vector Sum Excited Linear Prediction (or VSELP) [11], and Self Excited Vocoder (or SEV) [15] have made significant contributions to this class of speech coders. For example, SEV, SELP and VSELP use adaptive codebook search techniques for their pitch search,

and VSELP employs two small stochastical codebook structures.

According to the survey in [8], DoD CELP version 2.3, SELP and VSELP are the most promising coding algorithms. They all show very high performance in the following factors:

- Intelligibility (measured by DRT score,Diagnostic Rhyme Test)
- Acceptability (measured by DAM score, Diagnostic Acceptability Measure)
- Robustness; Runtime; Coding delay; Error tolerance; Algorithm expandability

...

In order to build a Proposed Federal Standard (in early 1989), the U.S. Government and AT&T defined a combined algorithm based on the DoD CELP version 2.3 and SELP programs but excluded those SELP features thus are proprietary to CELP. The combined algorithm is described in [7] (called DoD version 3.0). It uses the frame rates and parameter coding methods from CELP (which means the spectrum analysis of LPC is the same); the pitch excitation search from SELP and SEV [15], which contains 128 adaptive vectors; and the code structure of DoD's CELP and SELP. The code book contains samples of a zero-mean, unit-variance, white Gaussian sequence centre clipped at 1.2, resulting in approximately 75% sparsity (zero values). Each code word contains ***two*** new samples and all but two samples of the previous code word. The block diagram for the synthesis procedure is shown in the following Fig. 3.3a.

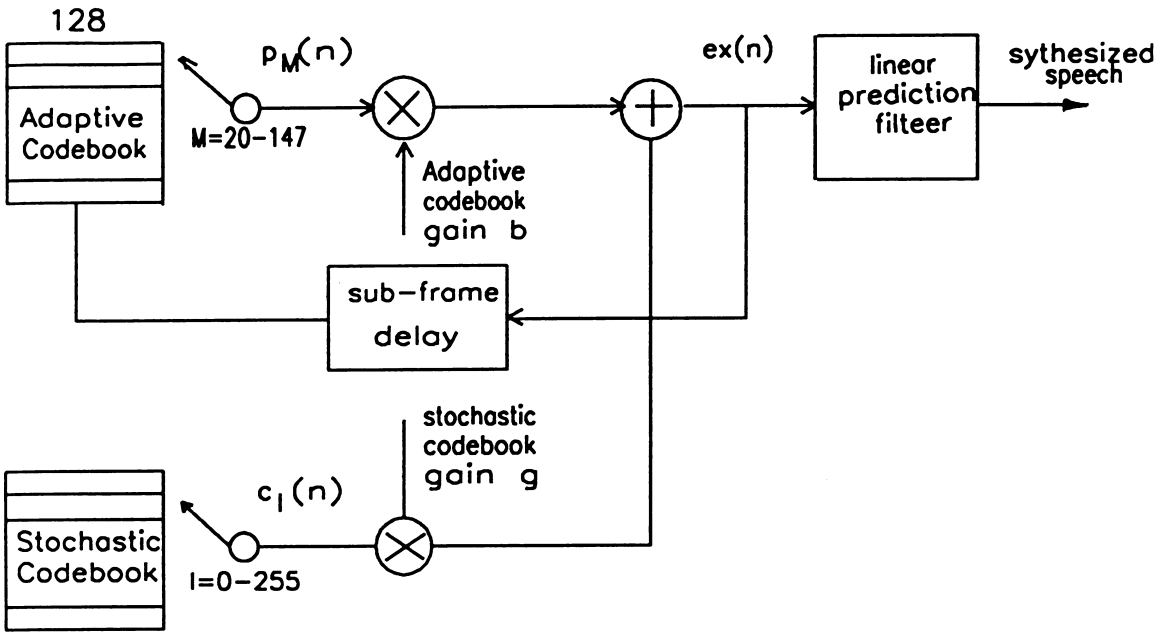


Figure 3.3a: DoD CELP Version 3.0

CELP version 3.0 has two major changes compared to version 2.3. Firstly the pitch predictor filter is replaced by an adaptive codebook operating as excitation vectors. Secondly the stochastic codebook uses a shift-two non-independent sparse Gaussian clipped codebook. References [12], [15] and [32] give the reasons for using an adaptive code book. Indeed, there is much similarity between the behaviours of the long-term predictor and of pulse excitation, but adaptive codewords working as excitation vectors make the coding system more compact. Also it has been proved by Kleijn [3] that the use of a shift 2 codebook will result in performance identical with that of a fully independent codebook.

As reported in [7], CELP version 3.0 is a revolutionary code and outperforms all U.S. Government standards operating at rates below 16,000 bps, and is robust in acoustic noise, channel errors, and tandem connections. However, in November 1989, after a short development period, anthor new standard: the Proposed Federal Standard 1016 (or DoD CELP version 3.1 [4]) was launched. This is described in the next section.

3.5 DoD CELP VERSION 3.1

DoD CELP version 3.1 is based on an enhanced version of the code selected in the survey (version 3.0). These enhancements maintain or improve the coder's speech intelligibility and quality, such as outperforming version 3.0 by 3 DAM points and producing identical DRT scores, channel robustness. The major enhancements to version 3.0 are changes in the code books, as shown in Fig 3.3b. The stochastic code book is ternary value (-1,0,+1), as suggested by Dan Lin [9],and half as large (512 codewords). The adaptive code book is twice as large (256 codewords) and contains 128 noninteger delays, (as suggested by Peter Kroon [16],) in addition to the original 128 integer delays.

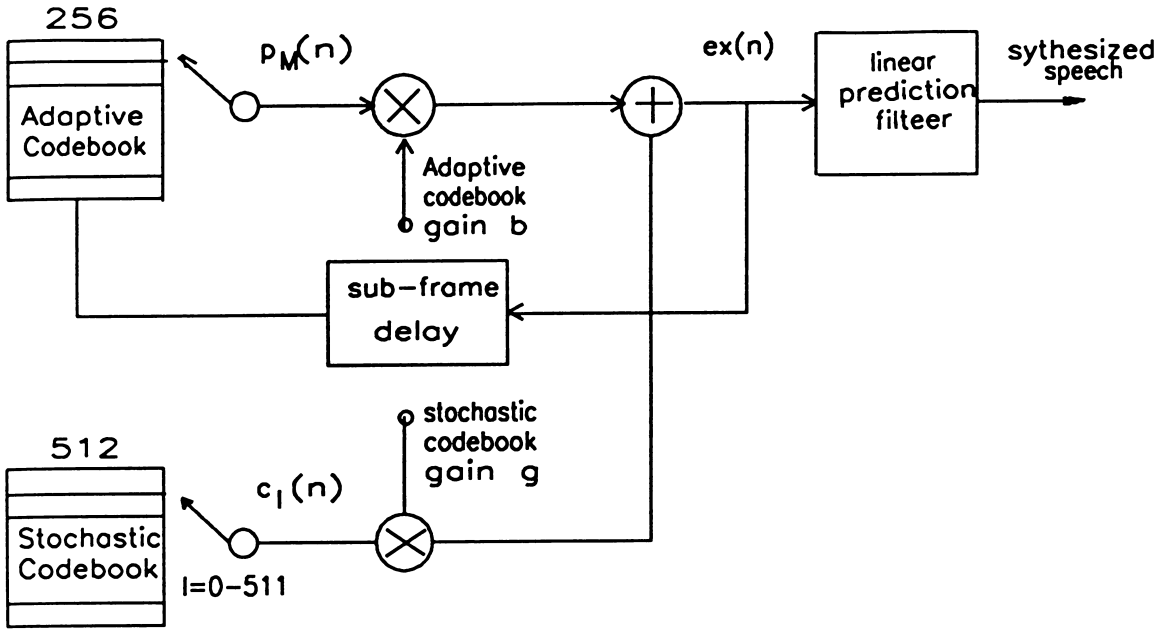


Figure3.3b : DoD CELP Version 3.1

DoD CELP Version 3.1 uses several techniques to reduce the complexity and memory but maintaining high quality speech. The main characteristics of the coder are:

(1) Frame rate 30 ms (240 samples)

(2) Four subframes(7.5 ms each, 60 samples)

(3) 147-element adaptive codebook with

- Closed loop analysis
- 128 integer delay codes (or vectors) overlapped by 1
- 128 implicit noninteger delay codes are optionally available

- Even/odd subframe delta search method
- submultiples of the delay search method

(4) 1082-element codebook with

- 512 codes (or vectors) overlapped by 2
- ternary value samples (-1,0,or +1)
- sparse structure (77% of elements zero)

The standard also specifies an error protection scheme utilizing a forward error-correcting Hamming code and parameter smoothing.

As usual, the major computational parts of the algorithm are the pitch search and the codebook search. Both of these are performed four times per frame. An important technique to reduce the computations is the end-correction convolution technique (recursive convolution and recursive energy calculation). This method reduces the number of multiply-adds by a factor of 18 to 20 relative to brute force techniques [7].

In addition, the codebook is designed to have approximately 77% of the samples equal to zero and the codebook is ternary valued (-1,0,+1). This allows many of the convolution updates in the codebook search to be reduced to a simple shift of a vector sample [2], and eliminates multiplications.

To reduce complexity further, the pitch search is limited in range. During every odd-numbered subframe, the optimum pitch search is performed over the range 256 (128 integer delays from 20 to 147 and 128 non-integer delays). On the even subframes, the search is only over the range 64 lags relative to the previous subframe. Also, before calling the search loop ,the minimum square power error (or MSPE) search criterion is modified to check the match score (defined in Chapter 4) at submultiples of the delay to determine if it is within 1/2 dB (ie 12 percent) of the MSPE. The shortest submultiple delay is selected if its match score satisfies the modified criteria. While maintaining high quality speech, this results in a smooth "pitch" delay contour that is crucial to delta coding and the receiver's smoother in the presence of bit errors.

More detail of the process will be found in Chapter 4.

Chapter 4

Excitation Search

4.1 INTRODUCTION

DoD synthetic speech is based on a source-filter synthesis model as illustrated in Fig.3.3b. The source (or excitation) combines two codes (or vectors):

- pitch vector from the adaptive codebook.
- code vector from the stochastic codebook.

In the synthesis procedure, a perceptually weighted mean-square error measure approach is used to select the appropriate excitation. Since the excitation search becomes a major computational part of the CELP system, various efforts in CELP version 3.1 have been made to reduce this computational complexity. For example, the use of a special codebook structure and the replacement of the long-term predictor (pitch filter in Fig.3.2) by an adaptive codebook have been used for this purpose. These will be discussed in turn.

Section 4.2 derives a code search model for the single vector excitation case. Section 4.3 discusses different joint vector search methods for a combined excitation. Sections 4.4 and 4.5 explain the adaptive code book and stochastic code book (ternary code book), respectively.

4.2 CODEBOOK SEARCH MODEL

Figure 4.1 is the model used for the derivation of the search procedure. **v**,**s**,$\hat{\mathbf{s}}^{(k)}$ and **e** are the L-dimensional row vectors, representing the excitation signal, the original speech signal, the synthetic speech signal and the weighted error signal, respectively.

H and **W** are L x L matrices whose j-th rows contain the truncated impulse response caused by a unit impulse δ(t-j) of the speech-synthesis (or prediction) filter (1/A(z)) and error weighting filter (A(z)/A(z/r)), respectively. $\hat{\mathbf{s}}^{(0)}$ is the output of the synthesis filter due to memory hangover from the previous synthesis interval (conventionally called zero input response of the synthesis filter).

In the general case, due to speech having a large dynamic range, it is advantageous to scale all codebook entries by an optimum scaling factor b_k before the evaluation of the error criterion. So the excitation vector **v** is a gain shaped vector $\mathbf{v}^{(k)}$, where **k** indicates the index of the codebook. $\mathbf{v}^{(k)}$ can be written as:

$$\boldsymbol{v}^{(k)} = b_k \boldsymbol{x}^{(k)} \tag{4.1}$$

where $\mathbf{x}^{(k)}$ is a code book vector.

The synthetic signal produced by $\mathbf{v}^{(k)}$ can be expressed as:

$$\hat{\boldsymbol{s}}^{(k)} = \hat{\boldsymbol{s}}^{(0)} + \boldsymbol{v}^{(k)} H \tag{4.2}$$

The error signal $\boldsymbol{e}^{(k)}$ is given by:

$$\boldsymbol{e}^{(k)} = (\boldsymbol{s} - \hat{\boldsymbol{s}}^{(k)}) W \tag{4.3}$$

Thus:

$$\boldsymbol{e}^{(k)} = \boldsymbol{e}^{(0)} - b_k \boldsymbol{y}^{(k)} \tag{4.4}$$

where

$$\boldsymbol{e}^{(0)} = (\boldsymbol{s} - \hat{\boldsymbol{s}}^{(0)}) W \tag{4.5}$$

and

$$\boldsymbol{y}^{(k)} = \boldsymbol{x}^{(k)} HW \tag{4.6}$$

Here, $\mathbf{y}^{(k)}$ is the residual weighted response of codeword (also called vector shape in the next section). Equation (4.4) then leads to the new model in Figure 4.2.

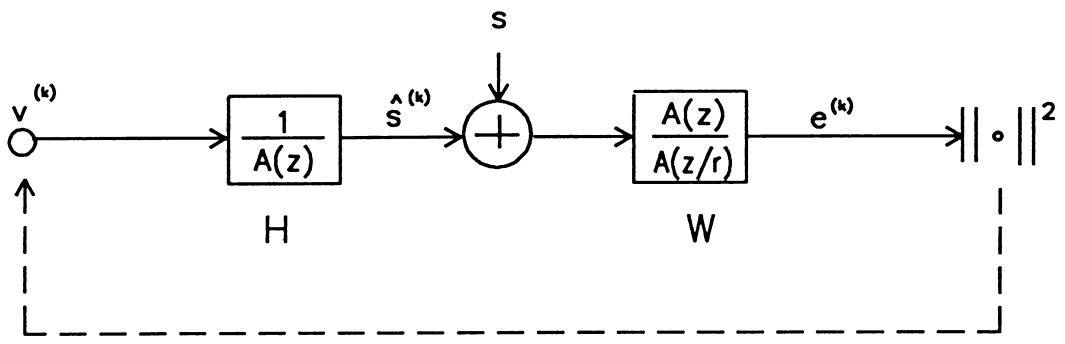


Figure 4.1: Code search model 1

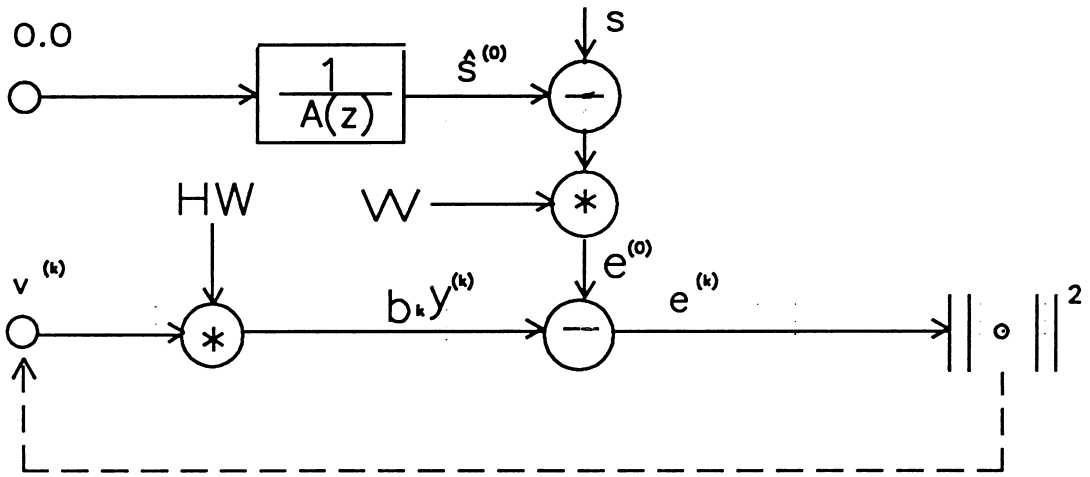


Figure 4.2: Code search model 2

The total squared errors:

$$E_k = \| e^{(k)} \|^2 = e^{(k)} e^{(k)T} \tag{4.7}$$

$$= e^{(0)} e^{(0)T} - 2b_k e^{(0)} y^{(k)T} + b_k^2 y^{(k)} y^{(k)T}$$

E_k is a function of both the gain factor b_k and the index k. For a given value of k the gain can be computed by setting the derivative of E_k with respect to the unknown gain value to zero($\frac{\partial E_k}{\partial b_k} = 0$). Therefore, the optimum gain is the ratio of the correlation of the residual and weighted response of the codeword to the energy of the weighted response of the codeword:

$$b_k = \frac{\boldsymbol{e}^{(0)}\boldsymbol{y}^{(k)T}}{\boldsymbol{y}^{(k)}\boldsymbol{y}^{(k)T}} \tag{4.8}$$

The gain is quantized to jointly optimize thr gain and index k:

$$\hat{b}_k = Q[b_k]$$

In Eq.(4.7), the first term $\boldsymbol{e}^{(0)}\boldsymbol{e}^{(0)T}$ is independent of the code book index, k, and therefore can be ignored. Minimizing E_k with respect to k is equivalent to maximizing the negative of the last two terms, which is called the "match score":

$$match_k = \hat{b}_k(2\boldsymbol{e}^{(0)}\boldsymbol{y}^{(k)T} - \hat{b}_k\boldsymbol{y}^{(k)}\boldsymbol{y}^{(k)T}) \tag{4.9}$$

If the gain quantization is ignored, Eq.(4.8) can be substituted in Eq.(4.9), and the match score is the ratio of the squared correlation to the energy:

$$match_k = \frac{(\boldsymbol{e}^{(0)}\boldsymbol{y}^{(k)T})^2}{\boldsymbol{y}^{(k)}\boldsymbol{y}^{(k)T}} \tag{4.10}$$

Thus, the code book search procedure is to find the codeword,k, which maximizes the match score, ($match_k$). The optimum excitation vector is entirely characterized by the index k and the corresponding gain factor b_k.

4.3 COMBINED EXCITATION SEARCH PROCEDURE

The excitation that combines the gain shape adaptive code and the gain shape stochastic code (Figure 3.3b), can be written as:

$$e = bp_M + gc_I \tag{4.11}$$

Let the original speech residual be $X = e^{(0)}$, (also named codebook search target [3][4]). Then the synthesis residual X' is the combined excitation e filtered by HW. Because HW is a linear filter, X' can be expressed as:

$$X' = bP + gC \tag{4.12}$$

where P and C are the pitch vector residual response ($P=p_M HW$) and the stochastic code vector residual response ($C=c_I HW$). The gain (b,g) of each vector is calculated by expressions (4.14) and (4.15), which minimize the error power ($|E|^2$) in expression (4.13).

$$| E |^2 = | X - bP - gC |^2 \rightarrow \min \qquad (4.13)$$

Seting the derivatives $\partial | E |^2 / \partial b$ and $\partial | E |^2 / \partial g$ to zero gives the optimum gains

$$b = \{(C,C)(X,P) - (C,P)(X,C)\}/\Delta \qquad (4.14)$$

and

$$g = \{(P,P)(X,C) - (P,C)(X,P)\}/\Delta \qquad (4.15)$$

where

$$\Delta = (P,P)(C,C) - (P,C)(C,P)$$

If this process is executed for ***all possible*** combinations of the adaptive and stochastic codebooks and the two vector gain (P,C,b,g), the **lowest** possible error power in expression (4.13) will be determined, and the optimum two-vector synthetic excitation signal (**X'**) can be calculated. However, searching through every possible combination of adaptive and stochastic code vectors results in an enormous amount of computation. Because of this, the two excitation vectors are selected sequentially. The pitch vector is usually selected first using expressions (4.16) and (4.17)(assuming g is zero), and then the pitch response (P) is considered fixed while the stochastic codebook is searched using expressions (4.13),(4.14) and (4.15). This gives *joint* optimization of the two vector gains and the codebook response (b,C,g). However, the joint optimization procedure still needs huge computation.

To force

$$| E' |^2 = | X - b'P |^2 \rightarrow \min \qquad (4.16)$$

leads to the optimum gain

$$b' = (X,P)/(P,P) \qquad (4.17)$$

This joint optimization can be achieved by orthogonalizing the code response vector (C) to the pitch response vector (P) using the Gram-Schmidt method [10]. Define:

$$P = x_{11}\Phi \tag{4.18}$$

and

$$C = x_{21}\Phi + U_n. \tag{4.19}$$

Where Φ is a unit vector and U_n is the orthogonalized shape vector. Then

$$x_{11}^2 = (P,P) = \Gamma$$

and

$$x_{21} = (C,\Phi) = (C,P)/\Gamma^{\frac{1}{2}}$$

Therefore

$$U_n = C - x_{21}\Phi \tag{4.20}$$

$$= C - \frac{(C,P)}{\Gamma}P$$

The search procedure is now needed to find the optimum stochastic code index I (corresponding to a response vector C) by searching for the minimization of

$$|E''|^2=|X-g'U_n|^2 \tag{4.21}$$

Finally the joint optimum gain (b,g) is obtained by (4.14) and (4.15).

The reason for using this algorithm is that a speech signal or its residual X consists of two parts, namely, voiced (V) and unvoiced (V_n) (or the predictable part of speech and the unpredictable part of the speech). They are uncorrelated or approximately orthogonal to each other (Figure 4.3a), and the synthetic speech residual X' is a combination of two gain response vectors bP and gC (Figure 4.3b).

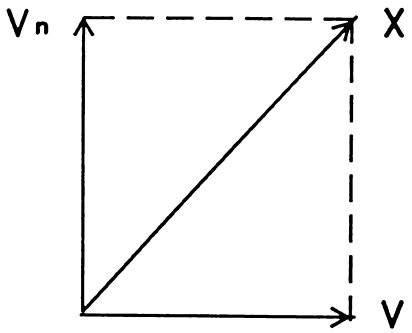


Figure 4.3a

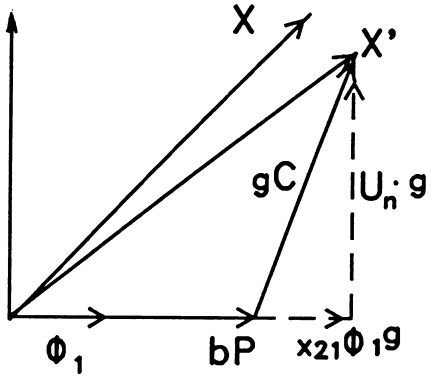


Figure 4.3b

If the pitch response vector is thought as contributing to the voice region $[PX = P(V + V_n) = PV]$, then the correlation between X and X' can be expressed as:

$$X'X = (bP + gC)X \tag{4.22}$$
$$= bPX + gCX$$
$$= bPV + g(x_{21}\Phi)V + gU_nV_n$$

Equation (4.22) shows that the correlation is the sum of three terms. Indeed, eqn.(4.16) is used to calculate the maximum correlation between pitch response P and residual speech (or its voiced part V) since PV = PX. Eqn.(4.21) determines the best response vector C that contributes to the unvoiced region (since $U_nV_n = U_nX$). After using (4.14) and (4.15) to optimize the gains (b,g), the synthesis residual X' from eqn.(4.12) will maximally correlate with X in eqn.(4.22).

The simulation of this work shows that this method produces high quality synthetic speech. Unfortunately, U_N needs to be evaluated eqn.(4.20) before the evaluation of the error criterion eqn.(4.21). Computation of eqn.(4.20) requires 2N IPS (where N = 60 is of subframe sample length). For a 512 codeword subframe , 512 * 2N (= 61,440)IPS are needed for one frame search. To avoid the problem, version 3.1 DoD celp code search does it in an efficient way. It is briefly described below:

1). let g = 0

2). determine b' and P by (4.16) and (4.17), and let b = b'

3). calculate the stochastic code book search target X_c (or residual speech after removal of the signal periodicity)

$$X_c = X - bP \tag{4.23}$$

4). determine g and C by (4.13)

$$\text{or } |E|^2 = |X_c - gC|^2 \rightarrow \min, \tag{4.24}$$

$$\text{giving } g = (X_c, C)/(C, C) \tag{4.25}$$

Experimental results have shown that the DoD CELP produces almost toll quality speech and that there is no perceptual difference between the algorithm using either the orthogonal method or the DoD method.

In fact, the value b' in (4.17) and value b in (4.14) are very close. After quantization, the difference between b and b' can be ignored. Secondly, the value of the first term in eqn.(4.22) is much larger than the sum of the other terms. This result has also been proved by Kroon and Atal [10]: the contribution of the pitch filter (meaning the adaptive code book here) to the final SNR of the reconstructed speech segment is significant even for non-periodic signals, and is over 50% of the final SNR, while the contribution of the code book is about 35%. Sometimes it is very small. Varying the stochastic codebook vector doesn't significantly affect overall SNR . Therefore, it is reasonable to search a stochastic code book that just

has maximum correlation with the target signal X_c (eqn.(4.23)).

In the DoD method, the search processes for the pitch and the code book are the same. Only the target vector (residual vector) is different. The target vector for the codebook search is the pitch target minus gain shape response bP shown in equation (4.23). This also leads to the coding system becoming more compact than the one using pitch for its long-term predictor, since an efficient search can operate on both the adaptive and the stochastic codebooks [12].

4.4 ADAPTIVE CODE BOOK AND PITCH SEARCH

4.4.1 Adaptive code book with integer and non-integer delays

An adaptive book, based on Rose's self excited coder [15], was first introduced by Kleijn [3]. It is advantageous to consider the speech signal (from a coding viewpoint) in the selection of a codebook for the vector quantization. This naturally leads to a two-stage vector quantization, where the first stage attempts to remove the periodicity from the signal, while the second stage attempts to match the remaining signal, which will be more random in nature.

Rose [32] indicates that there is a great deal of similarity between the behaviour of the long-term predictor and a pulse excitation. The gain value assigned

to a pulse depends on the degree to which the weighted response to an impulse in the present source analysis frame is correlated with the weighted speech [30]. Similarly, the gain of the long-term predictor reflects the degree to which the response to a past residual sequence is correlated with the weighted speech. Therefore, the past residual sequence can be just called as an excitation vector. It suggests some sort of codebook that contains past excitation vectors (or sequences) is required.

An adaptive codebook containing past excitation vectors is readily constructed. The most recently selected excitation vectors are concatenated to form a finite excitation history. In a voiced section of speech, which typically displays a high level of periodicity, the selected candidate vectors will generally be an integer number (e.g. 20-147) of pitch periods removed from the present frame. In unvoiced speech the adaptive codebook will effectively contain a set of overlapping random sequences. The DoD CELP Pitch search is performed by closed-loop analysis using a modification of the adaptive code book. This method was found to be superior to the conventional "filter approach" (or long term predictor) [13], especially for high pitched speakers (typical male and child speakers)[7].

Pitch predictors play an important role in CELP coders [10]. The temporal resolution of the delay r (integer number 20-147) is mainly determined by sampling frequency. However, for efficient coding, one would like to represent the pitch

predictor by a delay with arbitrary temporal resolution (i.e. noninteger delay) and a single predictor coefficient. The increased temporal resolution produces smoother contours for the pitch delay as a function of time and increases the prediction gain. Normally, CELP coders produce more perceivable distortions for female speakers than for male speakers. A noninteger delay pitch predictor makes it possible to use a higher resolution for the shorter delays than for the longer delays, thereby increasing the performance for female speakers relatively more than for male speakers. Noninteger delays also provide benefits by reducing the following: reverberant distortion; noise, because improved pitch prediction reduces the noisy stochastic excitation component; and pitch doubling and tripling, which improve delta coding [5]. These arguments led DoD to add another 128 non-integer delay codewords to the 4.8 kbps pitch excitation codebook.

It is also advantageous that there are no extra bits needed for a 256 codeword adaptive codebook search procedure. For every odd subframe, 8 bits are used for 128 integer and 128 non-integer delay search. For every even subframe, 6 bits are needed for delta search delays ranging up to 64. The average number of bits are just 7. This avoids the problem of the high order long-term predictor as mentioned in the last chapter. Moreover, noninteger values of the delay can be obtained without increasing the 8 kHz sample rate by using a 7-point Hamming windowed sinc resampling function [5].

Because the project followed the first draft report from DoD, in which the noninteger delays were only recommended to be optional, the implementation of DoD CELP version 3.1 lacks noninteger delays. Therefore, only integer delay is considered in this thesis.

4.4.2 Integer Delay Search

The Adaptive Code Book is just a length 147 shifting storage register [11]. It is updated at the end of each subframe. The optimum combined excitation has been determined. Suppose the combined excitation (Figure 4.4) is

$$e_x(n) = b\,p_M(n) + g\,c_I(n) \quad for\, 0 \le n \le 59; \tag{4.26}$$

where M and I are pitch lag and stochastic code index, respectively, with $20 \le M \le 147$ and $0 \le I \le 511$.

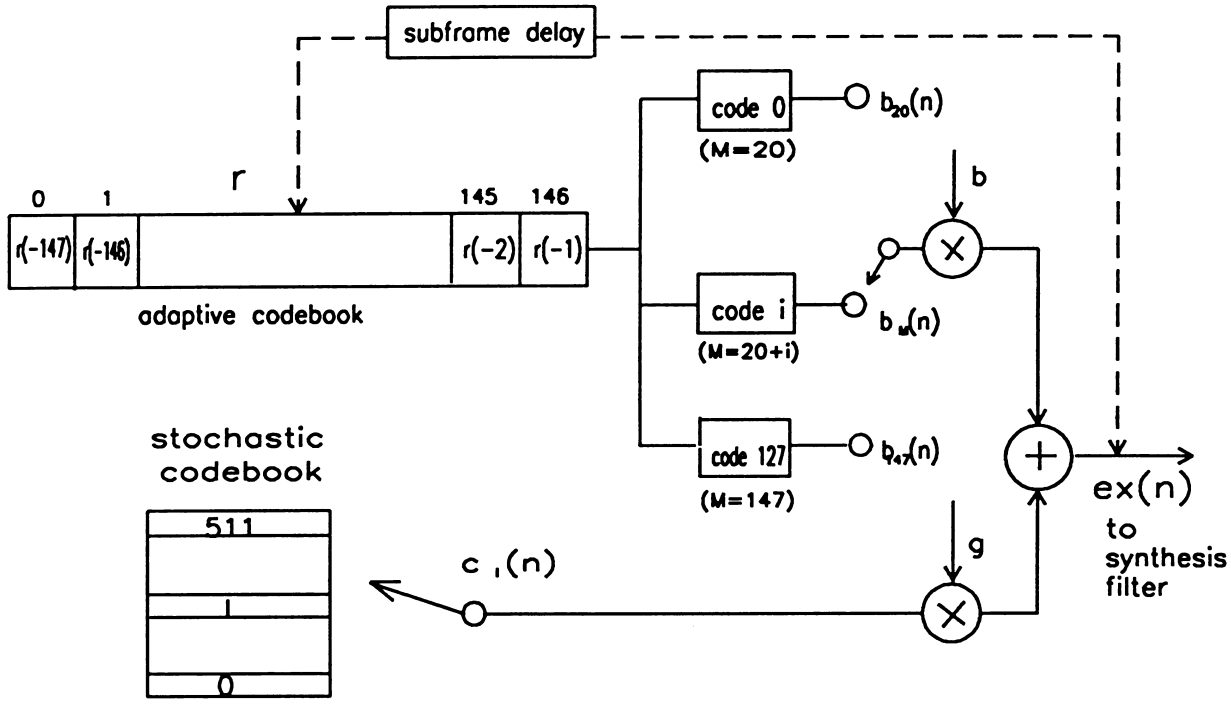


Figure 4.4: Adaptive Codes

The Adaptive Code Book memory is

$$r = r(-147), r(-146), \ldots, r(-1) \tag{4.27}$$

This 147 element linear array is updated by:

$$r(n) = r(n+60) \quad -146 \le n \le -61 \tag{4.28}$$

$$r(n) = e_x(n+60) \quad -60 \le n \le -1 \tag{4.29}$$

The Adaptive Code Book stores the history of the excitation. Element ordering is such that the first excitation elements going into the Linear Prediction Filter are the first going into the Adaptive Code Book.

During each subframe code search the Adaptive Code Book generates 128 candidate codewords (or pitch excitation vector)(p_M) corresponding to the pitch lag M. p_M is constructed by the previous excitation history, r, delayed by M sample.

$$p_M(n) = r(n - [(n+M)/M]M) \qquad 0 \le n \le 59 \qquad (4.30)$$

where [X] is the interger part of X (i.e. the largest integer not exceeding X).

Hence

p_{20}=r(-20),...,r(-1),r(-20),...,r(-1),r(-20),...,r(-1);

p_{21}=r(-21),...,r(-1),r(-21),...,r(-1),r(-21),...,r(-3);

: :

: :

p_{59}=r(-59),r(-58),...,r(-1),r(-59);

p_{60} = r(-60),...,r(-1);

p_{61} = r(-61),...,r(-2);

: :

: :

p_{147} = r(-147),...,r(-87);

This illustrates that :

* for M < 60,the p_M array repeats with a period of length M

* the neighbouring codeword is shifted by one position.

Because of this property, the pitch weighted response (convolution) can be done by using the end-correction technique [13] with recursive convolution and recursive energy, to reduce the computations.

Defining:

$$z_M(n) = \sum_{i=0}^{\min(n,M-1)} r(i-M)h(n-i), \tag{4.31}$$

the response of the codeword p_M is evaluated from:

$$P_M(n) = \sum_{i=0}^{\left[\frac{n}{M}\right]} z_M(n-iM) \tag{4.32}$$

Equation (4.31) shows that efficient computation can occur when M<60.

In this case z_M can be expressed as:

$$z_M(n) = \begin{Bmatrix} h_0 & \cdots & . & \\ h_1 & h_0 & \cdots & \\ \cdots & & & \\ h_{M-1} & \cdots & . & h_0 \\ h_M & \cdots & \cdots & h_1 \\ \cdots & & & \\ h_{59} & \cdots & . & h_{60-M} \end{Bmatrix} \begin{Bmatrix} r_{(-M)} \\ . \\ . \\ . \\ r_{(-1)} \end{Bmatrix} \tag{4.33}$$

Only M (≤ 60) codeword samples are used in (4.33), the length of the impulse response is also limited to M.

Another advantage is that $z_M(n)$ can be computed from $z_{M-1}(n)$ in a recursive manner using (4.34) and (4.35):

$$z_M(n) = z_{M-1}(n-1) + r(-M)h(n) \tag{4.34}$$

for ($1 \leq n \leq 59$) and

$$z_M(0) = r(-M)h(0) \tag{4.35}$$

This means (both in even and odd subframe process) that equation (4.31) is only used once prior to the whole search procedure, the rest of the pitch response vectors P_M are then updated using (4.32)- (4.35). Moreover, if $M \geq 60$ equation

(4.32) reduces to a simple form:

$$P_M(n) = z_M(n) \tag{4.36}$$

As mentioned in the last chapter the efficient pitch search procedure also includes both the pitch delta search technique and the MSPE technique.
The delta search is arranged so that for every odd subframe, the coding consists of 128 integer delays ranging from 20 to 147. For every even subframe, the delay is delta searched and coded with a 5 bit (32 lags) offset relative to the previous subframe. (If 128 noninteger delays are added in, both (even and odd) subframes need coding with one extra bit [4].)

The MSPE search criteria is modified to check the match score at submultiples of the delay and is selected if its match score satisfies the modified criteria. Once a submultiple of the delay is selected the pitch search can be terminated. All these techniques result in a smooth pitch delay contour that is crucial to delta coding and the receiver's smoother in the presence of bit errors [5].

4.5 TERNARY CODEBOOK AND THE CODE SEARCH

Speech coding using efficient pseudo-stochastic block codes was first introduced by Daniel Lin [9]. The pseudo-stochastic code is refered to as stochastically populated block codes in which the adjacent codewords in an

innovative codebook are non-independent. Typical examples of the pseudo-stochastic code are the Gaussian sparse shift-one (or shift-two) codebook as described in the last Chapter. From this code book, Daniel Lin constructed a codebook of ternary-valued innovation codes with values of -1, 0 and 1 [31]. The non-zero samples are set at +1 or -1 depending on the Gaussian codebook signs. Thus the ternary codebook has a high percentage of zeros (since the Gaussian codebook is sparse, i.e. 75% of the elements are zero).

Lin also proves that the sparse Gaussian excitation vector codebook and the ternary codebook are equivalent to the previously proposed Gaussian random codebooks (independent codebook) in terms of coding performance. The use of ternary codes means that all the information relating to the excitation codebook can be stored in a smaller amount of memory [14]. This approach also permits a very fast codebook search and results in no loss of objective or subjective performance.

The Ternary Code Book is presented in Appendix 1. There are 1082 element ternary values, which are assembled into 512 fixed codes of length 60 which are overlapped and shifted by two elements, as shown in Figure 4.5.

Index	*Code Book Elements In Codes*	
511	0,1,2,...	...,58,59
510	2,3,4,...	...,60,61
:	:	
.	.	
n	2(511-n), 2(511-n)+1,... , 2(511-n)+59	
:	:	
.	.	
1	1020, 1021, 1022,...	,1078, 1079
0	1022, 1023, 1024,...	,1080, 1081

Figure 4.5: structure of overlapped codebook

This specially structured codebook greatly reduces the computational load. The biggest savings come from the elimination of **all** but **one** of the convolutions for each subframe. It also permits a recursive convolution computation. The first codebook vector is convolved normally with the weighting synthesis filter. Subsequent convolutions, however, make use of the following relationships:

$$v_{i+1}(n) = \hat{u}_i(n-1) + x_{i+1}[1]\,h(n) \tag{4.37}$$

$$\hat{u}_{i+1}(n) = v_{i+1}(n-1) + x_{i+1}[0]\,h(n) \tag{4.38}$$

where $\hat{u}_i(n)$ is the generated residual from code c_i,shown in (4.39) and its next neighbour c_{i+1} in (4.40).

$$c_i = x_i[0], x_i[1], \ldots, x_i[59] \tag{4.39}$$

$$c_{i+1} = x_{i+1}[0], x_{i+1}[1], x_i[0], \ldots, x_i[57] \tag{4.40}$$

Equation (4.37) and (4.38) have the same forms as (4.34). These are the end-correction method. Indeed, since each of these equations is a shift-one process, we can write the shift function, and then just call the function twice to implement (4.37) and (4.38).

The convolution employing the next vector can be found with only 120(2×60) multiplies and adds. With the ternary codebook, this number can be further reduced.

If a shifting sample x is equal to zero, a shift-one process is :

$$u_{i+1}(n) = u_i(n-1)$$

If a shifting sample x = ±1, the process is then

$$u_{i+1}(n) = u_i(n-1) \pm h(n)$$

Evidently, ternary vectors can save all the multiplications (120/2=60). The vector is generated by centre-clipping a Gaussian noise source, which causes approximately 77% of the elements to be zero. Thus, 77% of the updates to the convolutions require no multiplications or additions (only shifting of the convolution element); the total instructions (or additions) are 60×25%= 15.

Chapter 5

Implementation and Performance of DoD 4.8 kps CELP coder

5.1 INTRODUCTION

The DoD 4.8 kbps CELP coder simulation is performed using a C language compiler (Microsoft Quick C) and mainly considers the encoder section of the system. The decoder program has not been written, since the output synthesised speech from the decoder would be identical to the synthesised speech that is already generated by the encoder for the analysis by synthesis procedure.

Section 5.2 briefly presents a formulation that describes all the encoder design, and discusses some reasonable changes to the original coder. Section 5.3 shows the result of speech quality tests.

5.2 ENCODER

Figure 5.1 is the encoder flowchart, Figure 5.2 is the code search block diagram.

It must be noted that the speech signal needs to be converted to digital form. So the speech prediction has to be preceded by the following processing stages:

1) Bandpass filter filtering input speech

2) Analog-to-Digital conversion based on an 8 KHz ± 0.1 percent sampling frequency and with a resolution of at least 12 bits

3) Amplitude scaling, such that the sampled points are in the range -32,768.0 to +32,767.0;

More details of this process can be found in Refs.[4], [5], [6]. The simplest way to do this process is to use a DSP-32C development board in which the analog speech signal can be converted to a digital signal.

In Figure 5.1, speech analysis is performed once per frame (30 ms frame size) in an open loop. The encoder reads in 240 speech samples. This speech block is firstly windowed by a 30 milliseconds Hamming window, and then evaluated by both auto-correlation and the Levinsion-Durbin algorithms to generate the 10^{th} order Linear Predictive filter coefficients.

A 15 Hz bandwidth expansion is necessary *prior* to the inverse filter coefficient quantization [7]. The reason for this is to fit LSP distributions under

the constraints that

(1) no adjacent LSPs can be coded to closer than 15 Hz

(2) the first and last LSPs may not be coded close to 0 and 0.5 rad, respectively.

These constraints will limit the LPC filter's prediction gain. Therefore, undetected LSP transmission errors are unable to cause loud blasts and squeaks commonly associated with spectrum errors, [7].

The Bandwidth expansion parameter Δf=15Hz can be expressed in terms of the parameter in equation $1/A(z/\alpha)$ by α=0.994 = $\exp(-\pi\Delta f T)$, where T is the sampling period. In software the 15-Hz bandwidth expansion can be implemented with the multiplication of the inverse filter coefficients by a vector consisting of the terms

$$g[i] = 0.994^i \quad \text{for } i = 0,1,...,N-1$$

$$(N = 11)$$

The spectrum (or inverse filter coefficients) is coded used 34 bit, independent, nonuniform scalar quantization of Line Spectral Pairs as described in Chapter 2. Because the LSPs are transmitted only once per frame, but are needed for each subframe, they are linearly interpolated to form an intermediate set for each of the four subframes (7.5 ms subframe size) so that the effects of errors are smoothed out [7]. The method of LSP interpolation is described in section 3.7.3 of Ref.[6].

To have the same synthesised speech at both the transmitter and receiver, the filter coefficients (a_K) of the inverse filter A(z) used in code search must be obtained from converting the quantized LSPs back to coefficients a_K. With weighting factor r = 0.8 (corresponding to 568 Hz bandwidth expansion), the weighting filter A(z)/A(z/r) is also easy to implement.

The adaptive code book search and stochastic code book search are both performed in closed-loop model using modified minimized squared prediction error criteria of the perceptually weighted error signal, detailed in Chapter 4. They are shown in Figure 10.

After finding the optimum code vectors (the adaptive codeword and the stochastic codeword), the synthesised speech is then simply obtained by passing the optimum excitation (combined from those two vectors multiplied by their optimum gain factors) to the LP filter 1/A(z).

5.3 PERFORMANCE EVALUATION

The measurement of the final quality and intelligibility of the coded speech signal is restricted in this project because that will require adequate facilities and

trained listeners. In this section, a simple waveform comparison, a SNR objective measure and a paired comparison of informal listening tests (to measure the subjective quality of the reconstructed speech) are carried out.

5.3.1 The Waveform Comparison Of The Original And Coded Speech

Two reconstructed speeches from DoD CELP coder are used for the evaluation. They are

1). Synthesised speech generated from the DoD coder using a 512 codebook;

2). Synthesised speech generated from using a 128 stochastic codebook only.

Figure 5.3a shows a sentence of male voiced speech waveform (input speech). The sentence, " read verse out loud for pleasure", is spoken at normal speed. Figs. 5.3b and 5.3c are reconstructed speech using the 512 and 128 codebooks, respectively. Fig.5.4 gives a similer set of female speech waveforms. The sentence is "the girl at the booth sold 50 pounds".

Fig 5.5 and 5.6 are one word speech samples of both male and female speech. The word 'read' is from a male source and 'sold' is from a female source. Figs. 5.7 to 5.10 give waveforms in both 120 ms and 30 ms segment of speech.

Comparing these waveforms, we conclude that DoD CELP code performs very well both in voiced and unvoiced regions, even in the sub-codebook (128) case. All synthesised speech segments show a highly accurately tracking pitch in the voiced segment and match the random features of the input speech. This might be due to the introduction of the adaptive codebook, which greatly enhances the performance of the coder. In a voiced section of speech, the adaptive codebook typically displays a high level of periodicity. In unvoiced speech the adaptive codebook effectively contains a set of overlapping random sequences, which have the same level of energy as input speech.

Two general conclusions can also be summarized here. Firstly, the 512 codebook synthesised speech appears to carry more prediction information of input speech than in the 128 codebook case. Secondly, the female reconstructed speech seems to be more distorted than male speech does.

5.3.2 Subjective Quality Testing Of The Synthesised Speech

Evidently subjective quality measurement of coded reconstructed speech is needed more than the objective quality testing. However, in this experimentation, proper subjective quality testing is beyond the resources available. Paired comparisons of informal listening tests to measure the subjective quality of the reconstructed speech are used. The synthesised speech sounds very close to the

original speech. As the result, DoD CELP version 3.1 coder produces an almost toll quality of speech. It also speech shows very high intelligibility with no perceptual background noise. The evident to prove that DoD Coder outperforms all the CELP-type coders will be arranged in the further work.

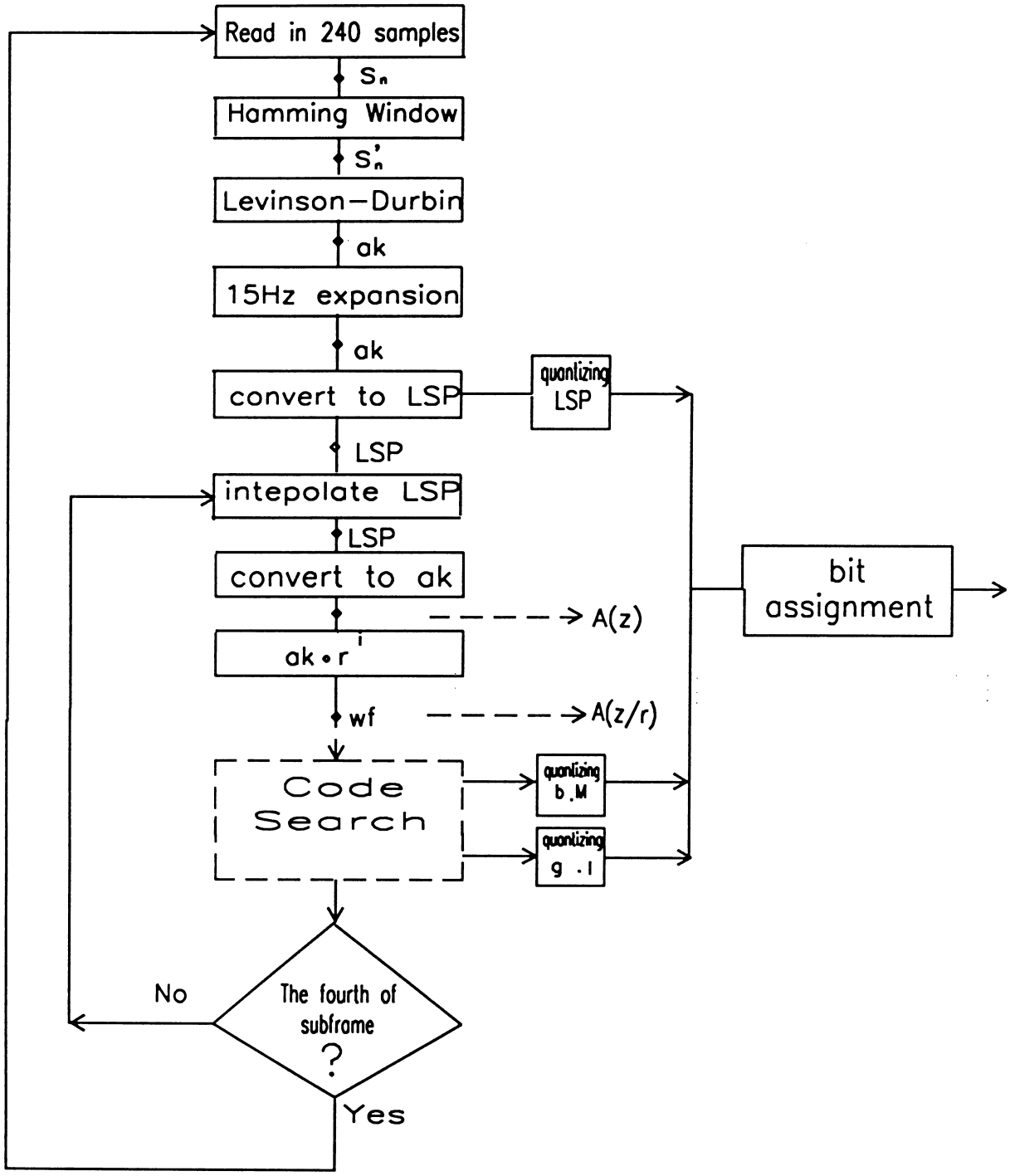


Figure 5.1: Encoder Flowchart

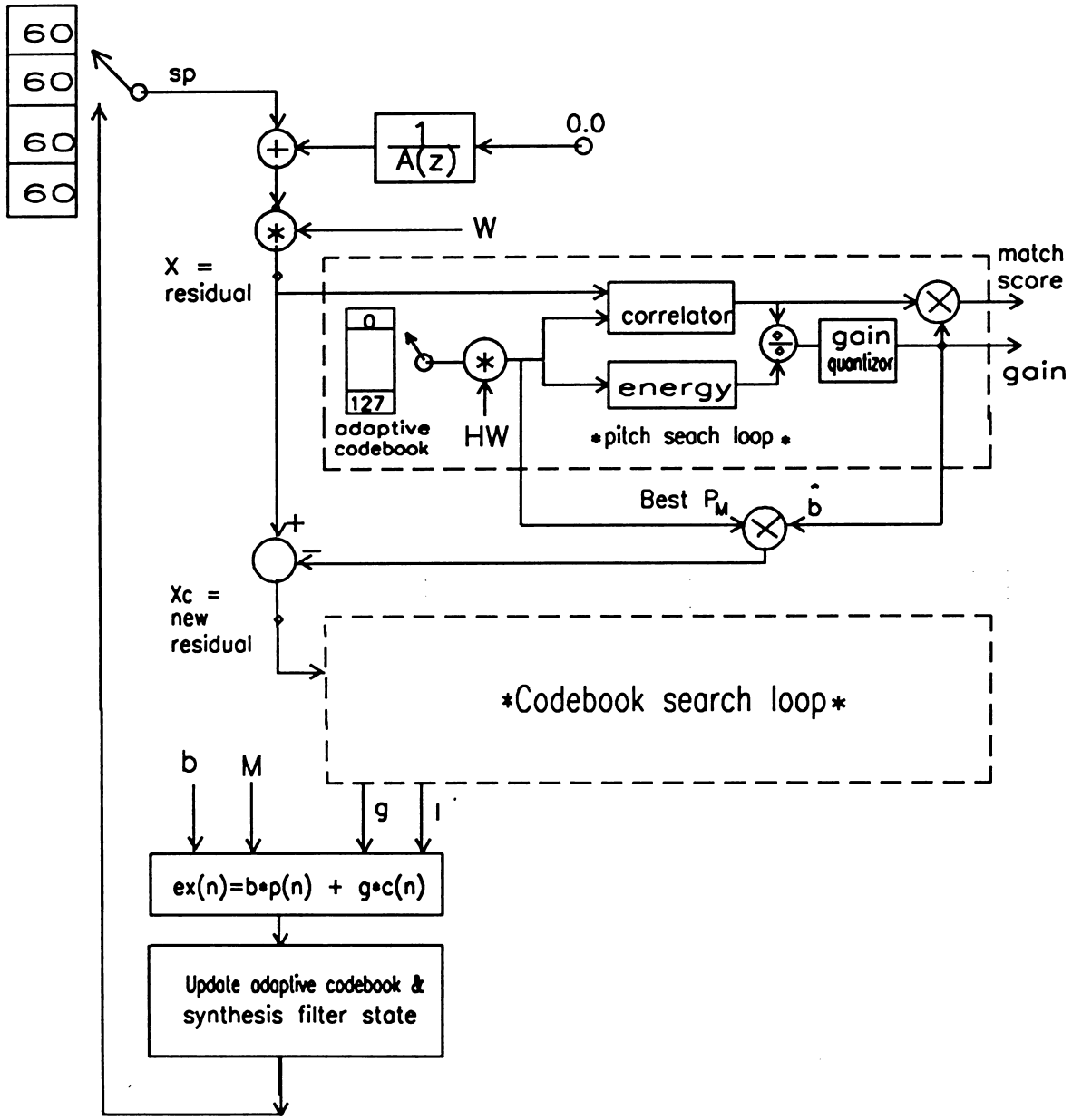
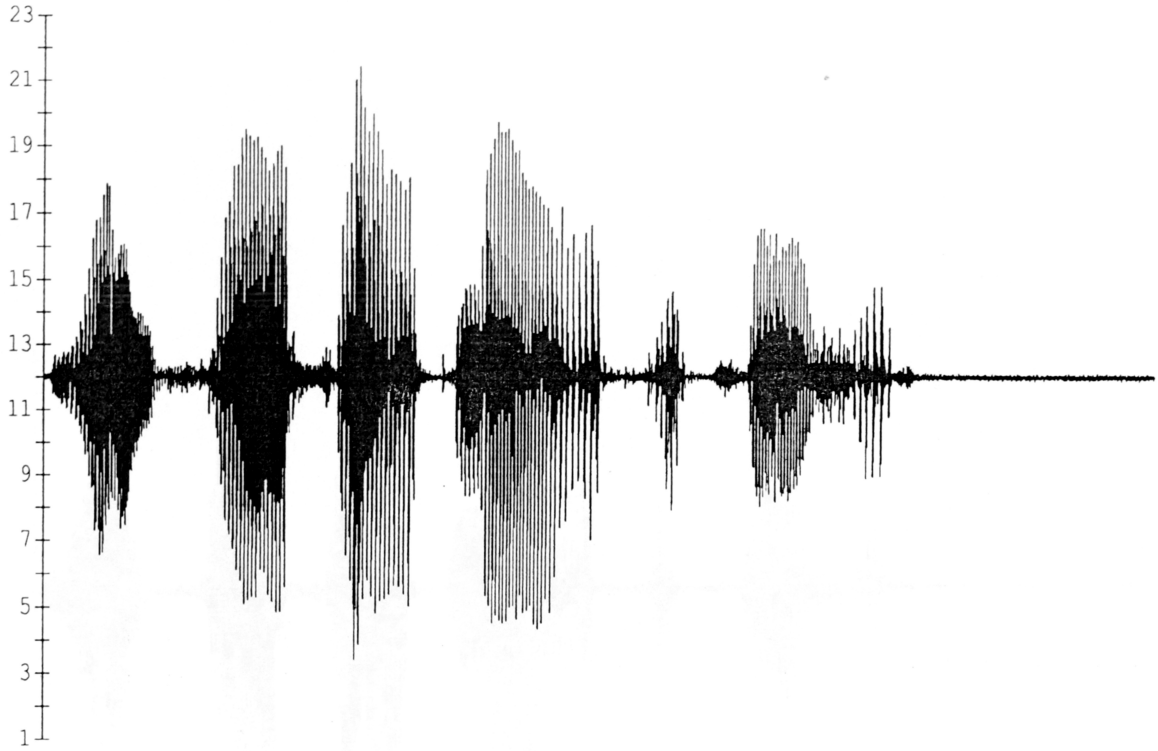
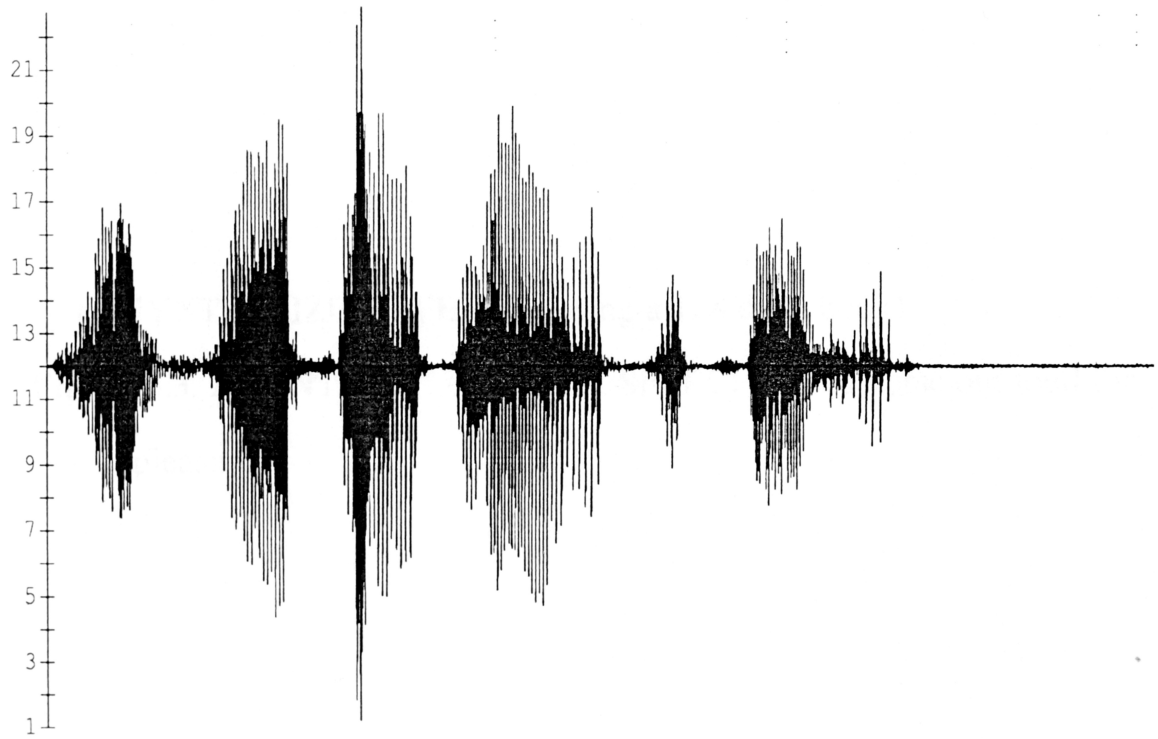


Figure 5.2: Code Search Flowchart



a). ORIGINAL SPEECH (2.5 seconds)

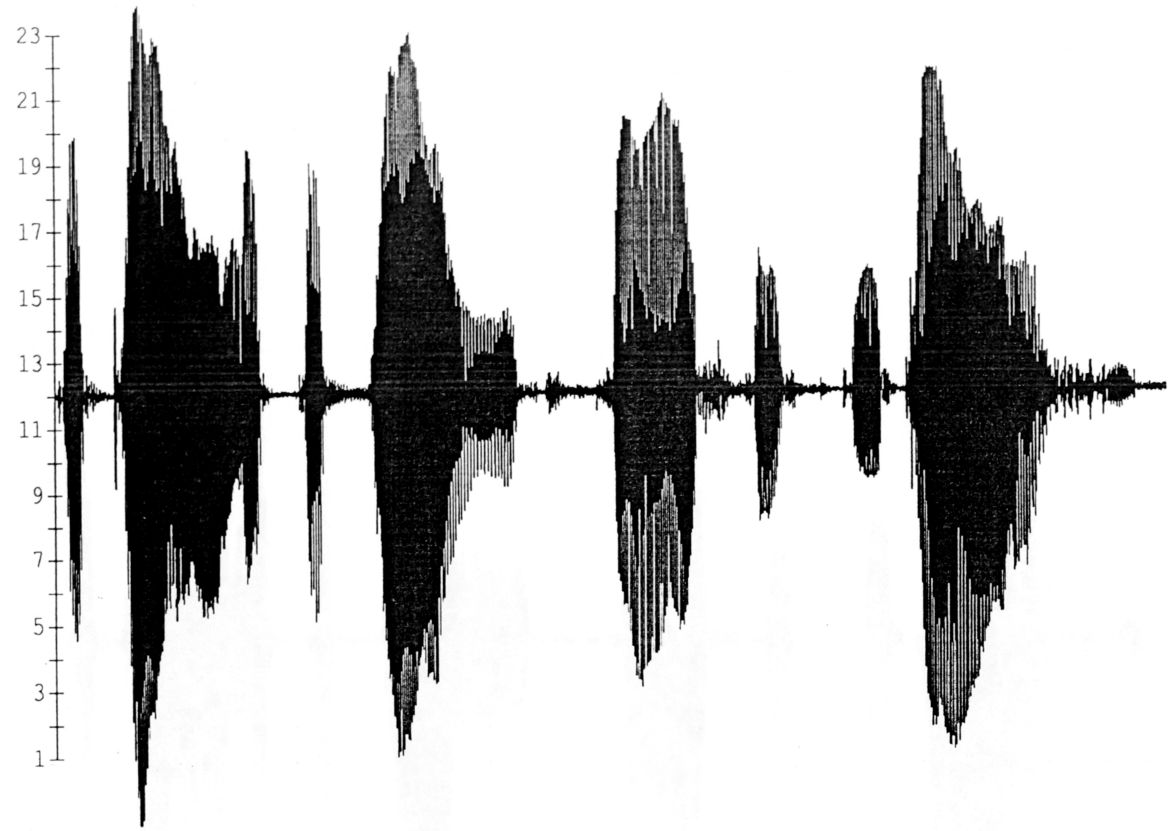


b). SYNTHESIZED SPEECH (using a 512 codebook)

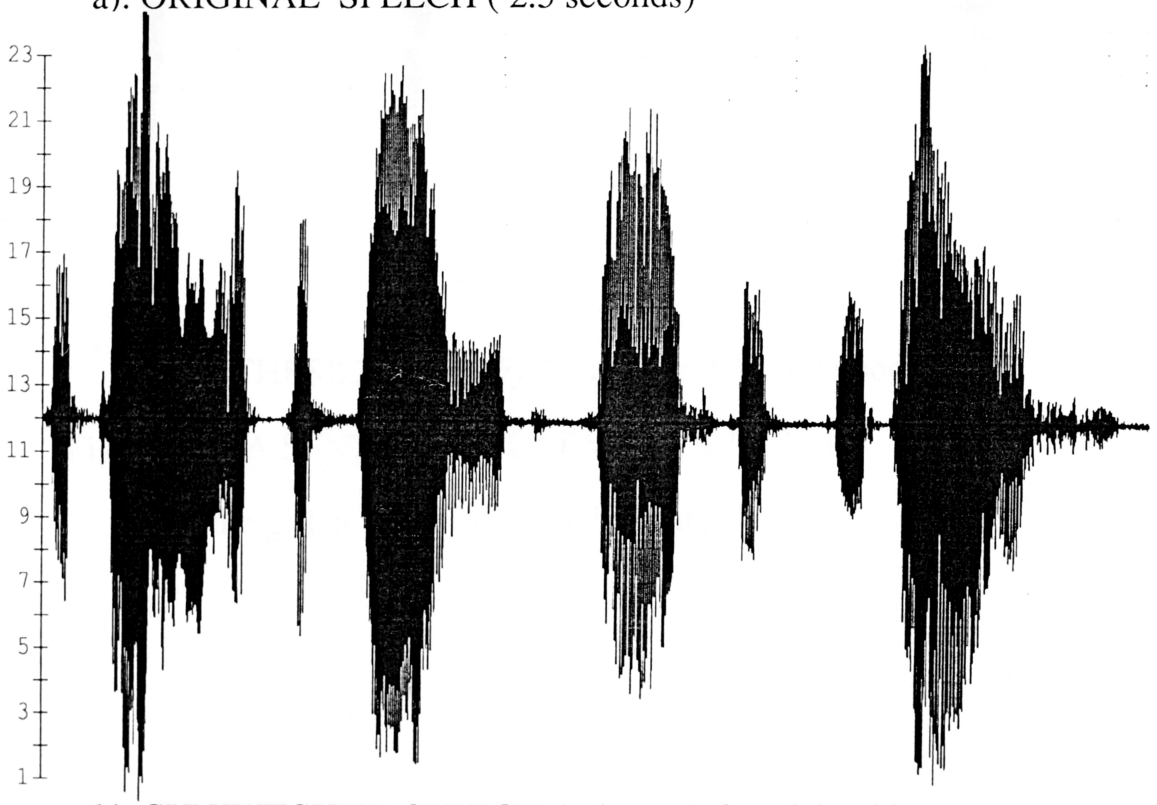
Figure 5.3. MALE SPEECH :"read verse out loud for pleasure"

c). SYNTHESIZED SPEECH (using a 128 code book)

Figure 5.3. A SENTENCE OF MALE SPEECH :"read verse out loud for pleasure"

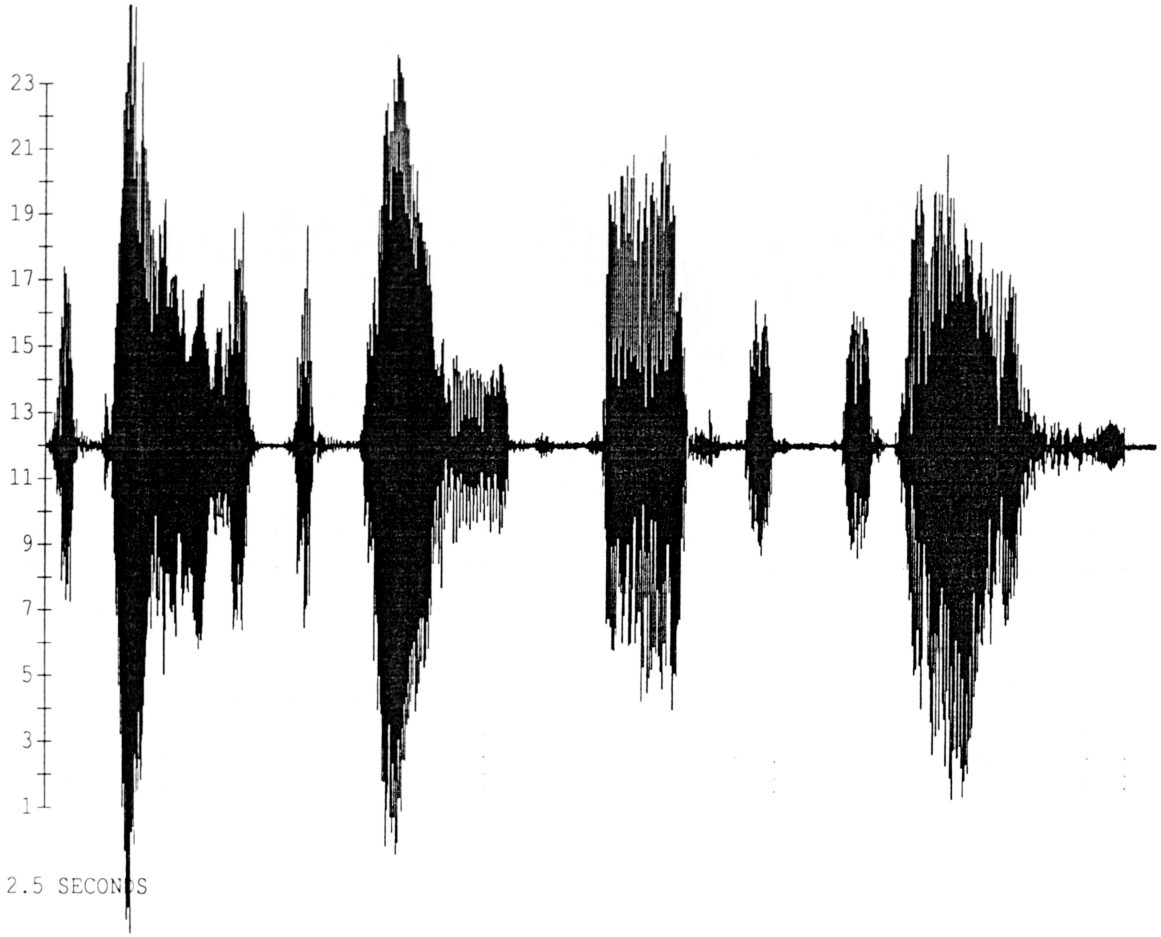


a). ORIGINAL SPEECH (2.5 seconds)



b). SYNTHESIZED SPEECH (using a 512 codebook)

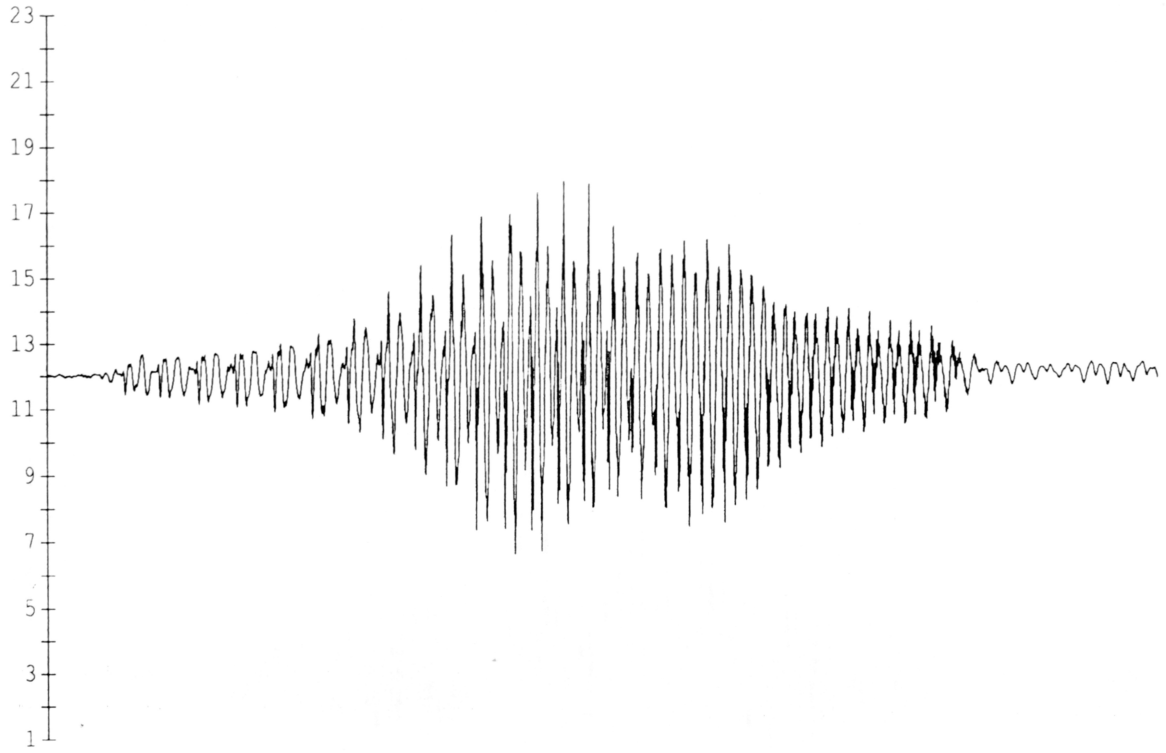
Figure 5.4. FEMALE SPEECH : "the girl at the booth sold 50 pounds"



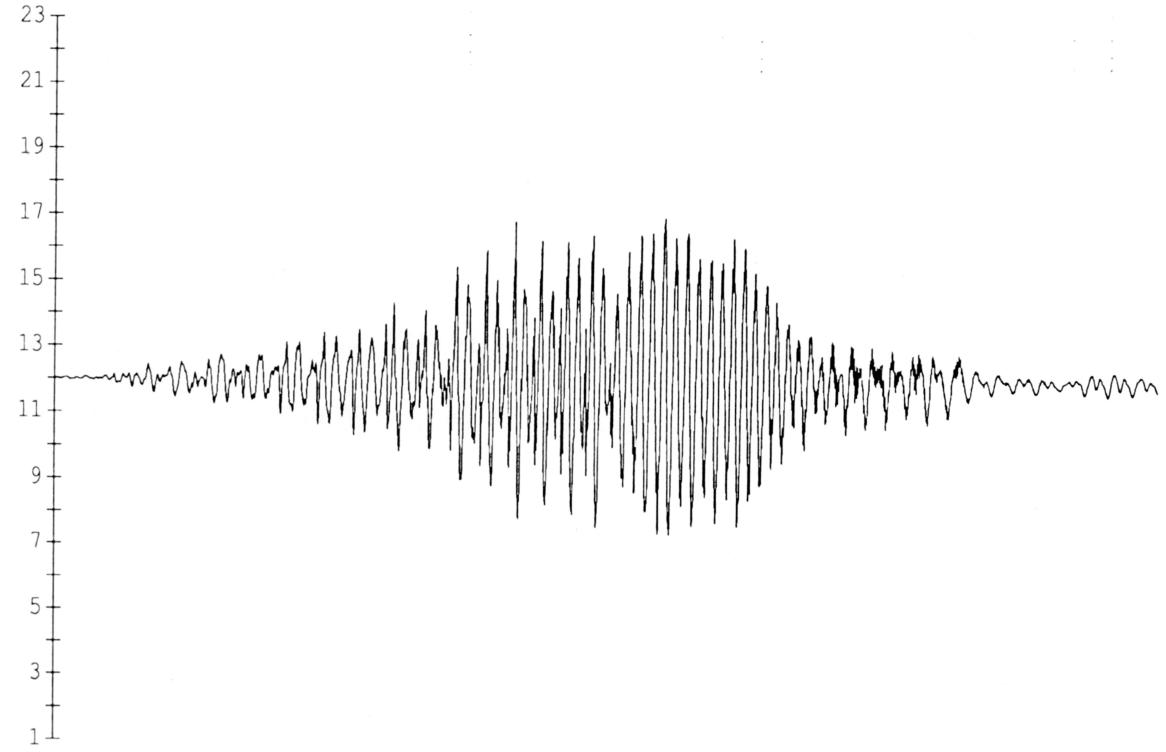
c). SYNTHESIZED SPEECH (using a 128 code book)

Figure 5.4. A SENTENCE OF FEMALE SPEECH :

"the girl at the booth sold 50 pounds"

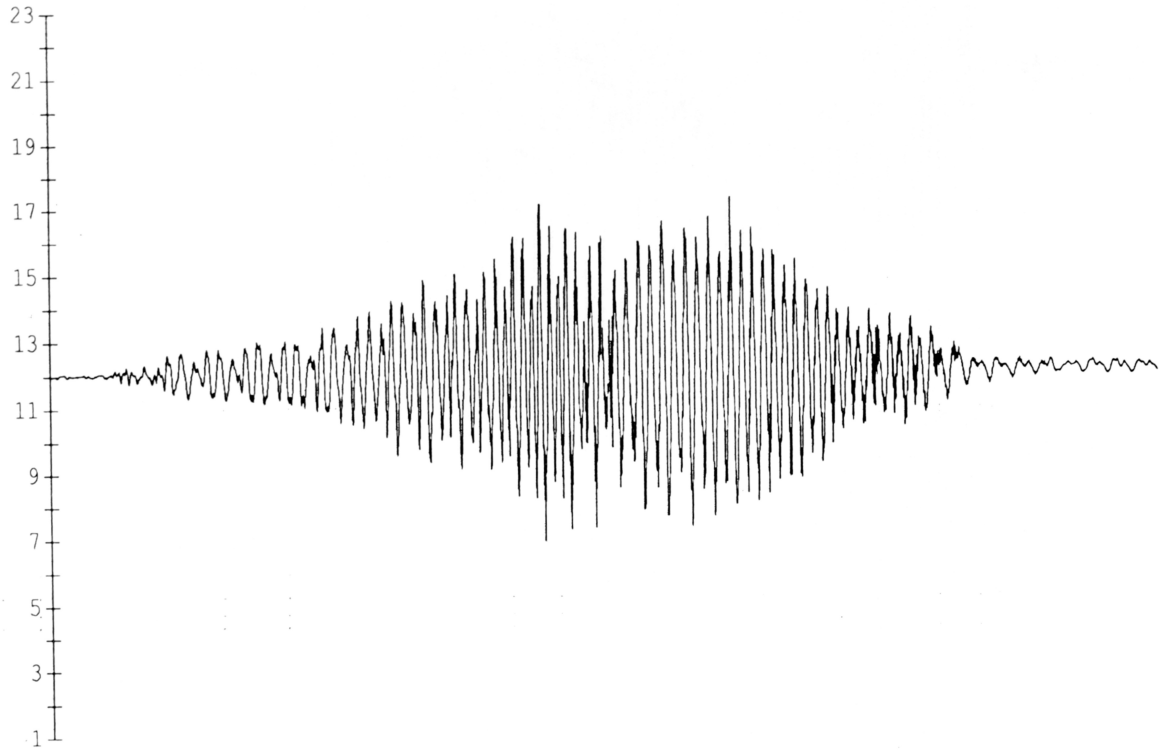


a). ORIGINAL SPEECH (300 ms)



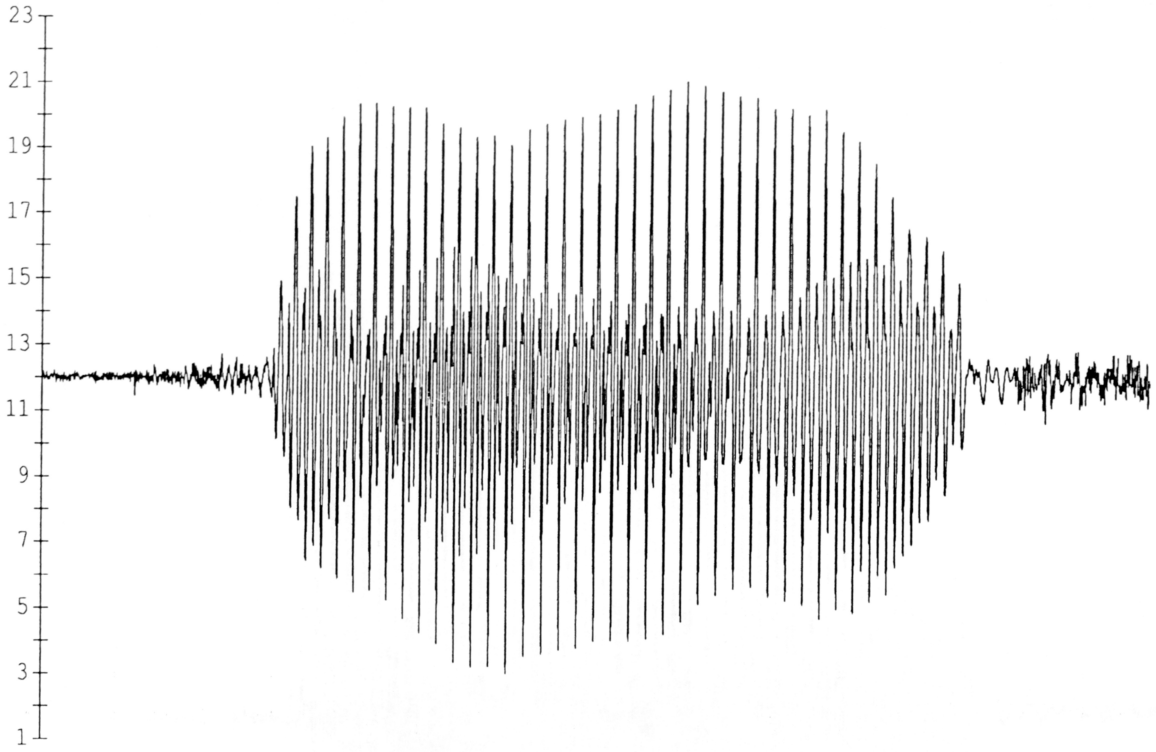
b). SYNTHESIZED SPEECH (using a 512 codebook)

Figure 5.5. ONE WORD MALE SPEECH : "read"

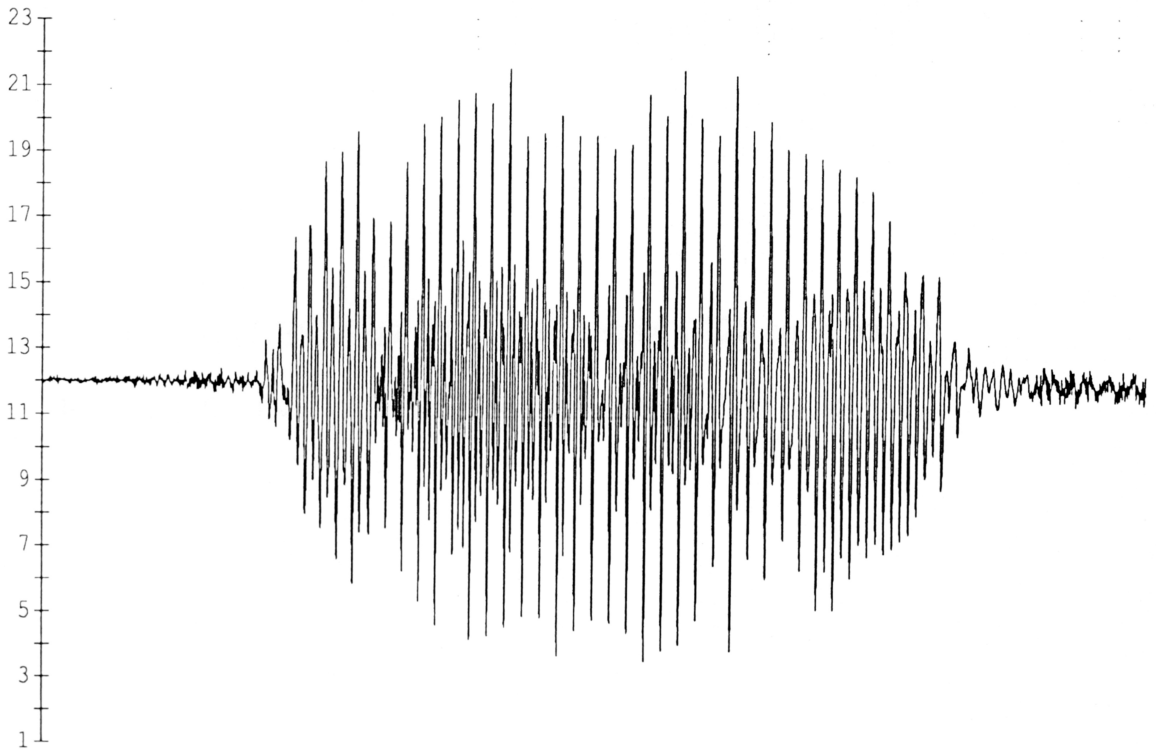


c). SYNTHESIZED SPEECH (using a 128 code book)

Figure 5.5. ONE WORD MALE SPEECH : "read"

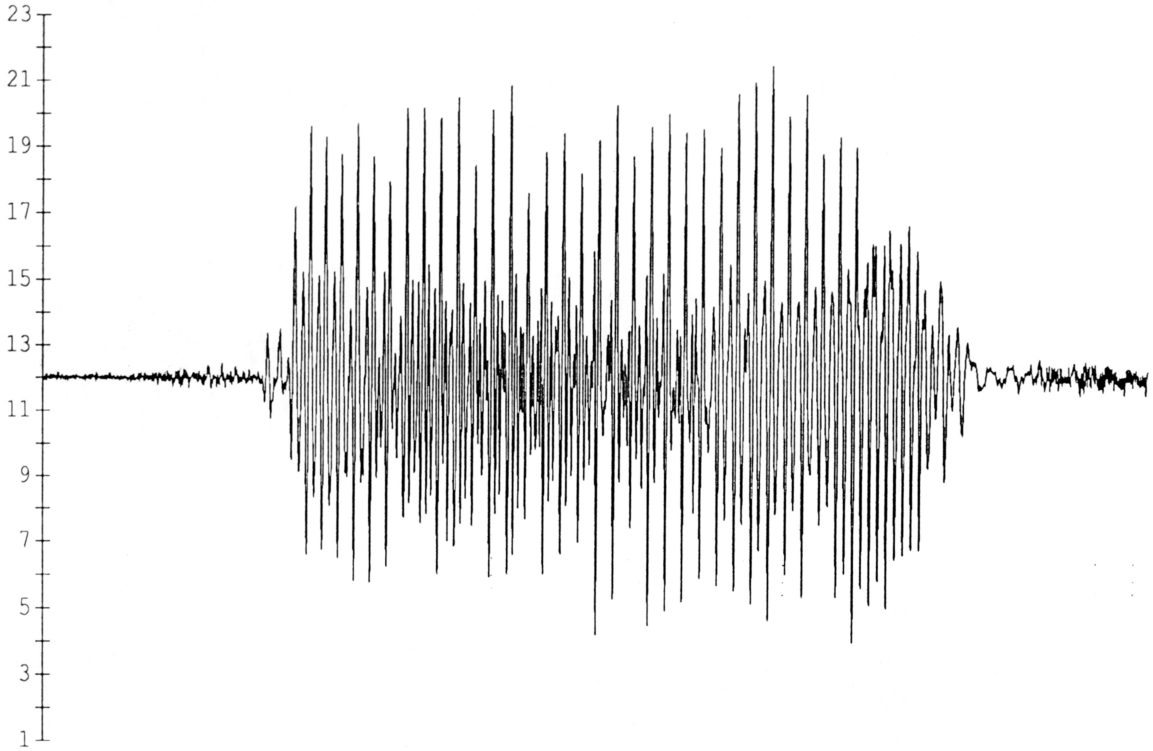


a). ORIGINAL SPEECH (300 ms)



b). SYNTHESIZED SPEECH (using a 512 codebook)

Figure 5.6. ONE WORD FEMALE SPEECH : "sold"



c). SYNTHESIZED SPEECH (using a 128 code book)

Figure 5.6. ONE WORD FEMALE SPEECH : "sold"

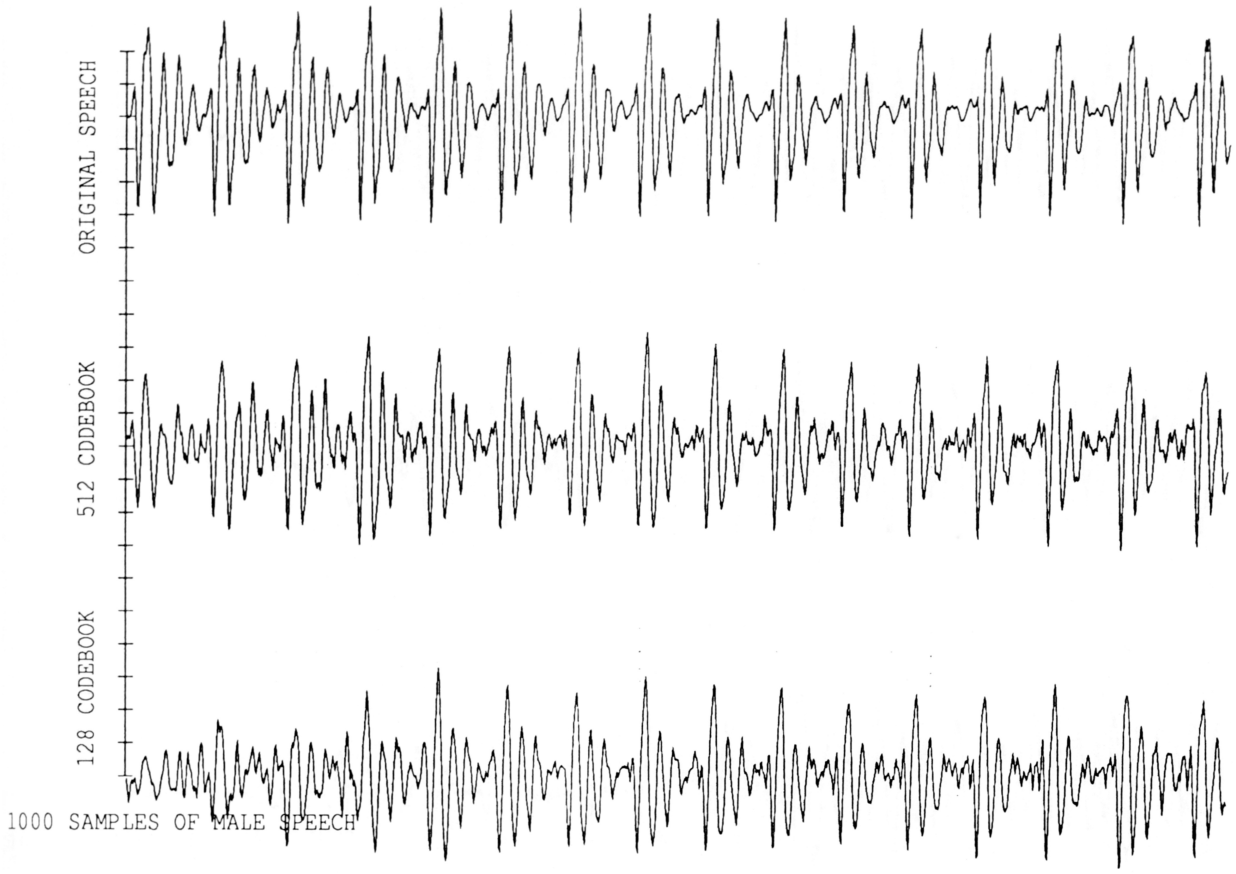


Figure 5.7. 100 ms MALE SPEECH (voiced region)

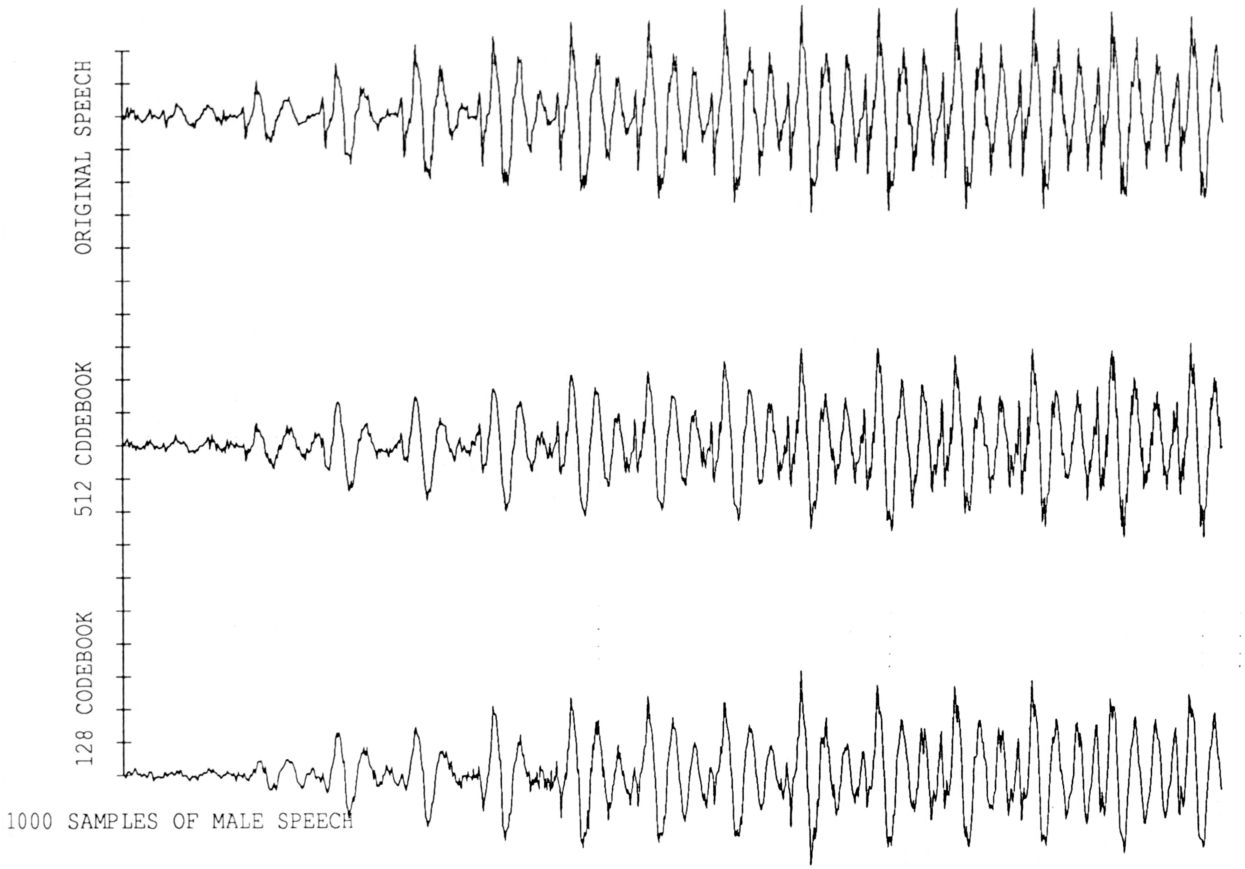


Figure 5.7. 100 ms MALE SPEECH (unvoiced and voiced region)

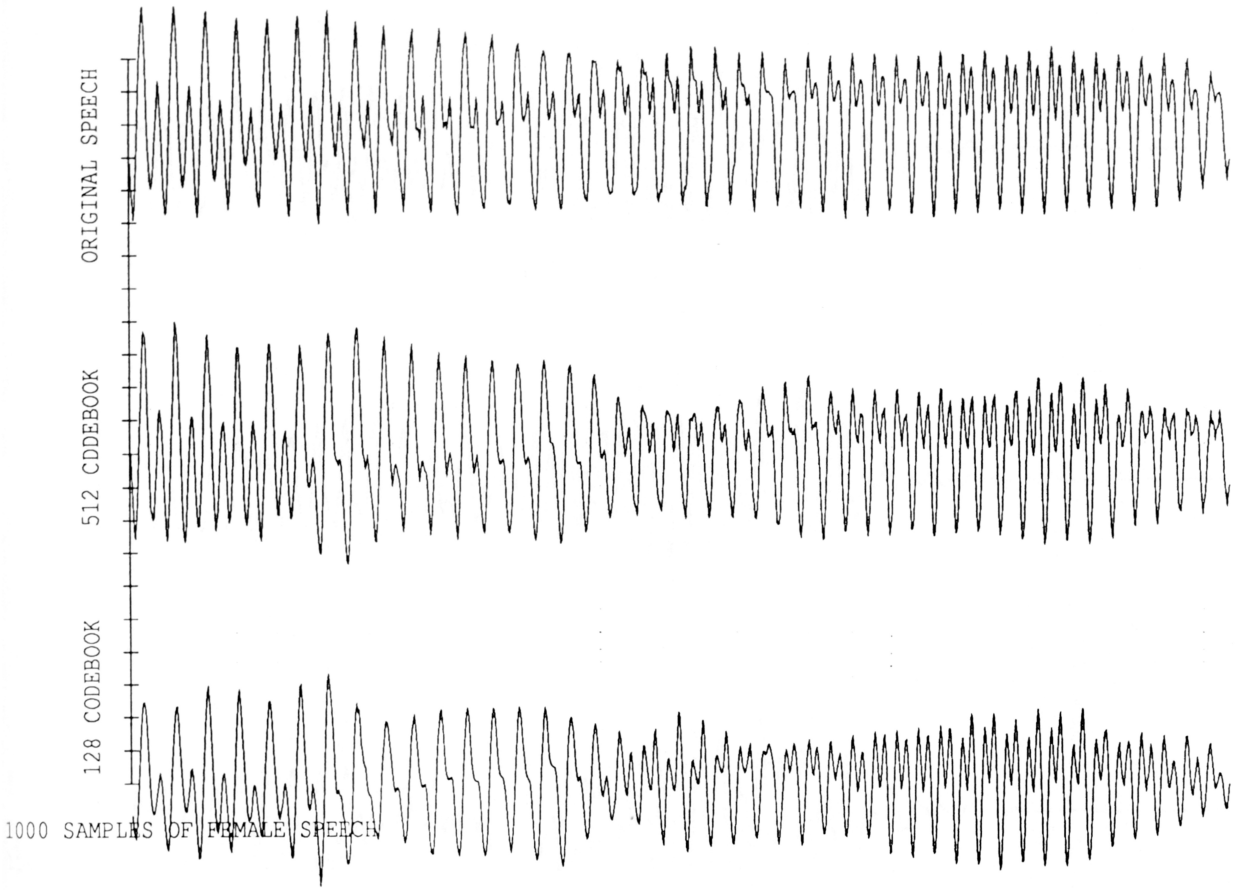


Figure 5.8. 100 ms FEMALE SPEECH (voiced region)

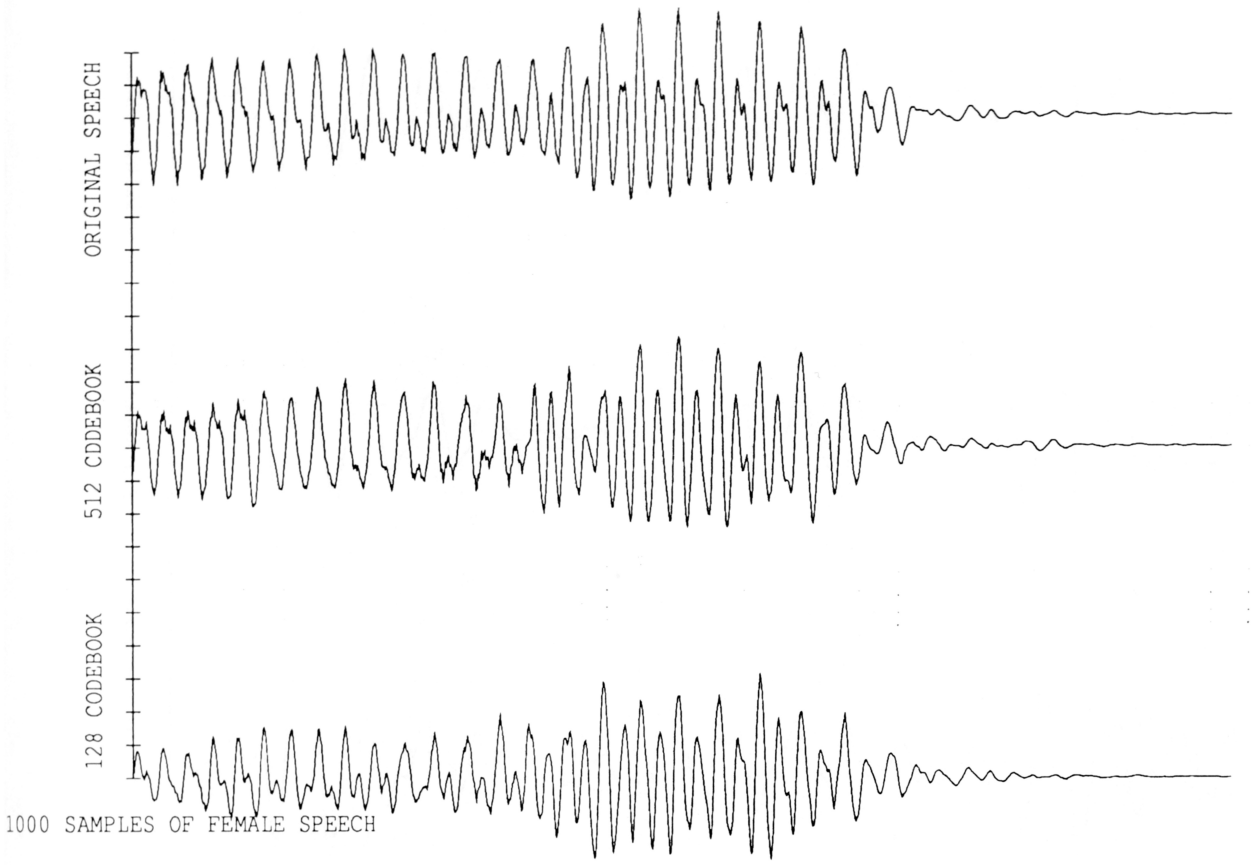


Figure 5.8. 100 ms FEMALE SPEECH (voiced and unvoiced region)

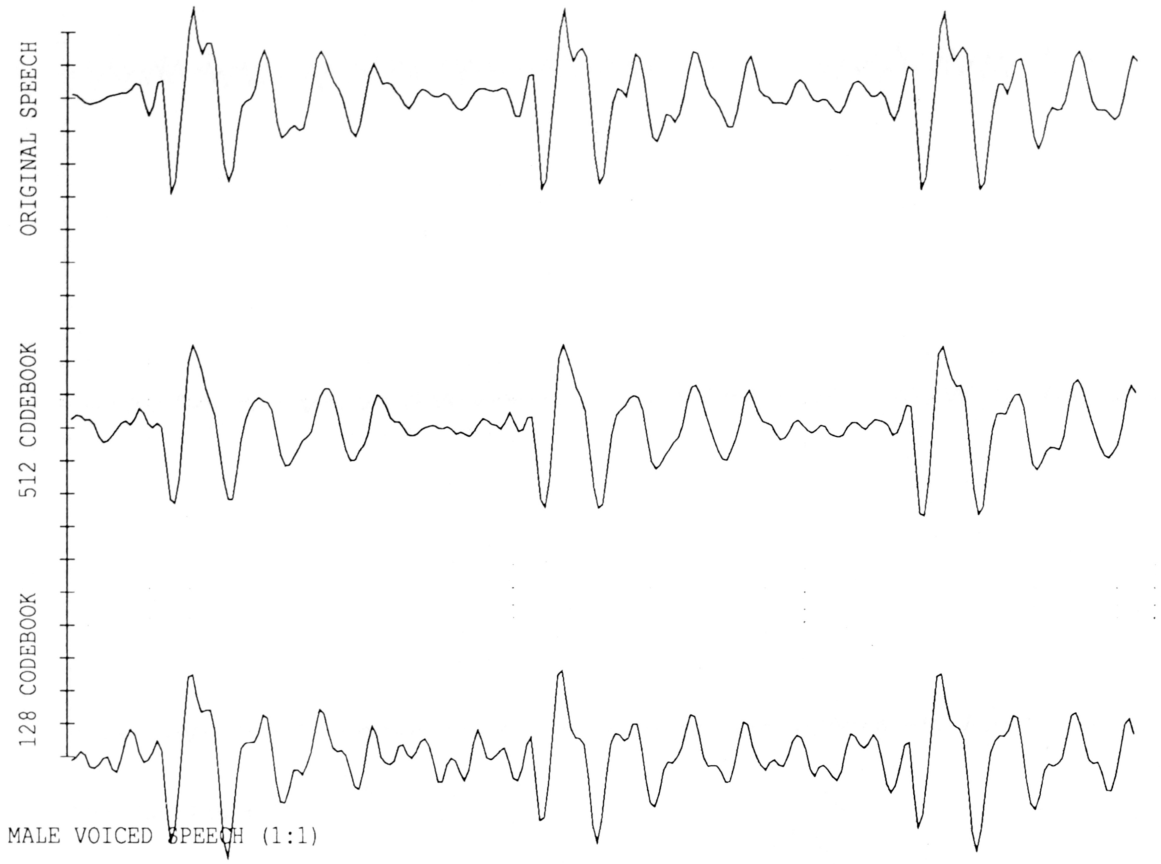


Figure 5.9. 30 ms MALE VOICED SPEECH

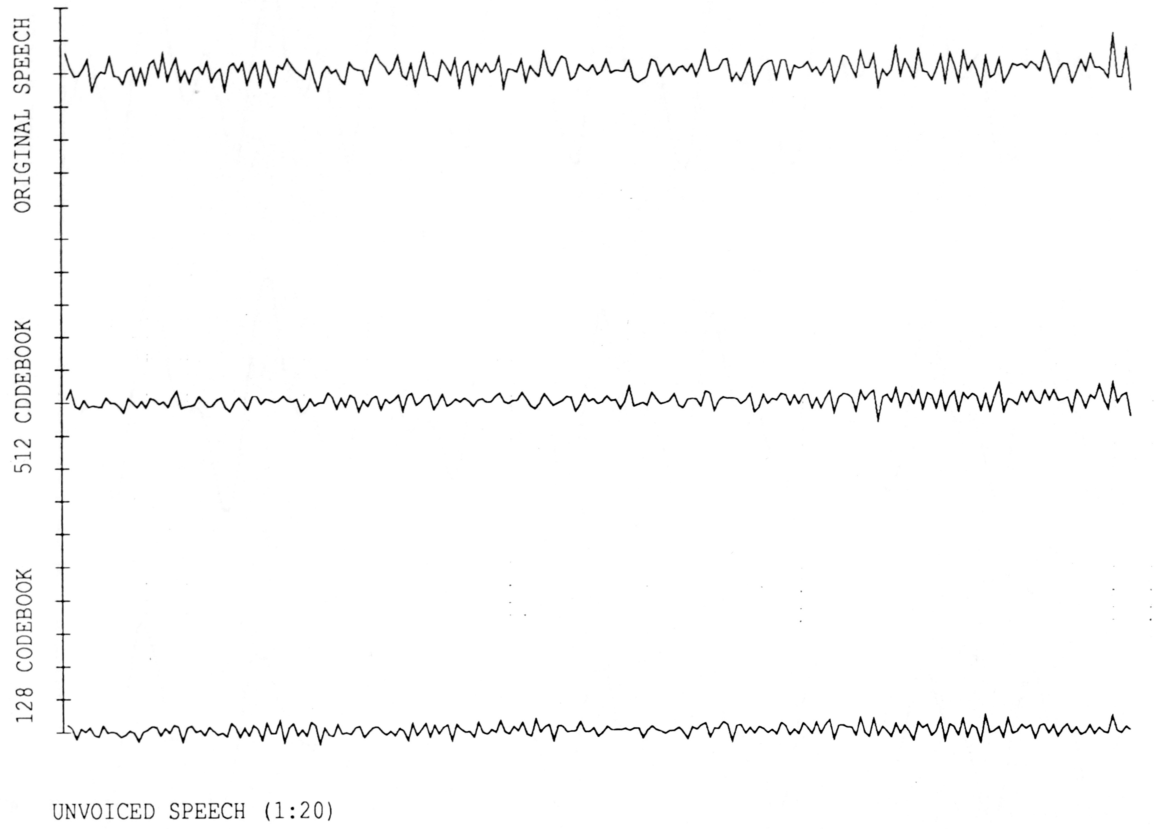


Figure 5.9. 30 ms UNVOICED SPEECH

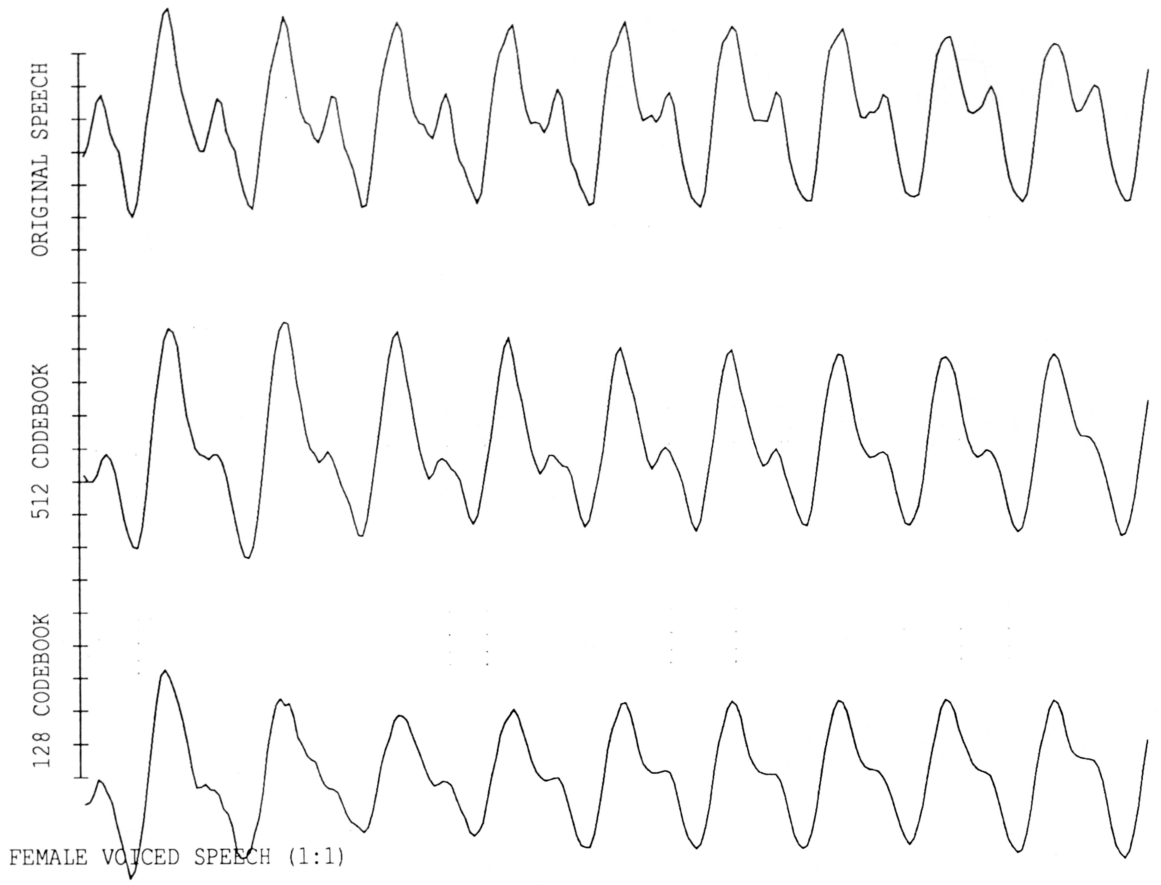


Figure 5.10. 30 ms FEMALE VOICED SPEECH

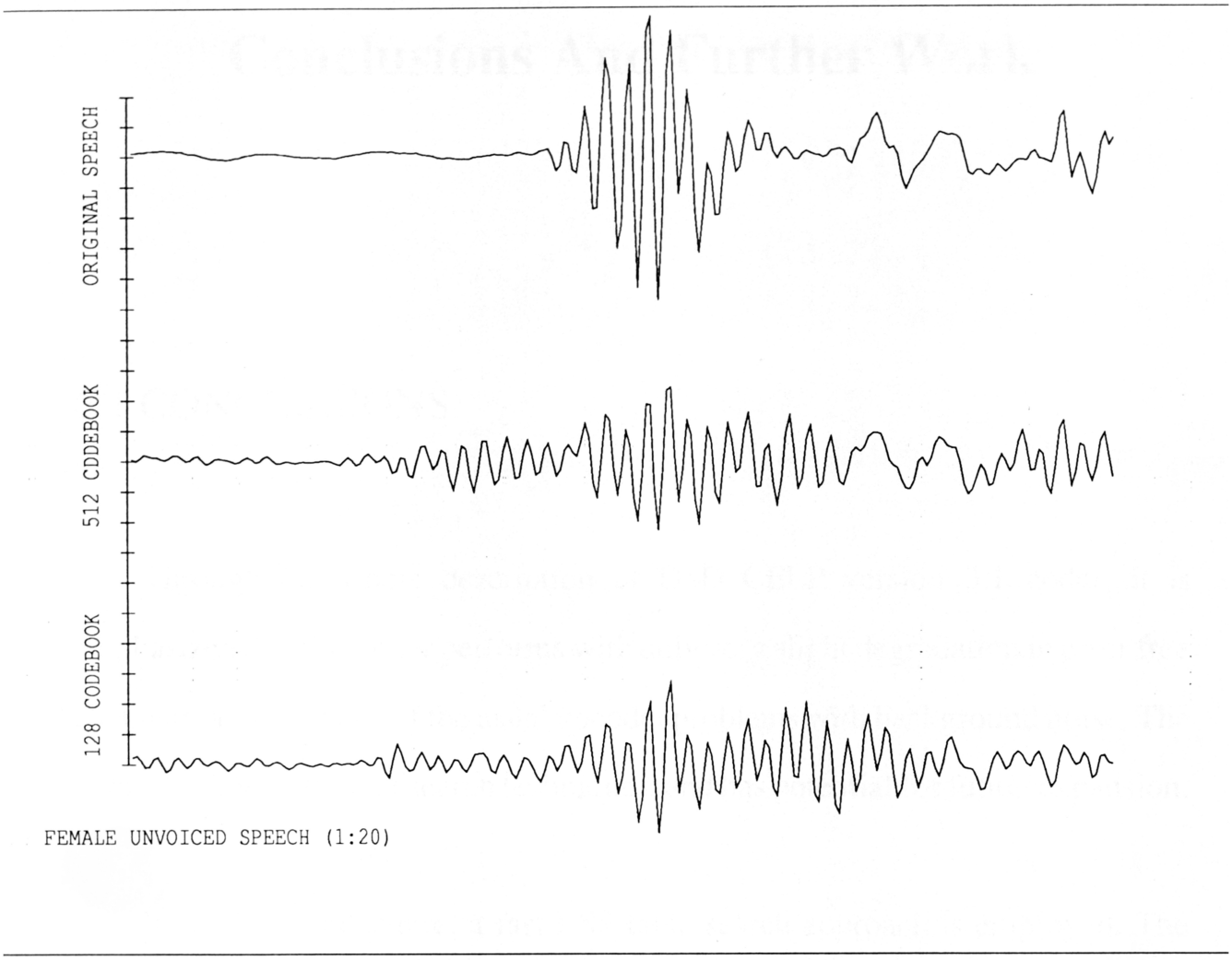


Figure 5.10. 30 ms UNVOICED SPEECH

Chapter 6

Conclusions And Further Work

6.1 CONCLUSIONS

Through the whole description of DoD CELP version 3.1 coder, it is emphasized that this coder performs with only very slight degradation in error free channels, and has none of the usual vocoder problems with background noise. The coder includes efficient search techniques, and has potential for future expansion.

In the LPC procedure, a fast LSP table search approach is employed. The LPC parameters are coded as monotonically increasing LSPs to guarantee a stable LPC filter. In addition, the effects of LSP errors are smoothed out by interpolating LSPs each subframe.

The adaptive codebook brings the greatest improvement in speech quality by providing high resolution for female speakers and lower resolution for male and child speakers. The techniques of using odd/even subframes, of determining

submultiples of delays, and of the end-point correction technique have greatly reduced the computational complexity and data rate while causing no perceivable loss in speech quality. The performance of the adaptive code book is superior to the conventional filtering approach.

The special form of stochastic code book containing sparse (77% zero values), overlapped shift by 2, and ternary valued samples (-1,0,+1) has reduced memory requirements. It is compact and allows a fast search procedure, causes no degradation in speech quality relative to other types of code book, and significantly reduces the search computation.

Moreover, the coder is practical for implementation in real-time using the AT&T DSP 32C chip.

6.2 FURTHER PLAN

The results presented in chapter 5 need to be confirmed by a formal measurement, e.g. using Dynastat's Diagnostic Rhyme Test (DRT) and Diagnostic Acceptability Measure (DAM) criteria.

To be a successful and efficient coding system, this coder needs the addition of two further techniques, namely the non-integer delay technique for improving

voiced speech prediction , and a forward error correcting (15, 11) Hamming code for channel error protection. Further confirmation will also include testing the performance with low percentage of bit error in order to show this coder is robuse to chnnel error and noisy enviroments.

The coder can operate at different levels of computational complexity to provide interoperability between simple and powerful implementations (e.g. a DSP 32C chip with 12.5 MIPS rating can be used for real-time implementation of a 128 codebook CELP, while for a 256 or 512 codebook a 16.6 or 25 MIPS DSP chip needs to be used [4]). The first stage of the real-time implementation can start with a 128 codebook. The Binary Engineering Standalone Hardware Platform with 12.5 MIPS rating DSP chip, (developed by Dipanjan Sen at the University of New South Wales Speech Group in 1991), could be used for a real time experiment.

Appendix 1 : STOCHASTIC TERNARY CODE BOOK

0.	1.	0.	0.	0.	1.	0.	0.	0.	0.	0.	0.	1.	0.	1.	0.	0.	0.	0.	-1.
0.	0.	-1.	0.	-1.	0.	1.	0.	0.	0.	0.	0.	0.	0.	-1.	0.	-1.	-1.	0.	0.
-1..	0.	-1.	0.	-1.	0.	-1.	0.	0.	0.	0.	0.	0.	0.	-1.	0.	0.	0.	1.	0.
0.	0.	0.	-1.	0.	0.	0.	0.	1.	0.	0.	0.	0.	0.	0.	0.	0.	1.	-1.	0.
1.	0.	1.	0.	-1.	1.	0.	0.	0.	0.	0.	-1.	0.	0.	1.	0.	0.	0.	-1.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	-1.	-1.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	-1.	-1.
0.	1.	-1.	0.	0.	-1.	-1.	0.	0.	0.	0.	1.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	-1.	0.	-1.	-1.	0.	0.	-1.	0.	1.	0.	0.	0.	1.	0.	0.	0.	0.
0.	-1.	0.	0.	0.	-1.	0.	0.	0.	0.	0.	0.	-1.	0.	1.	0.	0.	0.	0.	0.
0.	0.	-1.	1.	0.	0.	0.	-1.	1.	0.	0.	1.	0.	0.	-1.	0.	0.	0.	0.	0.
0.	-1.	0.	0.	1.	-1.	0.	0.	1.	0.	0.	0.	0.	0.	0.	1.	1.	1.	0.	0.
1.	1.	0.	0.	0.	-1.	0.	0.	0.	-1.	0.	0.	-1.	1.	0.	0.	0.	0.	0.	-1.
-1.	0.	0.	0.	0.	-1.	0.	0.	0.	0.	0.	-1.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	1.	1.	0.	-1.	-1.	0.	0.	1.	0.
0.	0.	0.	0.	0.	-1.	0.	0.	-1.	1.	0.	0.	0.	0.	0.	0.	-1.	0.	0.	-1.
0.	0.	0.	0.	0.	-1.	0.	0.	-1.	0.	0.	0.	0.	0.	-1.	-1.	-1.	0.	-1.	-1.
0.	1.	0.	0.	0.	0.	1.	0.	0.	0.	1.	0.	0.	0.	0.	1.	-1.	0.	0.	0.
0.	0.	0.	1.	-1.	0.	0.	-1.	0.	-1.	0.	0.	-1.	0.	0.	1.	0.	0.	0.	0.
-1.	-1.	0.	-1.	0.	0.	0.	0.	0.	0.	0.	0.	0.	1.	0.	0.	-1.	0.	0.	0.
-1.	0.	1.	0.	0.	-1.	0.	0.	0.	-1.	0.	0.	0.	-1.	0.	0.	0.	0.	-1.	0.
0.	-1.	-1.	0.	0.	0.	1.	0.	-1.	0.	0.	0.	0.	0.	1.	0.	0.	0.	0.	0.
0.	0.	1.	1.	0.	0.	0.	1.	1.	0.	0.	0.	-1.	0.	0.	0.	0.	0.	-1.	1.
1.	0.	0.	0.	0.	0.	0.	0.	0.	1.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	-1.	0.	1.	1.	0.	0.	0.	0.	1.	0.	0.	1.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	1.	0.	0.	1.	0.	-1.
-1.	1.	0.	0.	0.	1.	0.	1.	0.	1.	0.	0.	-1.	0.	0.	0.	0.	0.	0.	0.
0.	-1.	0.	-1.	0.	0.	0.	-1.	0.	0.	-1.	0.	1.	1.	0.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	-1.	-1.	0.	0.	0.	0.	1.	0.	0.	0.	0.	0.	0.
-1.	0.	-1.	-1.	0.	0.	-1.	0.	-1.	0.	0.	0.	0.	0.	1.	0.	1.	0.	0.	0.
0.	0.	0.	1.	-1.	0.	1.	0.	0.	0.	0.	-1.	0.	0.	0.	0.	0.	0.	1.	0.
0.	-1.	-1.	0.	0.	0.	0.	0.	0.	1.	0.	0.	0.	0.	1.	-1.	0.	1.	0.	1.
0.	1.	0.	0.	0.	1.	0.	0.	0.	0.	-1.	0.	0.	0.	-1.	0.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	-1.	0.	0.
0.	0.	0.	0.	0.	0.	1.	0.	0.	-1.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1.	0.	0.	1.	0.	0.	1.	0.	0.	0.	0.	0.	0.	-1.	1.	0.	1.	0.	0.	-1.
0.	0.	0.	0.	0.	0.	0.	0.	0.	1.	-1.	0.	0.	0.	-1.	0.	0.	0.	0.	-1.
0.	0.	0.	0.	1.	-1.	0.	-1.	1.	0.	0.	0.	0.	-1.	0.	0.	0.	1.	0.	0.
0.	0.	0.	-1.	0.	0.	0.	0.	0.	0.	-1.	0.	-1.	1.	1.	-1.	0.	0.	0.	1.
0.	1.	0.	0.	0.	-1.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	1.	0.	1.	0.
0.	0.	0.	0.	-1.	0.	0.	0.	1.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	-1.	0.	0.	0.	1.	0.	0.	0.	0.	-1.	0.	1.	-1.	0.	1.	0.	0.	0.
0.	0.	-1.	0.	0.	0.	1.	0.	-1.	1.	-1.	0.	0.	1.	0.	0.	0.	0.	-1.	0.
0.	0.	-1.	0.	0.	0.	0.	-1.	0.	0.	0.	0.	1.	0.	0.	0.	-1.	0.	0.	0.
0.	0.	0.	1.	0.	0.	0.	0.	0.	0.	1.	-1.	0.	0.	1.	0.	0.	0.	0.	-1.
0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	-1.	0.	0.	0.	1.	0.	0.	1.	0.	0.
0.	0.	1.	0.	0.	1.	0.	0.	-1.	0.	0.	0.	1.	0.	0.	0.	0.	0.	0.	0.
1.	0.	0.	-1.	0.	0.	-1.	0.	0.	0.	1.	0.	0.	0.	0.	0.	0.	0.	0.	0.
0.	0.	0.	1.	0.	0.	0.	0.	0.	1.	0.	0.	-1.	1.	-1.	0.	1.	0.	0.	0.
0.	0.	0.	0.	1.	-1.	0.	0.	0.	0.	0.	0.	0.	0.	-1.	1.	0.	0.	0.	0.
0.	0.	0.	0.	0.	0.	0.	1.	0.	0.	0.	0.	0.	-1.	0.	0.	0.	0.	-1.	0.
0.	0.	-1.	0.	-1.	0.	0.	0.	0.	0.	0.	-1.	0.	0.	-1.	0.	0.	0.	0.	0.
0.	0.	0.	1.	-1.	0.	0.	0.	0.	-1.	0.	1.	0.	0.	1.	0.	0.	0.	0.	-1.
0.	0.	0.	0.	1.	0.	0.	0.	0.	-1.	0.	0.	-1.	0.	0.	0.	0.	0.	0.	1.
0.	0.																		

REFERENCES

[1] M.R.Schroeder and B.S.Atal,"Code-Excited Linear Prediction (CELP): High-Quality Speech at Low Bit Rates," Proc. IEEE-ICASSP, pp.937-940, April 1985.

[2] G.Davidson and A.Gersho. "Complexity reduction methods for vector excitation coding". Proc. IEEE-ICASSP, pp.3055-3058. 1986.

[3] Kleijn.W, D.Krasinski and R.Ketchum,"Improved Speech Quality and Efficient Vector Quantization in SELP", Proceedings of ICASSP,pp.155-158,1988.

[4] J.P.Campbell, E.Tremain. " CELP Documentation Version 3.1 " U.S Government Department of Defense,R5 Fort Medde,MD 20755-6000.15 December 1989.

[5] Tremain.T, J.Campbell and V.Welch."The Proposed Federal Standard 1016 4800 bps Voice Coder: CELP". Volume 5 number 2. SPEECH Technology. April/May 1990.

[6] Fenichel, R., "Proposed Federal Standard 1016.Telecommunications: Analog to digital conversion of radio voice by 4,800 bit/second Code Excited Linear Prediction (CELP)." (First Draft and Second Draft), National 20305-2010, 13 November 1989.

[7] J.Campbell, Tremain.T and V.Welch." An expandable error-protected 4800 BPS CELP coder (U.S Federal Standard 4800 PS voice coder). Proc.IEEE-ICASSP, pp.735-738,1989.

[8] Kemp.D, R.Sueda and T. Tremain,"An Evalution of 4800 bps Voice Coders". Pro. IEEE-ICASSP, pp200-203, 1989.

[9] Lin, D., "Speech Coding Using Efficient Pseudo-Stochastic Block Codes", Pro. IEEE-ICASSP, pp1354-1357. 1987.

[10] Kroon. P, and B.Atal, "Strategies for Improving the Performance of CELP Coders at Low Bit Rates. Proceedings of ICASSP, PP.151-154, 1988.

[11] Gerson.I.A, and Jasiuk.M.A. "Vector Sum Excited Linear Prediction (VSELP) Speech Coding at 8 KBPS " Proc. IEEE-ICASSP. pp.461-464, 1990.

[12] W.B.Kleijn, D.J.Krasinski and R.H.Ketchum " An Efficient Stochastically Excited Linear Predictive Coding Algorithm for High Quality Low Bit Rate Transmission of Speech" Speech Communication. Vol.VII. pp.305-316, 1988.

[13] Thomas E.Tremain, Joseph P. "A 4.8 KBPS Code Excited Linear Predictive Coder", Processing of the Mobile Satellite Conference, pp. 491-496. 1988.

[14] Xydeas, C., M.Iteton and D.Baghbadrani, "Theory and Real Time Implementation of a CELP Code at 4.8 and 6.0 kbits/ second Using Ternary Code Excitation", Proceedings of the Fifth International Conference on Digital Processing of Signals in Communications, pp.167-174, 1988.

[15] R.C.Rose and T.P.Barnwell III,"Qualitycomparision of Low Complexity 4800 bps Self excited and Code excited Vocoders," Proc. ICASSP, pp1637-1640, 1987.

[16] Kroon.P, and B.Atal, "On Improving the Performance of Pitch Predictors in Speech coding Systems". Abstracts of the IEEE Workshop on Speech Coding of Telecommunications, pp.49-59, 1989.

[17] Kroon, P. And B.S Atal,"Quatization Procedures for the Excitation in CELP Coders," Proc. of Int. Conf. Acoust, Speech, and Signal Process, pp.2185-2188. 1987.

[18] J.D.Markel and A.H.Gray,Jr. "Linear Prediction of Speech" New York: Springer-Verlag, pp.130, 1976.

[19] Papamichalis, chapter 5 "Linear Prediction Coding: A Parametric Description of Speech" in "Practical Approaches to Speech Coding" 1986.

[20] Joel R.Crosmer," Very Low Rate Speech Coding Using the Line Spectrum Pair Transformation of the LPC Coefficients" PHD thesis ,The Georgia Institute of Technology, June 1985.

[21] F.K.Soong and B.H.Juang," Line spectrum pair (LSP) and speech data compression, " Proc.ICASSP-84, pp.1.10.1- 1.10.4, 1984.

[22] B.S.Atal,V.Cox and P.Kroon, "Spectral Quantization and Interpolation for CELP Coders". Proc. ICASSP-89, pp.89-72, 1989.

[23] Sugamura and Farvardin, " Quantizer Design in LSP Speech Analysis Synthesis," IEE Jon Sel. Areas in Comm, Vol 6, pp. 432-440,1988.

[24] Soong and Jauang, "Optimal Quantization of Line Spectrum Pair (LSP) Parameters," Proc. ICASSP-88, pp.394-397, 1988.

[25] F.ItaKure and N. Sugamura, " speech analysis and synthesis methods developed at ECL in NTT-FROM LPC to LSP," Speech Committee vol.5. pp.199-215, June 1986.

[26] J.Makhoul," Linear Prediction: A Tutorial Review" IEEE Trans. Audio Electroacoust, vol.63. No.4, April 1975.

[27] B.S.Atal," Predictive Coding of Speech Signals and Subjective Error Criteria" IEEE Trans. Acoust., Speech, Signal Processing, vol. ASSP-27, pp.247-254 June 1979.

[28] B.S.Atal," Predictive Coding of Speech at Low Bit Rates" IEEE Trans. Commn. Vol. COM-30, NO.4, APRIL 1982.

[29] S.T. Alexander," Adaptive Signal Processing - Theory and Applications" Springer-Verlag New York .

[30] Bishnu S. Atal and Joel R. Remde " A New Model Of LPC Excitation For reducing Natural-Sounding Speech At Low Bit Rates" Proc. IEEE-ICASSP, pp. 614-617. 1982.

[31] Lin D " New approaches To Stochastic Coding Of Speech Sources At Very Low Bit Rates" SOGNAL PROCESSING III: Theories and Applications. I.T.Young et Al. (editors) pp. 445-447, @ EURASIP,1986.

[32] R.C.Rose " Design and Performance of an Analysis-by-Synthesis Class of Predictive Speech coders" IEE Transactions on Acoustics, Speech and Signal Processing. VOL. 38 NO 9. pp 1489-1503, September 1990.