A 4.8 Kbps Multi-Band Excitation Speech Coder

by

John C. Hardwick

B.S., Massachusetts Institute of Technology, Cambridge, Ma

(1986)

Submitted in Partial Fulfillment
of the Requirements for the
Degree of

Master of Science
in Electrical Engineering and Computer Science

at the

Massachusetts Institute of Technology

May 1988

Signature of Author _

Department of Electrical Engineering and Computer Science
May 13, 1988

Certified by _

Professor Jae S. Lim
Thesis Supervisor

Accepted by _

Arthur C. Smith
Chairman, Departmental Committee on Graduate Students

A 4.8 Kbps Multi-Band Excitation Speech Coder

by

John C. Hardwick

Submitted to the
Department of Electrical Engineering and Computer Science
on May 13, 1988 in partial fulfillment of the requirements
for the Degree of Master of Science

Abstract

This thesis presents a high quality speech coding system. The system is based on the Multi-Band Excitation (MBE) speech model as developed by Griffin [2]. The system divides speech into 25 ms. segments, and for each speech segment a set of MBE speech model parameters are estimated. These parameters are then quantized, transmitted and finally used to synthesize a speech signal. The primary advantage of this approach is that the modelling process reduces the amount of information needed to represent speech. Unfortunately, this approach often causes a loss of speech quality, due to the limitations of the speech model. One of the primary advantages of the MBE speech model is that it can produce substantially higher quality speech, especially when the speech is in the presence of background noise. The MBE speech model uses a more flexible representation for the excitation sequence, thereby eliminating the "buzziness" traditionally associated with model-based coding of noisy speech.

The primary emphasis of this thesis is the quantization of the MBE model parameters. Earlier work resulted in MBE speech coders producing high quality speech at 9.6 kbps and 8 kbps. This thesis explores the hypothesis that the use of more complex quantization methods can allow a major reduction in the bit rate while maintaining high speech quality. This research shows that there are substantial inter-dependencies which exist between the model parameters. Quantization algorithms which exploit these dependencies can be used to achieve high quality speech at 4.8 kbps.

Thesis Supervisor: Jae S. Lim

Title: Associate Professor of Electrical Engineering

Dedication

To my parents,

for their love and support.

Acknowledgments

I would like to thank all of the people who have helped me both directly and indirectly during the completion of this thesis.

In particular I would like to thank Dan Griffin for his time and patience in answering my questions. In addition I would like to thank my thesis advisor, Professor Jae Lim, for his guidance and comments during the course of this thesis. These two individuals have made a major contribution to the development of this thesis. The entire membership of the M.I.T.'s Digital Signal Processing Group also deserves praise for their patience and helpfulness. The ability to learn in a friendly environment is a feature which cannot be overly appreciated.

The people who have helped me indirectly also deserve attention. This is particularly true of my family, which has been extremely supportive and attentive. The education which I have received is a result of their sacrifice, and my appreciation and love will always be with them. I am also thankful for the friendships which have made my education here a truly rewarding experience.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation

In a typical speech analysis/synthesis system (vocoder), a model is used to characterize the speech signal. In speech analysis a set of model parameters is estimated, and then in speech synthesis these parameters are used to generate a synthetic speech signal. This process is illustrated in Figure 1.1.

One important application area for vocoders is speech coding. Speech is generally regarded to have an effective bandwidth of 4 kHz. If speech is sampled at the Nyquist frequency of 8 kHz and quantized using 8 bit samples, then a bit rate of 64 kbps is required. Significant reductions in the bit rate can be achieved through the use of a vocoder. The number of bits which are needed to characterize the model parameters is considerably less than the bits needed to represent the actual samples. The exact bit rate which can be achieved is dependent upon the required speech quality and the particular vocoder which is being used. Reduction of the bit rate past a certain point will cause significant defects in any vocoder system. In addition performance is ultimately restricted by the underlying speech model. In order to be successful, a speech model must be flexible enough to reproduce the wide variety of sounds found in typical speech. Also in most applications some amount of background noise will be present. Therefore it is desirable for a speech model to be robust with respect to the signal-to-noise ratio. A highly accurate and robust speech model is essential for the production of high quality vocoder speech.

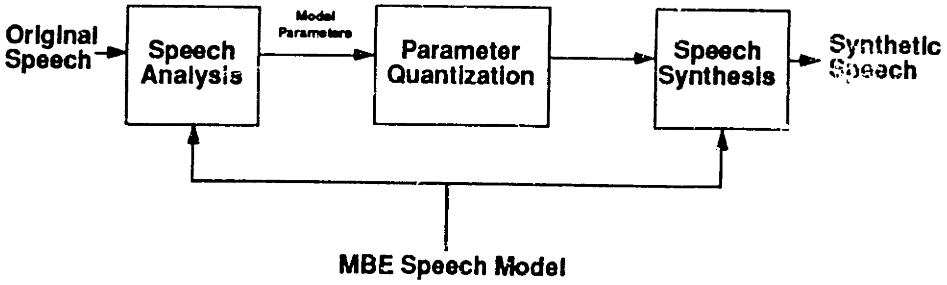

Figure 1.1: Vocoder Block Diagram

1.2 Background

Most current vocoders model speech as the response of a time varying linear filter to some excitation sequence. Since speech is not a stationary signal it is necessary to analyze speech over time intervals which are small compared to its time rate of change. This is accomplished by multiplying the speech signal by a window which is zero outside of some designated interval. By shifting the location of the non-zero portion of the window over the speech signal, it is possible to segment the entire signal. The vocoder analyzes each segment by estimating the parameters which characterize the linear filter and the excitation sequence for that segment. This is facilitated by dividing speech segments into two classes depending on the nature of the excitation sequence. In voiced speech the excitation is modeled as a periodic impulse train, while in unvoiced speech a white noise sequence is used as the excitation. Using this distinction the excitation parameters for each segment consist of a voiced/unvoiced decision and a pitch period, or fundamental frequency, for voiced speech. The synthesis system uses these excitation parameters to generate either a white noise sequence for unvoiced speech or an impulse train with the desired period for voiced speech. This sequence is then passed through a system which is specified by the linear filter parameters, and the output is taken as the synthesized speech.

Vocoders based on the model described above vary primarily by the manner in which they extract the model parameters. Different techniques are exemplified by linear prediction vocoders, homomorphic decoders and channel vocoders. While some techniques out-perform others, due to their superior ability to estimate the model parameters, they are all limited by the validity of the underlying model. Although this model has proved sufficient to allow the synthesis of intelligible speech, it has not resulted in a system capable of producing high quality speech. In an attempt to circumvent this limitation, considerable work has been done to improve the correlation between the model and actual speech. In [5,2] a new model is presented which does not perform a binary voiced or unvoiced classification of the excitation sequence. Instead each speech segment is modeled using a partially voiced and partially unvoiced excitation. This Multi-Band Excitation (MBE) speech model provides more flexibility in the selection of the excitation sequence, consequently, it can be used to generate higher quality speech. In addition a new algorithm for estimating the pitch and spectral envelope has been developed in [2,6] which offers superior performance over previous methods. By combining these techniques a more robust and accurate vocoder can be generated, which can potentially be used in a number of applications requiring high quality speech.

One important application of the MBE speech model is in speech coding. Speech coding systems have typically fallen into two broad classes. At higher bit rates, systems which reproduce the speech waveform on a sample by sample basis have prevaled. These "waveform coders" typically operate above about 10 kbps. At these rates there are sufficient bits to provide a reasonably accurate characterization of each sample of the speech signal. Often this characterization takes the form of a difference between the actual sample value and some predicted sample value. However, in other systems the samples are characterized in the frequency domain, or through some other representation. The primary advantage of these techniques is that they do not force any structure on the output signal, thereby enabling it to code both speech and non-speech signals. Given sufficient bits a "waveform coder" can provide high quality speech and a high degree of robustness to background noise. The traditional problem with these systems is that they have been restricted to higher bit rates. At lower bit rates there are not enough bits to accurately characterize each sample. The result is a sharp decrease in speech quality.

Very low bit rate speech coding systems have traditionally relied on vocoders. As mentioned previously, vocoders operate under the assumption that their input can be accurately represented by a speech model. Since the number of bits required to represent the model parameters is typically much smaller than the number of bits required to represent the samples, the vocoder can operate at much lower rates than a "waveform coder". The major problem with vocoders, however, has been that the degradations induced by the model were more significant than those due to quantization. For this reason vocoders were generally limited to very low bit rates, below about 4 kbps. These systems have been shown to be capable of producing intelligible speech, however they were not capable of producing high quality speech. In addition the performance of these systems deteriorates rapidly in the presence of background noise [1].

Several recent developments have extended the useful range of both "waveform coders" and vocoders to rates between 4 kbps and 10 kbps. Vector quantization has been applied to waveform coders and thereby reduced the required bit rate. The code-excited linear prediction (CELP) schemes found in [14] have been shown to be capable of producing high quality speech at these bit rates. Similarly new speech models have made corresponding improvements in the quality of vocoder speech.

The sinusoidal model described in [11] is an example. This system is very similar to the MBE speech model except that it lacks the voiced/unvoiced decisions. This difference is important in the context of coding noisy speech as discussed in [2].

The applicability of the MBE speech model to high-quality mid-rate speech coding has been shown in several systems. In [3] a 9.6 kbps MBE speech coder was demonstrated. A latter system which was described in [2] operated at 8.0 kbps. Both of these systems produced high quality speech in low and high Signal-to-Noise Ratio (SNR) conditions. The advantage of the MBE vocoder was most apparent from the lack of "buzziness" in the noisy speech. However in both systems the effects of quantization could be discerned. The most notable of these was an added reverberance in the speech of male speakers and a number of small artifacts in the speech of female speakers. The primary cause of these degradations was the lack of enough coded phase information and the low frame rate which was used.

The goal of this thesis was to investigate means of lowering the bit rate toward an eventual target of 4.8 kbps. As mentioned above, the effects of quantization were beginning to become apparent at 8 kbps. Since a rate of 4.8 kbps represented a significant reduction over that used in the earlier system, it was reasonable to assume the degradations would become much more prevalent. Early experimental evidence verified this hypothesis and showed that at 4.8 kbps the quality of the speech was unsatisfactory. In order to maintain a high quality speech capability, a number of alternative coding schemes were investigated. The techniques which were used to code the parameters in [2] did not fully utilize the redundancy in the speech parameters. A more efficient coding algorithm can exploit these additional dependencies. The problem is therefore reduced to finding a coding technique which is efficient enough to provide the desired speech quality at the desired rate. The remainder of this thesis is focused on this problem for the particular case of a MBE speech coder.

In the next chapter the MBE speech model is described. Later chapters deal with the use of this model in a speech coding system. Specifically, chapter 3 discusses speech analysis, while chapter 4 is dedicated to the synthesis of speech from the model parameters. Chapter 5 describes the quantization techniques which were used in the development of a 4.8 kbps MBE speech coder. Chapter 6 then presents the results which were obtained with this system for both informal listening tests and Diagnostic Rhyme Tests. Chapter 7 concludes this thesis with several ideas for future research.

Chapter 2

Multi-Band Excitation Speech Model

The Fourier Transform, $S_w(\omega)$, of a windowed speech segment, $s_w(n)$, can be modeled as the product of an excitation spectrum $E_w(\omega)$ and a spectral envelope $H_w(\omega)$. The primary difference between the MBE speech model and previous ones lies in the form of the excitation spectrum. In previous models the excitation is entirely specified by the fundamental frequency ω_o and a voiced/unvoiced decision for the entire speech segment. For voiced speech $E_w(\omega)$ is equal to $P_w(\omega)$, the Fourier Transform of a windowed impulse train with spacing equal to $M = 2\pi/\omega_o$. If the effects of aliasing are ignored, $P_w(\omega)$ corresponds to the Fourier Transform of the window sequence centered at each harmonic of ω_o. Speech segments which do not exhibit this periodic property are declared unvoiced. $E_w(\omega)$ for these segments is modeled as the spectrum of a windowed white noise sequence. This approach allows only two different types of excitation as shown in Figure 2.1. Consequently a conventional model's ability to represent the full range of speech signals is severely limited.

The extension which has resulted in the MBE speech model is to replace the binary voiced/unvoiced distinction with a more continuous division as shown in

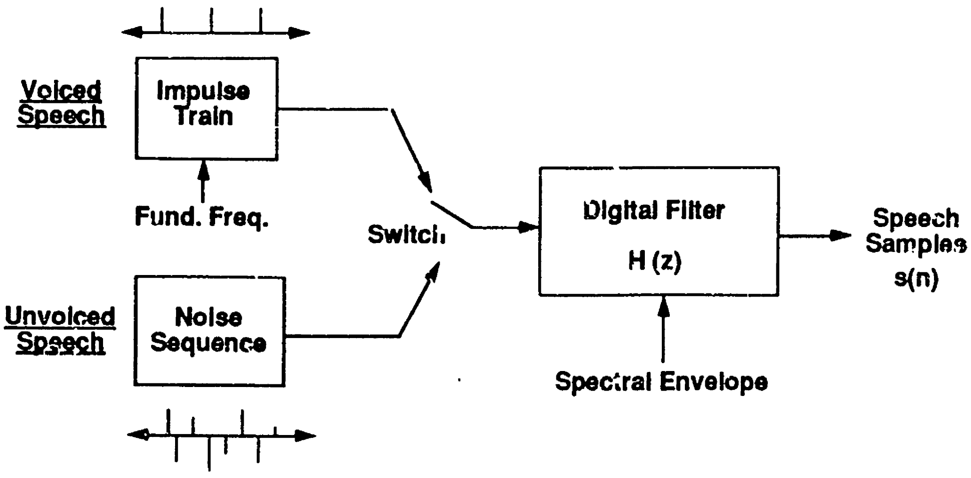

Figure 2.1: Conventional Speech Model

Figure 2.2. Instead of making a single voiced/unvciced decision for each speech segment, the new model divides the spectrum into a number of regions. Within each of these regions the speech spectrum is analyzed, and a voiced/unvoiced decision is made. The resulting excitation spectrum is a frequency dependent mixture of voiced and unvoiced energy. In each band which is declared voiced $P_w(\omega)$ is used in the excitation spectrum. The remaining frequency bands correspond to unvoiced regions, and they are filled with noise energy. The use of many voiced/unvoiced decisions allows the MBE speech model to have fine control over the makeup of the excitation spectrum.

This approach was motivated by the observation that many speech segments have some frequency regions which are dominated by noise energy while other regions are filled with periodic, voiced energy. This is especially the case in mixed voicing segments of clean speech and in voiced segments of noisy speech. In previous speech models these cases resulted in degradations in the quality of the synthesized speech. This degradation often took the form of a "buzzy" sound, which is due to the replacement of unvoiced energy with voiced energy. The flexibility provided by

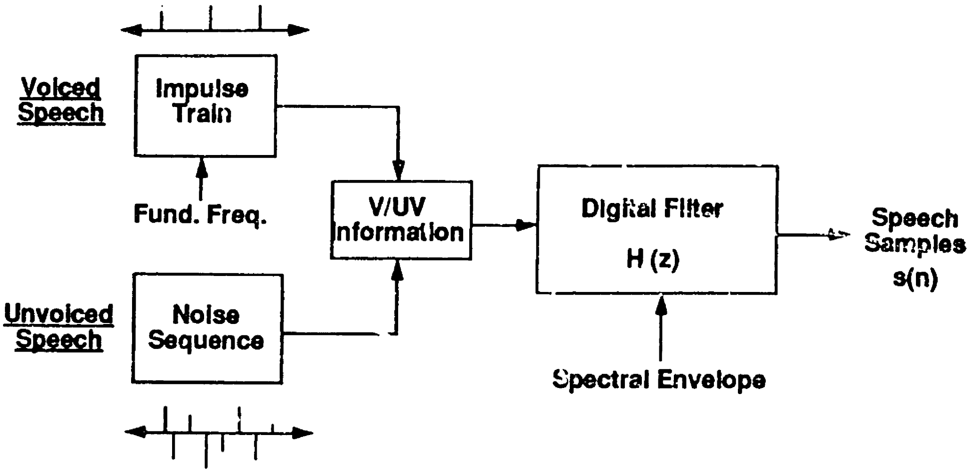

Figure 2.2: Multi-Band Excitation Speech Model

the MBE speech model allows it to avoid the "buzziness" typically associated with vocoders.

The spectrum of each speech segment is only partially determined by the excitation spectrum. In addition the spectral envelope $H_w(\omega)$ determines the relative amplitude of each frequency component. Since the excitation spectrum is assumed to have a constant amplitude, the spectral envelope provides the scaling between $E_w(\omega)$ and the actual speech spectrum. In this manner $H_w(\omega)$ can be viewed as the frequency response which will map $E_w(\omega)$ into $S_w(\omega)$. Since $H_w(\omega)$ is generally slowly varying, it is often assumed to be a smoothed version of the actual speech spectrum $S_w(\omega)$.

In Figure 2.3a the spectrum of a typical speech segment is shown. This was obtained by windowing the speech signal with a 256 point Hamming window and then calculating the Discrete Fourier Transform of the windowed sequence. Figure 2.3b shows the spectral envelope which has been found for the segment. One can see that this is a smooth contour containing the general shape of the spectrum shown in 2.3a. The pitch period which has been estimated for this segment is 78 samples at a 10 kHz sampling rate. $P_w(\omega)$ corresponding to this pitch period is shown in Figure 2.3c. The voiced/unvoiced information is displayed in Figure 2.3d. A high value on this graph corresponds to a voiced region of the spectrum where $P_w(\omega)$ would be used in the excitation spectrum. Frequency regions having a low value in Figure 2.3d are unvoiced and noise energy as shown in Figure 2.3e is used in the excitation spectrum. This combination of voiced and unvoiced spectra is then multiplied by the spectral envelope to create the synthetic speech. This product is shown in Figure 2.3f.

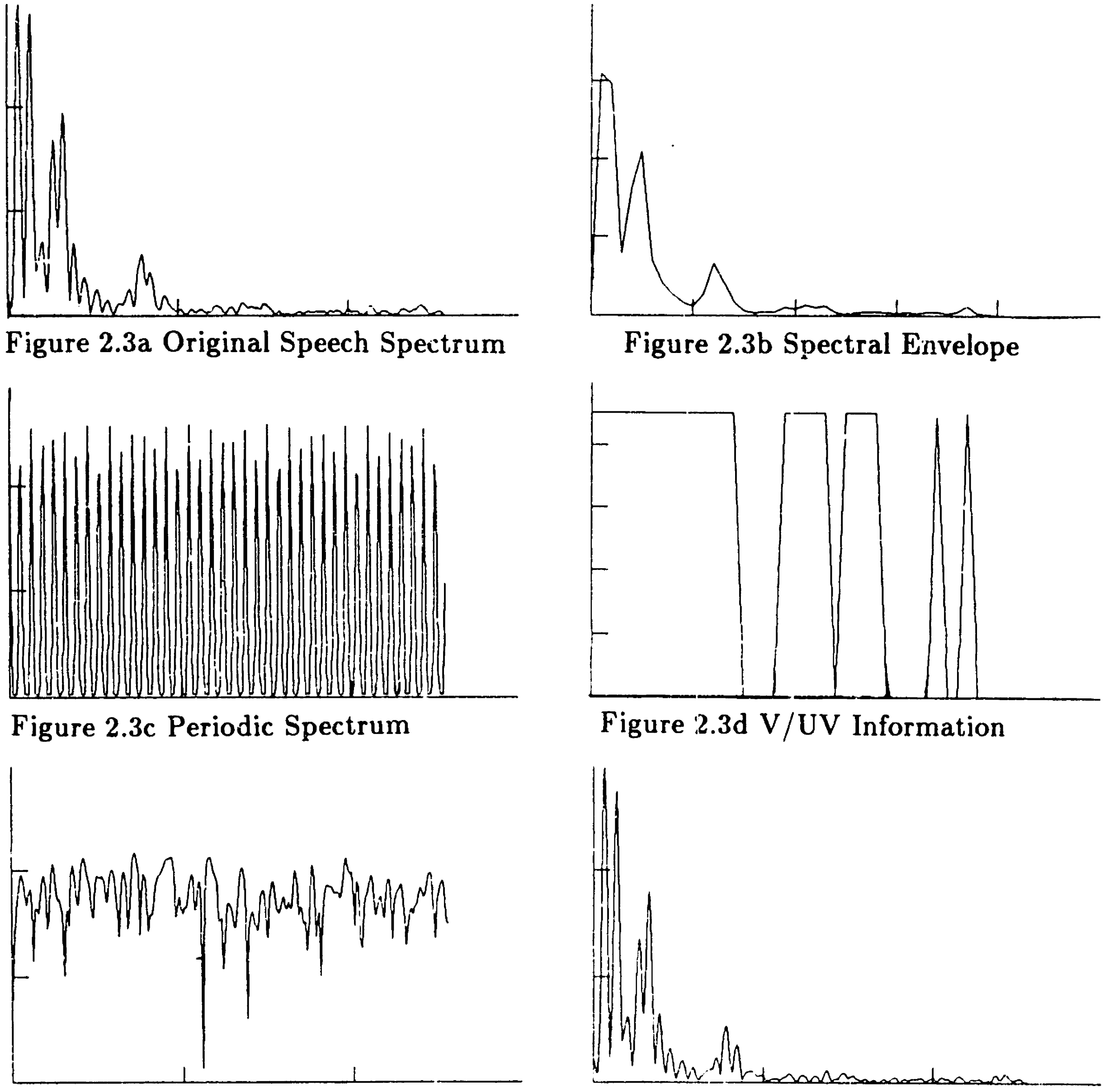

Figure 2.3a Original Speech Spectrum

Figure 2.3b Spectral Envelope

Figure 2.3c Periodic Spectrum

Figure 2.3d V/UV Information

Figure 2.3e Noise Spectrum

Figure 2.3f Synthetic Speech Spectrum

Figure 2.3: MBE Speech Spectra

Chapter 3

Speech Analysis

In order to produce high quality speech an accurate and robust algorithm must be used to estimate the model parameters. The technique described by Griffin in [2] has been shown to be very successful. This approach differs from previous ones in an important way. Instead of trying to estimate the excitation sequence and the spectral envelope separately, these parameters are found simultaneously. This approach allows the error between the original and synthetic speech spectra to be minimized in a least squared sense.

In general each speech segment can be either voiced, unvoiced or a mixture of the two. For unvoiced speech, the pitch period or fundamental frequency is not relevant, since the excitation spectrum is the transform of a white noise sequence. However, since most segments will be at least partially voiced, a fundamental frequency will be needed to characterize the periodic portions of the excitation spectrum. Therefore, the approach which is taken is to first assume the segment to be entirely voiced. The fundamental frequency and spectral envelope are then found which minimize the difference between the original speech spectrum and the synthetic spectrum. The result of this operation will produce small differences over voiced regions of the spectrum and large differences in the unvoiced regions. At this point the spectrum is broken up into bands which correspond to groups of harmonics of the chosen fundamental frequency. The error is evaluated within each of these regions and is used as the basis for a voiced/unvoiced decision. The model parameters for voiced areas have already been determined through the error minimization process. For unvoiced regions the spectral envelope must still be characterized. This is done by finding the average magnitude of the original speech spectrum around each harmonic within the region.

3.1 Pitch Estimation

In early MBE systems a frequency domain algorithm was used to estimate the pitch and spectral envelope simultaneously. This procedure matched the synthetic speech to the original speech in a least squares sense. In [6] the accuracy of this pitch estimate was shown to be considerably higher than that found with more conventional pitch estimators. This high degree of accuracy was found to be necessary in order to achieve reliable voiced/unvoiced determination [2]. An inaccurate pitch estimate can result in large differences between the original and synthetic speech. This is especially the case at higher frequencies where small pitch errors are accentuated. The system interprets this difference as a lack of voicing in the speech, and consequently voiced/unvoiced errors can be made.

The technique described above was shown to result in accurate pitch and spectral envelope estimates [5,3]. The primary disadvantage to this approach, however, was the large amount of computation needed to perform the estimation. The estimation is performed by minimizing the error between the original and synthetic speech spectrum for some assumed fundamental frequency. For each fundamental frequency there is an "optimal" synthetic spectrum and a corresponding minimal error. By evaluating this error over a large range of fundmental frequencies the best estimate of the pitch and the spectral envelope can be found. In practice this is accomplished in two stages. At first the error is evaluated on a coarse grid. The minimum of this error is found and the corresponding fundamental frequency is used as an initial estimate in the second stage. The error is now reevaluated at finer increments in a small band around the initial estimate. The fundamental frequency resulting in the minimal error in this stage is taken as the refined estimate. This procedure can be repeated on ever finer grids, until the desired accuracy is achieved. The problem lies in obtaining the initial estimate. In the first stage the error must be evaluated for a large number of fundamental frequencies, while in the second stage the error only needs to be evaluated at about 5 to 10 fundamental frequencies. Since the final accuracy of the estimate is determined by the second stage, a less accurate technique can be used to gain the initial estimate without adversely effecting the performance of the system.

The technique used to obtain the initial pitch estimate is based on an autocorrelation function. As will be shown below, this procedure is equivalent to the frequency domain method for integer pitch periods. In [2] the accuracy of this technique was shown to be sufficiently less than that achievable using the frequency domain technique. Although this accuracy was found to be insufficient for reliable voiced/unvoiced determination, it was shown to be sufficient for determination of the initial estimate. The main advantage of the autocorrelation technique is its computational savings. By using the Fast Fourier Transform (FFT) to perform the autocorrelation, the initial pitch estimate can be obtained very efficiently.

In the next subsections the two pitch estimation algorithms are presented. The frequency domain approach is shown first. The expression for the "optimal" spectral envelope for an assumed fundamental frequency will be derived. This will then be used to evaluate the resulting least-squared error. These results are then reformulated to produce the autocorrelation algorithm. The two expressions are shown to be approximately equal for integer pitch periods.

3.1.1 Frequency Domain Pitch Detection

In order to find the parameter set resulting in the minimum error between the original and synthetic speech it is necessary to solve a highly non-linear optimization problem. In general the error between the two can be expressed as:

$$E = \frac{1}{2\pi}\int_{\omega=-\pi}^{\pi} G(\omega)|S_w(\omega) - \bar{S}_w(\omega)|^2 d\omega \tag{3.1}$$

where $\bar{S}_w(\omega)$ is the synthetic speech spectrum and $G(\omega)$ is a frequency dependent weighting function. Since $\bar{S}_w(\omega)$ is equal to the product of $H_w(\omega)$ and $E_w(\omega)$, (3.1) can be rewritten as the following:

$$E = \frac{1}{2\pi}\int_{\omega=-\pi}^{\pi} G(\omega)|S_w(\omega) - E_w(\omega)H_w(\omega)|^2 d\omega \tag{3.2}$$

At this point several assumptions must be made in order to simplify the form of the error expression. The first is that the speech segment is entirely voiced. In addition the fundamental frequency ω_o is assumed to be known. Given this information the excitation spectrum takes the following form:

$$E_w(\omega) = P_w(\omega) = \sum_{m=0}^{M-1} W_m(\omega) \tag{3.3}$$

where $W(\omega)$ is the Fourier Transform of the segmentation window $w(n)$, $W_m(\omega) = W(\omega - m\omega_o)$, and $M = \lceil 2\pi/\omega_o \rceil$. For typical segmentation windows such as the Hamming window, $W(\omega)$ has an effective bandwidth which is less than ω_o. Consequently the effects of aliasing between $W_j(\omega)$ and $W_i(\omega)$ can be ignored for all $i \neq j$. In addition the impulsive nature of $W_m(\omega)$ allows the spectral envelope, $H_w(\omega)$ to be approximately characterized by a set of M complex scale factors. These scale factors are determined by error minimization around the region occupied by each harmonic. Using these assumptions, the error around the m^{th} harmonic can be expressed as the following:

$$E_m = \frac{1}{2\pi}\int_{\omega=a_m}^{b_m} G(\omega)|S_w(\omega) - A_m W_m(\omega)|^2 d\omega \tag{3.4}$$

where A_m is the complex scale factor associated with the m^{th} harmonic. The range of the integral is set such that it has a width equal to ω_o, centered at $m\omega_o$. From this information $a_m = (m - .5)\omega_o$ and $b_m = (m + .5)\omega_o$. By differentiating (3.4) with respect to A_m, the scale factor which minimizes the error over the region $a_m \leq \omega < b_m$ is obtained. This leads to the following expression for the optimum A_m:

$$A_m = \frac{\frac{1}{2\pi}\int_{\omega=a_m}^{b_m} G(\omega)S_w(\omega)W_m^*(\omega)d\omega}{\frac{1}{2\pi}\int_{\omega=a_m}^{b_m} G(\omega)|W_m(\omega)|^2 d\omega} \tag{3.5}$$

The optimum A_m can be used to evaluate the resulting minimum error. Substituting (3.5) into (3.4) and rearranging, the following expression is obtained for $E_{m_{min}}$, the minimum error over the m^{th} harmonic:

$$E_{m_{min}} = \frac{1}{2\pi}\int_{\omega=a_m}^{b_m} G(\omega)|S_w(\omega)|^2 d\omega - \frac{|\frac{1}{2\pi}\int_{\omega=a_m}^{b_m} G(\omega)S_w(\omega)W_m^*(\omega)d\omega|^2}{\frac{1}{2\pi}\int_{\omega=a_m}^{b_m} G(\omega)|W_m(\omega)|^2 d\omega} \tag{3.6}$$

Since a_m and b_m are dependent upon the assumed fundamental frequency, the values of A_m and $E_{m_{min}}$ are also dependent on this value. For a given ω_o (3.5) can be used to generate the scale factors which minimize the error over a particular harmonic region. These scale factors can then be used to evaluate the minimum error as shown in (3.6). This process can be extended to find the minimum error over the entire spectrum. This quantity is denoted by $E_{T_{min}}$ and is given by:

$$E_{T_{min}} = \sum_{m=0}^{M-1} E_{m_{min}} \tag{3.7}$$

where $E_{m_{min}}$ is defined by (3.6).

Equation (3.7) can be used to select the best fundamental frequency out of a set of hypotheses $E_{T_{min}}$ is calculated for each possible value of ω_o. This information is then processed to find the best hypothesis. The value of ω_o which corresponds to this hypothesis is then used with equation (3.5) to calculate the optimum scale factors.

3.1.2 Autocorrelation Pitch Detection

The minimum error as a function of ω_o is given in the frequency domain by (3.7). This expression can be reformulated in the time domain, yielding an alternative pitch estimation algorithm. This time domain approach is approximately equal to the frequency domain approach for integer pitch periods. In addition, an efficient implementation can be found, which gives the time domain algorithm a substantial computational advantage.

Subsequent analysis is simplified if $G(\omega) = 1$ for $-\pi < \omega \leq \pi$. This condition does not cause a loss of generality since $S_w(\omega)$ and $W_w(\omega)$ can be prescaled accordingly. Substituting (3.6) into (3.7) gives the following expression for the minimum error over the entire spectrum:

$$E_{T_{min}} = \frac{1}{2\pi}\int_{\omega=-\pi}^{\pi} |S_w(\omega)|^2 d\omega - \Psi(\omega_o) \tag{3.8}$$

where $\Psi(\omega_o)$ is given by:

$$\Psi(\omega_o) = \sum_{m=0}^{M-1} \frac{|\frac{1}{2\pi}\int_{\omega=a_m}^{b_m} S_w(\omega)W_m^*(\omega)d\omega|^2}{\frac{1}{2\pi}\int_{\omega=a_m}^{b_m} |W_m(\omega)|^2 d\omega} \tag{3.9}$$

From (3.8), minimizing the error against ω_o is equivalent to maximizing $\Psi(\omega_o)$ against ω_o. In the limit where $E_{m_{min}}$ approaches zero, $\Psi(\omega_o)$ approaches the energy in the signal. For all other values $\Psi(\omega_o)$ will be less than the energy in the signal.

Since the bandwidth of the window function $W(\omega)$ is assumed to be less than ω_o, the limits on the integrals in equation (3.9) can be extended. This yields the following equation:

$$\Psi(\omega_o) = \sum_{m=0}^{M-1} \frac{|\frac{1}{2\pi}\int_{\omega=-\pi}^{\pi} S_w(\omega)W_m^*(\omega)d\omega|^2}{\frac{1}{2\pi}\int_{\omega=-\pi}^{\pi} |W_m(\omega)|^2 d\omega} \tag{3.10}$$

If the window is energy normalized, the following equation holds:

$$\frac{1}{2\pi}\int_{\omega=-\pi}^{\pi} |W_m(\omega)|^2 d\omega = 1 \tag{3.11}$$

Imposing this condition on the window function, and using Parseval's Theorem, a time domain expression for $\Psi(\omega_o)$ can be realized. The result is given by:

$$\Psi(\omega_o) = \sum_{m=0}^{M-1} |\sum_{n} s_w(n)w_m^*(n)|^2 \tag{3.12}$$

Since $W_m(\omega) = W(\omega - m\omega_o)$ and $s_w(n) = s(n)w(n)$, equation (3.12) can be expanded to yield:

$$\Psi(\omega_o) = \sum_{n}\sum_{k} s(n)s(k)w^2(n)w^2(k) \sum_{m=0}^{M-1} e^{-j\omega_o m(n-k)} \tag{3.13}$$

where $s(n)$ and $w(n)$ are known to be real sequences. If ω_o is constrained to only allow integer pitch periods, then $\omega_o = \frac{2\pi}{M}$, where M is equal to the pitch period. Substituting this result into equation (3.13), the following expression for $\Psi(M)$ is found,

$$\Psi(M) = \sum_{n}\sum_{k} s(n)s(n-kM)w^2(n)w^2(n-kM) = \sum_{k} \Phi(kM) \tag{3.14}$$

where,

$$\Phi(l) = \sum_{n} s(n)s(n-l)w^2(n)w^2(n-l) \tag{3.15}$$

Equation (3.14) shows that $\Psi(M)$ is a function of the autocorrelation sequence $\Phi(l)$. Since the autocorrelation of a sequence can be computed very efficiently with a FFT algorithm, $\Psi(M)$ can also be calculated very efficiently. From equation (3.8), the pitch period corresponding to the maximum of $\Psi(M)$ also yields the minimum error. Therefore $\Psi(M)$ can be used to form an initial estimate for the pitch period. One simple approach which could be used to form this initial estimate would be to choose the pitch period yielding the maximum value of $\Psi(M)$. A better technique, however, is to combine the information contained in $\Psi(M)$ with some restrictions on pitch continuity, thereby improving the pitch estimate.

3.1.3 Pitch Refinement

The function $\Psi(M)$, which is generated by the autocorrelation pitch estimation algorithm, is used in the generation of the pitch estimate. However, before the final estimate is obtained the data must undergo several additional processing steps. These steps include bias removal, pitch tracking, harmonic checking, and finally an increase in the accuracy of the estimate. Proper execution of these steps is important for accurate and robust pitch estimation.

As shown in [2], $E_{T_{min}}$ is biased such that longer pitch periods are favored over shorter pitch periods. If speech is modeled as a periodic signal in white noise, then the expected value of $E_{T_{min}}$ is given by the following:

$$E[E_{T_{min}}] \approx \sigma^2(1 - M \sum_{n=-\infty}^{\infty} w^4(n)) \tag{3.16}$$

where σ is the standard deviation of the white noise. This expression can be used to remove the bias from the error function as shown in [2].

A pitch tracking algorithm is used to process the unbiased error in order to improve the continuity of the pitch estimates. This algorithm uses the unbiased error from the current frame, several future frames and several past frames in order to find a pitch track with the minimum total error. Dynamic programming techniques as described in [12,2] are used to reduce the computational complexity of this algorithm.

The output of the pitch tracking algorithm is an initial estimate of the pitch period (or fundamental frequency) for the current speech frame. This pitch estimate is checked against its harmonic sub-multiples in order to ensure that the initial estimate is not a harmonic of the true pitch period. Figure 3.1 shows a typical graph of unbiased pitch error versus pitch period. As can be seen, the error is comparable at the true pitch period of 35 samples and at its near multiples of 69 and 104 samples. The pitch estimate must be chosen to equal the true pitch period of 35 samples, regardless of whether the initial estimate was equal to 35, 69 or 104 samples.

Once the multiples of the true pitch period have been discarded the accuracy of the new pitch estimate must be increased. As discussed earlier, the fundamental frequency, which is inversely related to the pitch period, must be known very accu-

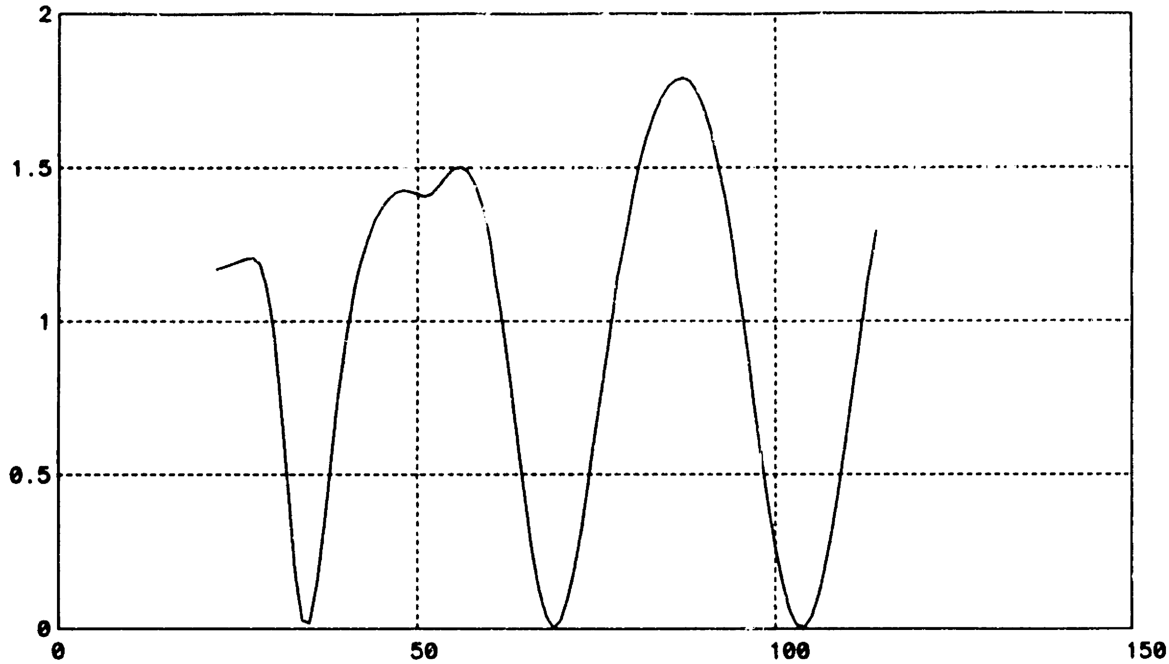

Figure 3.1: Error vs. Pitch Period

rately in order to make reliable voiced/unvoiced decisions. Since the autocorrelation pitch estimation algorithm is essentially restricted to integer pitch periods, it cannot provide sufficient accuracy. In [2] it was shown that a fundamental frequency accurate to within 1 Hz. was sufficient for reliable voiced/unvoiced determination. In addition it was shown that the frequency domain pitch estimation algorithm discussed above was capable of achieving this accuracy. The minimum error for this algorithm as a function of the estimated fundamental frequency is given by equations (3.7) and (3.6). This error function is evaluated over a small range of fundamental frequencies, which is centered at the inverse of the refined pitch estimate. The fundamental frequency which results in the minimum error is chosen as the final estimate.

3.2 Spectral Envelope Determination

As discussed previously, the frequency domain pitch estimation algorithm estimates the optimum complex scale factors, A_m, for a given fundamental frequency. Since these scale factors characterize the spectral envelope for voiced speech, only the unvoiced portions of the spectral envelope remain undetermined. In order to determine the spectral envelope for the unvoiced speech, the speech spectrum must first be divided into voiced and unvoiced regions. This process is described in the following section. Once a region has been determined to be unvoiced, the spectral envelope is characterized in that region by the average of the spectral magnitude. The phase of the spectral envelope is not obtained for unvoiced regions.

3.3 Voiced/Unvoiced Determination

In the attempt to minimize the error between the original and the synthetic spectrum the speech segment was assumed to be all voiced. The estimation algorithm uses this assumption to find the optimum fundamental frequency and complex scale factors. The voicing information is determined by first dividing the spectrum into regions which correspond to groups of three harmonics of the fundamental. Within each region a decision is made as to whether the speech spectrum is voiced or unvoiced. This determination is made by examining the error between the original speech spectrum and the estimated speech spectrum within this region. Since the estimated spectrum assumes the speech to be voiced, the error will be low in voiced regions and high in unvoiced regions. Therefore the error can be used to make the voiced/unvoiced decision by comparing it against a predetermined threshold.

The value of the threshold level is set to give the proper mix of voiced and unvoiced energy. If the threshold is set too high, then the speech takes on a hollow, reverberant sound due to the predominantly voiced nature of the speech. Similarly, if the threshold is set too low, then the speech takes on a hoarseness due to the large amounts of unvoiced energy. Listening tests can be used to set the threshold to the point where the ratio of voiced to unvoiced energy is perceptually optimal.

The value of the threshold is extremely important for high quality synthesis of noisy speech. Early experiments with noisy speech showed a tendency for unwanted voiced energy to appear at high frequencies. Lowering the threshold to eliminate this effect caused the quality of the clean speech to decrease. In order to resolve this problem the fixed threshold was replaced with one which was a fixed function of frequency. The threshold function which is shown in Figure 3.2 was found to solve the aforementioned problem. Voiced/unvoiced decisions are made by comparing the normalized error for each region with the value of the threshold at the center of that region.

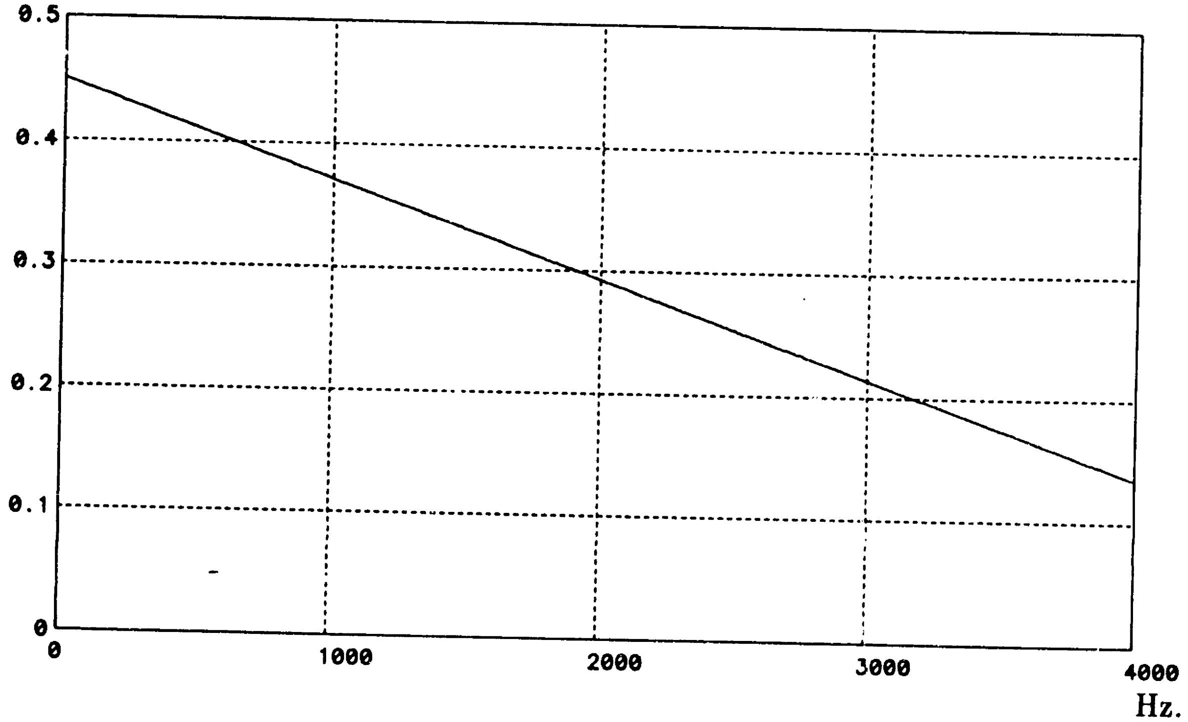

Figure 3.2: Voiced/Unvoiced Threshold Function

Chapter 4

Speech Synthesis

The problem of synthesizing a speech signal from the MBE model parameters is discussed in detail in [2]. The basic approach is to separate the speech signal into its voiced and unvoiced components. The two components are then synthesized separately and finally combined to produce the complete speech signal. The algorithms which are used to synthesize the voiced and unvoiced portions of the speech are based on two very different techniques. The remainder of this chapter provides an overview of these algorithms.

4.1 Voiced Synthesis

For each speech frame the analysis algorithms estimates a set of parameters. These parameters consist of a fundamental frequency, the voiced/unvoiced information and a set of harmonic magnitudes and phases. Since voiced speech is modeled as being discrete in frequency, the synthesis procedure can be implemented as a bank of tuned oscillators. For a particular speech segment, an oscillator is assigned to each harmonic which has been declared voiced. Using this approach the voiced speech synthesis problem reduces to finding the amplitude and phase function for each oscillator. One simple solution to this problem would be to set the amplitude equal to the current harmonic magnitude, and to use a linearly varying phase function, specified by the harmonic phase and fundamental frequency of the current frame. The problem with this approach is that it causes discontinuities at the edges of each speech segment. This is due to the fact that the voiced portion of the speech is not periodic over intervals consisting of several analysis frames. Variations in the estimated parameters at consecutive frames can cause amplitude and phase discontinuities in the synthesized speech. These discontinuities result in a substantial degradation of speech quality. In order to solve this problem it is necessary to base the amplitude and phase functions on the estimated parameters for the current and the future frame. These functions are determined in such a manner as to ensure that the voiced speech is continuous at the frame boundary.

The output of the oscillator for the m^{th} harmonic can be expressed as:

$$\bar{s}_{v_m}(t) = A_m(t) \cos \theta_m(t) \tag{4.1}$$

where $A_m(t)$ is the amplitude function for the m^{th} harmonic and $\theta_m(t)$ is the phase function for that harmonic. In order to ensure continuity at the beginning and end of a speech segment, the amplitude function $A_m(t)$ is linearly interpolated between the estimated value at the current segment and the estimated value one segment in the future. If the time between speech segments corresponds to $t = S$ then the amplitude function is given by:

$$A_m(t) = A_m(0) + [A_m(S) - A_m(0)]\frac{t}{S} \tag{4.2}$$

where $A_m(0)$ is the estimated harmonic magnitude for the current segment, and $A_m(S)$ is the value of the corresponding harmonic magnitude one frame in the future. If the m_{th} harmonic for either of the two segments is declared unvoiced then its associated magnitude component in (4.2) is set to zero. This process ensures a smooth transition as a harmonic changes from voiced to unvoiced, or vice-versa.

The phase function for the m_{th} harmonic, $\theta_m(t)$, is determined by the model parameters in a manner very similar to the amplitude function. In general the phase function between two speech frames can be expressed as:

$$\theta_m(t) = \int_0^t \omega_m(\xi)d\xi + \phi_m \tag{4.3}$$

This expression separates the phase function into a frequency track, $\omega_m(t)$, and an initial phase ϕ_m. At $t = 0$ or $t = S$ the frequency is defined as the m_{th} harmonic of the fundamental frequency which was chosen for that segment. Since the frequency at these two points may not be identical, it is desirable to interpolate the frequency in between. This can be accomplished in the following fashion:

$$\omega_m(t) = m\omega_o(0) + m[\omega_o(S) - \omega_o(0)]\frac{t}{S} + \Delta\omega_m \tag{4.4}$$

In order to match (4.3) with the known phase value at $t = 0$ and $t = S$, the variables ϕ_m and $\Delta\omega_m$ must be properly chosen. If the m_{th} harmonic has been declared voiced for both speech segments, then the initial phase, ϕ_m, is set equal to the estimated phase at the segment corresponding to $t = 0$. The frequency deviation, $\Delta\omega_m$, then corresponds to the smallest value which will result in $\theta_m(t)$ having the phase which was estimated at $t = S$. If the harmonic at either frame was declared unvoiced then ϕ_m is set equal to the estimated phase of the voiced segment. Since the phase of the unvoiced segment is irrelevant the frequency deviation is set to zero.

The phase function $\theta_m(t)$ which is determined by equations (4.3) and (4.4) is a quadratic polynomial in t. Since a quadratic polynomial is completely specified by three parameters, it can only be made to satisfy three arbitrary boundary conditions. However, the phase and frequency are specified at both $t = 0$ and $t = S$, yielding a total of four boundary conditions if both harmonics are voiced. Since a quadratic phase function is not able to meet all four conditions, one or more of the boundary conditions must be perturbed. This is done in equation (4.4) through the inclusion of the variable $\Delta\omega_m$. This variable is set such that the phase boundary conditions are matched exactly. As a consequence both frequency boundary conditions are slightly perturbed. Although it is possible to change only one of the frequency boundary conditions, the magnitude of the resulting alteration would have to be twice as large. Changing both equally minimizes the maximum perturbation.

An alternative approach to the construction of the phase function is to use a cubic polynomial [11], which can match the four boundary conditions exactly. Several tests were conducted to compare the quadratic and cubic phase functions. The result of these tests showed that there was no noticeable difference between the two techniques. This can be explained by the fact that for most speech segments the variable $\Delta\omega_m$ is very small in comparison to the fundamental frequency. In this situation the perturbation in the frequency track is not perceivable.

Once the oscillator parameters, $A_m(t)$ and $\theta_m(t)$, have been calculated for each harmonic, the voiced portion of the speech signal between $t = 0$ and $t = S$ is formed. This is accomplished by adding the contributions of each harmonic oscillator. The voiced speech component is given by the following:

$$\bar{s}_v(t) = \sum_{m=0}^{M-1} A_m(t) \cos\theta_m(t) \qquad (4.5)$$

4.2 Unvoiced Speech Synthesis

The unvoiced component of the speech is generated from the harmonics which are declared unvoiced. The synthesis algorithm uses a large Gaussian noise sequence as a reference signal. For each speech segment the corresponding section of the reference signal is windowed and transformed with a Fast Fourier Transform (FFT). The regions of this spectrum which correspond to voiced harmonics are set equal to zero. The remaining regions of the spectrum correspond to the unvoiced harmonics. In these regions the average magnitude is set equal to the value which was estimated during speech analysis. The phase in these regions is not modified and, therefore, corresponds to the phase of the original noise sequence. The inverse transform of this modified noise spectrum corresponds to the unvoiced speech for that segment. However, because the length of the synthesis window is longer than

S, the unvoiced speech for each segment overlaps that of neighboring segments. The weighted overlap-add procedure is used to average these sequences in the overlapping regions. This technique, which is described in [7], is a method for generating a signal from its Short-Time Fourier Transform (STFT). When applied to unvoiced speech synthesis, it averages the overlapping sequences so that the result has a STFT which is as close as possible to the modified noise spectra. The result can then be added to the voiced speech component to complete the speech synthesis procedure.

4.3 Speech Synthesis Evaluation

The techniques described in this and the previous chapter have been used to create a high quality speech analysis/synthesis system. The performance of this system has been found to be very good for both clean and noisy speech. Although it is possible to discern the original from the synthesized speech, the analysis/synthesis system can produce a natural sounding replica with virtually no degradation. However, in order to obtain this high quality speech, the distance between analysis frames, S, must be kept sufficiently small. If S is too large, then the frame to frame variation in the speech will be too great for the synthesis algorithm to accurately reproduce. Consequently, the synthesized speech lacks the clarity of the original speech. For small values of S the speech varies gradually from frame to frame. This condition is necessary in order for the amplitude and phase functions defined in (4.2) and (4.3) to accurately interpolate the speech in between speech segments. One disadvantage of a small value of S is that the number of parameters which are estimated per unit time increases in inverse proportion to its value. This can complicate the use of the system in such applications as speech coding.

One important aspect of any speech analysis/synthesis system is its associated delay. Due to the presence of speech segments, or frames, which are processed in blocks, some samples of the speech cannot be synthesized until more future samples are obtained. For the system described in this thesis the largest portion of the delay is caused by the pitch tracking algorithm used in speech analysis. The three frame look-ahead feature results in 60 ms. of delay for $S = 20$ ms. There is an additional 37.5 ms. of delay induced by the size of the analysis and synthesis window. This yields a total delay of approximately 100 ms.

Chapter 5

Multi-Band Excitation Speech Coding

5.1 Introduction

One application of the MBE speech model is in speech coding. An MBE speech coder operates by first estimating the MBE model parameters as described in Chapter 3. These parameters are then quantized and transmitted. At the receiver, the quantized parameters are reconstructed, and then used to synthesize speech in the manner described in Chapter 4.

The quality of the coded speech is limited by two factors. The first is the accuracy of the speech model, and the other is the distortion induced by quantization of the model parameters. Since quantization can only degrade the system's performance, the highest quality which can be achieved is found in the absence of quantization. As discussed in the previous chapter, the unquantized model parameters can be used to synthesize very high quality speech. Therefore, given sufficient bits, an MBE speech coder can do equally well.

A variety of quantization techniques exist, of which many could be used to quantize the MBE model parameters. These different techniques all offer a unique combination of advantages and disadvantages. The choice of which techniques should be used depends on the bit rate at which the system is designed to operate, and on the relative importance of speech quality versus computation, storage, and delay. The system which was designed as part of this thesis was required to have high quality at 4.8 kbps, while maintaining reasonable computation and storage requirements. In addition the coding delay was restricted to around 100 ms. These requirements were set in such a manner that the resulting system would be applicable to real-time speech communication.

The remainder of this chapter is used to describe the 4.8 kbps MBE speech coding system which was designed as part of this thesis. First a description of previous MBE speech coding work is given. Then the problems of applying these previous techniques to a 4.8 kbps system are presented. This is followed by a short discussion of several alternatives which could be used to solve these problems. The chapter concludes with a detailed description of the 4.8 kbps MBE speech coder which was developed.

5.2 Background

The MBE speech model has been used in the development of several speech coding systems. Griffin first described a 9.6 kbps MBE speech coder in [3]. His later work included an 8 kbps MBE speech coder which is described in [4,2]. These two systems use analysis and synthesis algorithms which are very similar to the ones which have been described in this thesis. The quantization of the model parameters is done in a slightly different manner in the 8 kbps system in comparison to the 9.6 kbps system, however, the general techniques are the same.

The 8 kbps system mentioned above provided the starting point for much of the work done in this thesis. This system was designed to operate with 4 kHz. bandwidth speech sampled at 10 kHz. The analysis was done every 20 msec., yielding a parameter frame rate of 50 Hz. At this frame rate 160 bits were available for the coding of the model parameters. These bits were divided between the fundamental frequency, the voiced/unvoiced decisions, and the harmonic magnitudes and phases. These parameters were then quantized using the assigned number of bits. The resulting bit stream is then passed to the decoder/synthesis system which reproduces the speech.

In this system the number of harmonic magnitudes and phases is a function of the fundamental frequency and the voiced/unvoiced information. Therefore, the fundamental frequency and voiced/unvoiced information are encoded first, allowing the decoder to determine the correct bit assignment for the remaining parameters. The fundamental frequency in this system is quantized to 1 Hz. increments between 80 Hz. and 500 Hz. This quantized value is then encoded using a fixed length, nine bit, codeword. The voiced/unvoiced information is obtained by dividing the spectrum into 12 regions, and a binary voiced/unvoiced decision is made for each region. These decisions are then encoded using a single bit per decision, yielding a total of 12 bits.

The next parameters to be quantized are the harmonic phases. Since phase information was thought to be most important for low frequency harmonics, and since the phase of unvoiced harmonics is not needed by the synthesis algorithm, phase information is only retained for voiced harmonics which lie in the range of 1 to 12. The phase of these harmonics is quantized by forming a predicted phase based on the previous phase and frequency information. A phase residual is then found as the difference between the actual phase and the predicted phase. This residual, $\Delta\theta_m$ is equal to:

$$\Delta\theta_m = S \cdot \Delta\omega_m \tag{5.1}$$

where $\Delta\omega_m$ is defined in equation (4.4) and S is equal to 20 msec., the distance between analysis frames. The reason for quantizing the phase residual, instead of the phase itself, is that the phase residual has a lower variance, thereby allowing more efficient quantization [2]. A 13 level, non-uniform Max-Lloyd quantizer is used to take advantage of the lowered variance. The total number of bits which are used for quantizing the phase depends on the number of voiced harmonics in the range 1 to 12. If all 12 harmonics are voiced then the maximum of 45 bits are used for coding the phase information. However, if all 12 harmonics are declared unvoiced then no bits are required for the phase information. In general the bit requirement for the harmonic phases is somewhere between these two extremes.

The harmonic magnitudes are quantized last, using all of the bits remaining after the fundamental frequency, the voiced/unvoiced decisions and the harmonic phases have been quantized. Due to the variable number of bits required to code the harmonics phases, the number of bits available for the harmonic magnitudes varies between 94 and 139. The harmonic magnitudes are quantized by first distributing the available bits over all of the harmonic magnitudes. This is done by integrating the bit density curve shown in Figure 5.1 over the region occupied by each harmonic. The percentage of the curve within each harmonic region corresponds to the percentage of the available bits which are assigned to quantize that harmonic magnitude. An important feature of this bit assignment procedure is that it assigns more bits to the low frequency harmonic magnitudes, than to the higher frequency harmonic magnitudes. This feature reflects the fact that the long term power spectrum of speech has a low-pass characteristic [13]. Assigning more bits to low frequency harmonics can provide for a reduction in the quantization error, averaged over all of the harmonic magnitudes.

Once the bits have been assigned to each harmonic magnitude, their values are quantized in a manner similar to that used by channel vocoders [8]. In this scheme the log magnitude of the first harmonic is quantized. Then the difference between the log magnitudes of each succeeding pair of harmonics is quantized. The quantizers are all uniform "mid-rise" quantizers, with the step size being a function of the number of assigned bits. This function, which is tabulated in Table 5.1, was

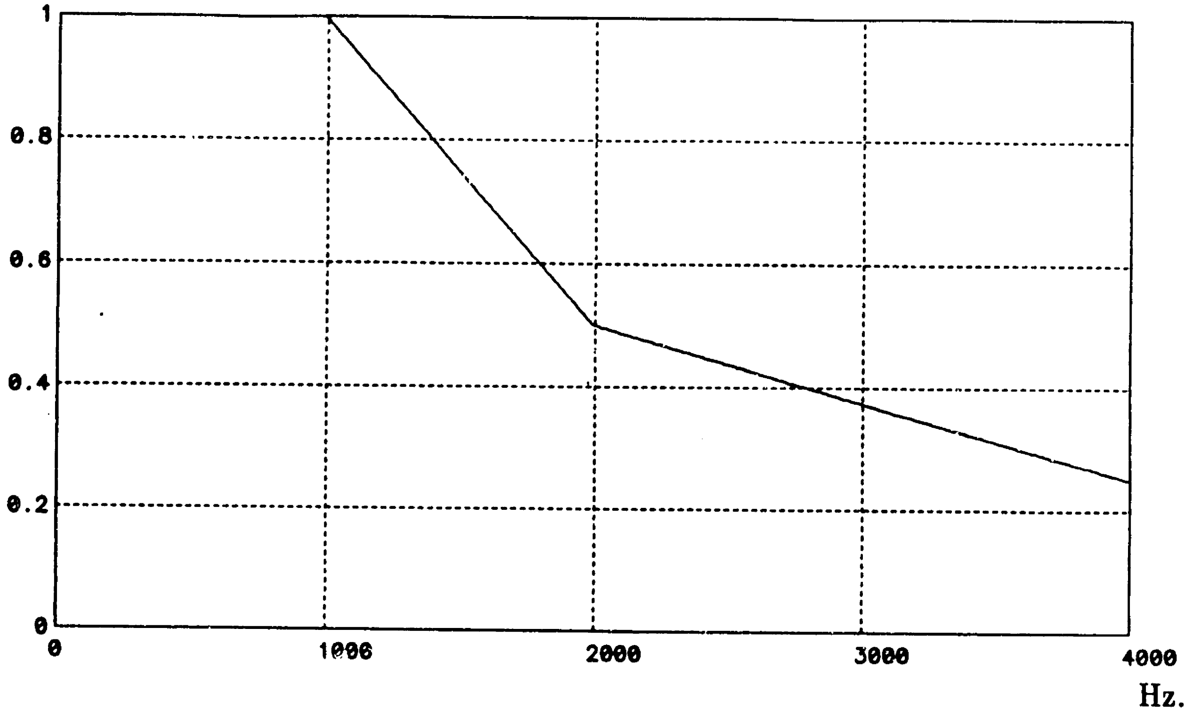

Figure 5.1: Bit Density Curve for Harmonic Magnitudes

found in [2] to provide good results for the quantization of the harmonic magnitudes.

The coding techniques, which are described above, were found in [2] to provide high quality speech at a rate of 8 kbps. Informal listening tests were used as a basis for this quality assessment. More formal testing was done to determine the intelligibility of the system. A series of Diagnostic Rhyme Tests (DRT) were done in order to quantitatively measure the intelligibility of the system. The results for both clean and noisy speech are given in Table 5.2. Additional tests, documented in [2], provide a comparison between the 8.0 kbps MBE vocoder and a more traditional vocoder with only a single voiced/unvoiced decision per frame. The major conclusion which was gained from the DRT results is that the MBE vocoder could provide highly intelligible speech. In addition the degradation of intelligibility in the presence of background noise was substantially less for the MBE vocoder than for the more traditional vocoder.

Bits	Step Size (dB.)	Min (dB.)	Max (dB.)
1	8	-4	4
2	6.5	-9.75	9.75
3	5	-17.5	17.5
4	3	-22.5	22.5
5	2	-31	31
6	1	-31.5	31.5
7	0.5	-31.75	31.75
8	0.25	-31.875	31.875

Table 5.1: Step Sizes for Harmonic Magnitudes

Although the quantization techniques described in the previous section resulted in satisfactory quality at 8 kbps, the quality of the coded speech was found to degrade quickly as the bit rate was reduced. At a frame rate of 50 Hz., a 4.8 kbps speech coder can only use 96 bits per frame. Quantization of the MBE model parameters with this number of bits, using the same methods as at 8 kbps, resulted in seriously degraded speech. The best system which could be achieved with these methods had two major problems. First it was considerably more reverberant and "buzzy", due to the reduction in the amount of quantized phase information. In addition, exceedingly coarse quantization of the harmonic magnitudes caused the speech to sound weak, especially in the presence of background noise. One simple

System	Mean Score	Std. Dev.
Original Clean Speech	96.9	0.28
8 kbps Clean Speech	93.6	0.53
Original Noisy Speech	51.7	1.1
8 kbps Noisy Speech	49.6	1.1

Table 5.2: DRT Results for 8 Kbps MBE Coder

solution to this problem is to reduce the frame rate, thereby yielding more bits per frame. However, experimental results showed that lower frame rates cause a loss of clarity in the speech, even in the absence of quantization. The presumed effect of this degradation is a loss of intelligibility, regardless of the quantization scheme which is employed. Since one goal of this research is to maintain performance comparable to that found in the 8 kbps system described above, the frame rate was left fixed at 50 Hz. The resulting problem is then to develop a speech coding system which could yield high quality performance at 96 bits per frame.

The solution to this problem was found by examination of the MBE model parameters. Experimental evidence showed that there are substantial inter-dependencies amongst the model parameters which were not being exploited by the current quantization algorithms. A well known principle of information theory states that the efficiency of a quantization algorithm can be improved by reducing the amount of redundancy which is present in the data. An improvement in the efficiency of a quantizer corresponds to reducing the quantization error at a fixed bit rate, or equivalently, maintaining the same quantization error at a lower bit rate. Speech and image coding literature present several well known techniques for reducing the redundancy present in a data source. Two techniques which were considered for use in this thesis are vector quantization and transform coding. The next section provides an overview of the issues associated with each of these techniques.

5.3 Review of Relevant Quantization Approaches

Vector quantization and transform coding are both block quantization algorithms. Block quantization refers to the fact that the data which is to be quantized is first grouped into a fixed length block. The block is then quantized, transmitted, and then reconstructed at the receiver. The advantage of a block quantization algorithm is that it provides a convenient manner for accessing the redundancies in the data. In addition the size of each block can be easily varied to meet a number of performance objectives.

5.3.1 Vector Quantization

Vector quantization represents each data block or vector by a single codeword. An essential part of this technique is a stored table of N code vectors. Each input data block is compared against all N code vectors and the one which is deemed closest is chosen to represent the data block. In practice closeness is often determined by evaluating the mean-square error between each of the stored code vectors and the input vector. The one resulting in the minimum error is chosen as the closest code vector. If the receiver and transmitter both have an identical table of stored code vectors, then only the index of the chosen code vector needs to be transmitted. If a fixed length binary code is used to send this information, then $\lceil \log_2(N) \rceil$ bits must be used. The quantized value of the data block is then equal to the closest code vector.

Since every data block is represented by one of the stored code vectors, the performance of this quantization scheme is highly dependent on the chosen set of code vectors. Several different approaches to the design of this table have been proposed, however, a simple algorithm exists for the selection of a nearly optimal set. This algorithm, which is often referred to as the k-means algorithm, has been shown to converge to a local minimum of the quantization error function [10]. For a large training set, these code vectors are assumed to be nearly optimal. Results confirm the advantages of designing the code vectors in this manner.

The primary benefit of a vector quantizer is that it achieves excellent quantization efficiency. As discussed in [10], a vector quantizer can utilize both linear and non-linear dependencies within a data block. In addition it can utilize the added dimensionality of the data, and the shape of its probability density function to gain additional efficiency. These properties allow a vector quantizer to approach the rate-distortion bound for a stationary source. The primary disadvantage of this technique is that its computation and storage requirements are extremely high. If each data block contains M elements, then MN memory elements are needed to store the table of code vectors. In addition the search over the N code vectors requires MN multiplies and $(2M-1)N$ additions, if mean-square error is used as the distance measure. In order to achieve good redundancy removal it is often necessary to use a large block size. In addition low quantization error often requires the use of a large table of code vectors. Since the size of this table, N, is exponentially related to the bit rate, the addition of a single bit will double the computation and storage requirements. For these reasons vector quantization is often limited to low bit rates and small block sizes.

5.3.2 Transform Coding

Transform coding is another well known method of reducing the redundancy from a block of data. Each input block is first transformed into a new data block. The elements of the transformed data block are then scalar quantized and sent to the receiver, where the inverse operation is used to generate the quantized version of the original data block. The principle behind this technique is that the transformed data block can be quantized more efficiently than the original data block. In general this is caused by a reduction in the amount of correlation which exists in the transformed elements, relative to the original data elements. Since the transforms under consideration are restricted to be linear, transform coding techniques generally can only take advantage of linear dependencies in the data. Therefore, although transform techniques can result in significant improvements in coding efficiency, it cannot in general achieve the same performance level as a vector quantizer.

The primary factor in the design of a transform coding algorithm is the selection of the transform. A number of different transforms have been shown to be useful in coding applications. The choice of the transform determines the algorithm's ability to decorrelate the input data block. In addition certain transforms have advantages in terms of computation and storage requirements.

One transform which is frequently used in image and speech coding is the Discrete Cosine Transform (DCT) [9]. This transform has a number of desirable properties. First it is data independent, which means that the basis vectors of the transform do not need to be calculated and stored. In addition the DCT can be implemented with a FFT algorithm, which reduces the computational requirement of an M element transform from M^2 to $M \log M$. Finally the DCT has been shown to yield good decorrelation of stationary sources.

In order for a transform coder to improve the coding efficiency, a bit allocation scheme must be used which takes advantage of the non-stationarity of the transform coefficients. The principle is to use more bits to quantize the transform coefficients with large variances, and less bits to quantize the coefficients with small variances. If the total number of bits per block is constrained to equal R_T, then the optimal bit allocation rule is given in [9] to be:

$$R_m = \frac{1}{M}R_T + \frac{1}{2}\log_2\left(\frac{\sigma_{tm}^2}{[\prod_{j=0}^{M-1}\sigma_{tj}^2]^{\frac{1}{M}}}\right) \tag{5.2}$$

where R_m is the optimum number of bits to allocate to the m'th transform coefficient, and σ_{tm}^2 is the variance of the m'th transform coefficient. In practice this rule must be modified to account for the non-negative integer constraint on R_m. The important feature of this bit allocation rule.is that the number of bits is determined by the logarithm of the element variances. This rule agrees with the intuitive notion that a coefficient with twice the amplitude, and hence four times the variance, would receive one more bit.

5.4 4.8 Kbps System Development

Using the information which is presented above, a basic approach was devised for a high quality 4.8 kbps MBE speech coder. As mentioned previously the parameters for each frame consist of a fundamental frequency, a set of voiced/unvoiced decisions, and a set of harmonic magnitudes and phases. The quantization of each frame is done in a manner similar to the previously designed 8 kbps system. However, a series of different algorithms are used which are designed around the characteristics of each parameter. By incorporating some of the techniques which were discussed in the previous section, the new system is able to achieve performance comparable to that of the earlier 8 kbps system.

As in the previous system, the number of parameters which occur for any frame is a variable. Therefore the parameters cannot be quantized and transmitted in an arbitrary order. Specifically, the fundamental frequency must be transmitted first, since it determines the number of harmonics. In addition the voiced/unvoiced information must be received before the harmonic phases, since there is no phase information for unvoiced harmonics. A consequence of the variability in the number of parameters per frame is the need for a bit allocation algorithm, which assigns the available number of bits over the number of parameters in the current frame. Although this increases the complexity of the quantization process, it increases the flexibility and performance of the system.

5.4.1 Fundamental Frequency Encoding

The primary characteristics of the fundamental frequency are determined by the estimation algorithm described in Chapter 3. Since an analysis-by-synthesis approach is used to estimate this parameter, it's value is only known to some fixed resolution. The estimation algorithm therefore acts as a quantizer, fixing the value of the fundamental to one of 512 levels between 70 Hz. and 400 Hz. Since additional quantization is not necessary or desirable, fixed length encoding of the fundamental frequency would require 9 bits. However, due to the pitch tracking portion of the estimation algorithm, and the nature of speech, the fundamental frequency usually only makes small variations from frame to frame. An ideal model for this form of behavior would be a discrete Markov source, where the state corresponds to the previous value of the fundamental frequency. An optimal coding scheme could be based on this model and used to lower the average number of bits required to encode this parameter. Unfortunately the large number of states in this model would add far too much complexity for the few bits which could be saved. A simple alternative to this approach is to use a sub-optimal coding scheme which still captures the basic memory in the process. The one which is used codes the difference between successive fundamental frequencies with 6 bits if that difference is small. Otherwise the current fundamental frequency is encoded using 10 bits. This scheme has a long term average of about 7 bits per frame, which is a savings of 2 bits per frame over fixed length coding.

5.4.2 Voiced/Unvoiced Encoding

The voiced/unvoiced information consists of a series of binary decisions which indicate the nature of different regions of the spectrum. In the 4.8 kbps system each region corresponds to the portion of the spectrum covered by three consecutive harmonics. Therefore the first decision corresponds to the first, second and third harmonic, the second to the fourth, fifth, and sixth harmonic, etc... . There are a maximum of 12 decisions, and any harmonics past the 36'th are set to be unvoiced by default. This is different than the scheme which was used in the 8 kbps system, where the number of decisions was always equal to 12 and the size of each region was varied to cover the entire 4 kHz. bandwidth. These two schemes produce nearly equivalent sounding speech. The motivation behind fixing the size of each region to 3 harmonics is that it simplified the phase quantization algorithm.

Since the voiced/unvoiced decisions only consist of binary information, a quantization algorithm is not needed. Instead a simple and efficient method of representing this information needs to be employed. In the previous 8 kbps system the voiced/unvoiced decisions were encoded using a single bit per decision. However, this technique does not take advantage of the redundancy in the voiced/unvoiced information. One approach which can utilize this redundancy is to block the decisions into groups of four. A Huffman code is then used to represent the sixteen possibilities with the minimum expected codeword length. The Huffman code tends to assigns fewer bits when the block is either all voiced or all unvoiced. This reflects the fact that the voicing information is often clustered into long strings of similar decisions. The actual code is shown in Table 5.3, which uses a 1 to refer to a voiced decision and a 0 to refer to an unvoiced decision.

5.4.3 Quantization of the Harmonic Phases

In contrast to both the fundamental frequency and the voiced/unvoiced information, the harmonic phases are not quantized by the analysis algorithm. In the 8 kbps

V/UV String	Prob.	Bits	Code Word
0000	0.114	4	1000
0001	0.066	4	1001
0010	0.029	5	11000
0011	0.065	4	1010
0100	0.029	5	11001
0101	0.035	5	11010
0110	0.016	7	1111110
0111	0.086	4	1011
1000	0.02	6	111100
1001	0.021	6	111101
1010	0.01	7	1111111
1011	0.054	5	11011
1100	0.019	6	111110
1101	0.035	5	11100
1110	0.026	5	11101
1111	0.375	1	0

Table 5.3: Huffman Code for Voiced/Unvoiced Decisions

system the quantization was done by passing the phase residual through a non-uniform Max-Lloyd quantizer [2]. The advantage of quantizing the phase residual is that it possesses less entropy than the actual phase and therefore it can be coded more efficiently. An alternative viewpoint is that the phase prediction algorithm removes the redundancy between the previous harmonic phase and the current one. The phase residual is then the new information which is not contained in the previous phase. The use of a Max-Lloyd quantizer then provides the minimum quantization error for a scalar quantizer.

The problem with this approach is that although the phase estimation algorithm reduces the interframe dependencies, it does not address the dependencies which exist between adjacent harmonics within the same frame. One approach to solving this problem is to replace the scalar quantizer with a vector quantizer. If the phase residuals are grouped into blocks and then passed through a vector quantizer, both the interframe and intraframe dependencies are exploited. As discussed in the previous section, vector quantization provides nearly optimal performance in terms of quantization efficiency. However, the computation and storage costs can easily become prohibitive.

In order to examine the advantages and disadvantages of this approach a vector quantizer with a block size of 3 and $N = 64$ was designed for the harmonic phases. This block size was chosen because it corresponded to the size of each voiced/unvoiced region. For each region which was declared voiced, three consecutive harmonic phases could be blocked together and passed to the vector quantizer. It was presumed that the use of consecutive harmonic phases would result in the greatest coding efficiency from the vector quantizer. Since the size of each voiced/unvoiced decision region was fixed at 3 harmonics, the implementation of the vector quantizer was simplified, considerably. If this size was not fixed, then either several different vector quantizers would have to be employed, or non-consecutive phases would have to grouped together.

Quantization of the harmonic phases using this vector quantizer yielded very good results. When the harmonic phases were quantized in this manner, the resultant speech sounded the same as when the 13 level Max-Lloyd quantizers were used. The advantage of the vector quantizer is that it is able to represent 3 phases with 6 bits, for an average of two bits per phase. This is a 1.7 bits per phase improvement over the scalar quantizer. The computational and storage requirements of this new approach are still reasonable due to the small block size and the small number of code vectors. In order to quantize each phase block the algorithm must do 192 multiplies and 320 additions. In addition 192 memory elements are required for the storage of the code vectors. Although these figures may seem significant, they are negligible compared to the requirements of the analysis and synthesis algorithms.

These results led to the inclusion of this vector quantizer in the 4.8 kbps speech coding system. As in the 8 kbps system, only the voiced harmonics between the first and the twelfth are quantized. The phase residual, defined by equation (5.1), is calculated for each of these harmonics. These residuals are then grouped into blocks of three, and vector quantized to one of 64 levels. The decoder reconstructs each block of phase residuals, and combines them with the predicted phases to form the actual phase information. Phase information for voiced harmonics beyond the twelfth is not transmitted to the decoder. Therefore these phases are reconstructed as random values between $-\pi$ and π. Since phase information is not required for unvoiced harmonics, this information is not quantized or reconstructed. The total number of bits required to encode the harmonic phases is dependent on the voiced/unvoiced information. The total number varies between 0, if all of the first twelve harmonics are unvoiced, and 24, if all of the first twelve harmonics are voiced.

5.4.4 Quantization of the Harmonic Magnitudes

The last parameters which are quantized in each frame are the harmonic magnitudes. These parameters are quantized using all the bits which remain after the fundamental frequency, the voiced/unvoiced decisions and the harmonic phases are encoded. For the 4.8 kbps speech coder, the number of bits which are available varies between 50 and 89. At the same time the number of harmonic magnitudes is varying between 9 and 50, depending on the value of the fundamental frequency. As mentioned previously a bit allocation algorithm is used to match the number of available bits to the number of harmonic magnitudes. Once all of the bits have been allocated, the harmonic magnitudes are quantized. The quantization algorithm must be capable of achieving high efficiency, and it must be able to operate with a variable number of parameters and available bits.

Since the basis behind high quantization efficiency is the removal of redundancy from the data, several simple experiments were conducted which looked for correlation among the harmonic magnitudes. One major finding was that there is a significant amount of correlation which exists between adjacent parameters. Although this correlation is high for adjacent harmonics within the same frame, it is actually higher for magnitudes which occupy the same frequency region in neighboring frames. Another important finding was that there are also significant higher order correlations between harmonics. These findings indicate that an algorithm should be able to quantize the harmonics in a substantially more efficient manner than was done in the 8 kbps system. Since this system quantized the log difference between harmonics at adjacent frequencies, it was only utilizing the first-order intra-frame correlation. An algorithm which also incorporates the first-order inter-frame correlation and higher-order correlations could yield substantially better performance. In addition there may also be non-linear dependencies, which could also be exploited.

The algorithm which was developed to remove these additional redundancies is based on a new time-frequency framework which is shown in Figure 5.2. Every 20 ms., the harmonic magnitudes for a new speech frame are estimated. Since these parameters correspond to spectral information in the speech, a two dimensional representation can be constructed, with time on one axis and frequency on the other.

The frequency index, m, corresponds to the harmonic number of the magnitude, $|A_n(m)|$, while the time index, n, corresponds to the frame number. This representation is very similar to the spectrogram representation of a one-dimensional signal. A convenient way of accessing the redundancies in the data is to divide these parameters into time-frequency blocks. Each block can then be quantized using either a transform coding algorithm or a vector quantizer. By varying the size of the blocks and the quantization method, this approach can be made to accommodate a variety of performance requirements. In particular these variables determine the algorithm's quantization efficiency, its computation and storage requirements, and its coding delay.

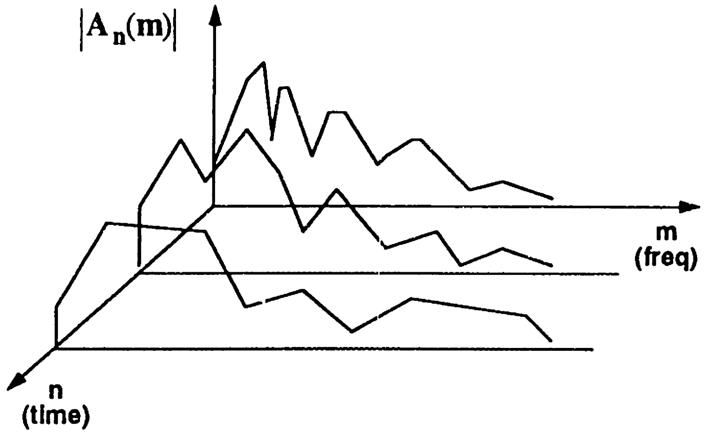

Figure 5.2: Time - Frequency Representation of the Harmonic Magnitudes

For the 4.8-kbps system, a transform coding algorithm, rather than a vector quantizer, was chosen for the quantization of each time-frequency block. A vector quantizer was ruled out because of its computation and storage requirements. The variability in the number of available bits and the number of magnitudes can result in a situation where there are nearly 10 bits per harmonic. In this situation a vector quantizer would become almost unmanageable, even if there is only a single magnitude per block. For larger block sizes the memory and storage requirements would exceed the capabilities of the computer. Another problem with a vector quantizer is that it is usually designed to operate with a constant number of bits per block. Since the number of available bits is changing from frame to frame, a standard vector quantizer would not work. One option would be to design a set of quantizers, each operating with a different number of bits per block. This solution, however, would complicate the storage problem, since the number of stored code vectors would increase dramatically. A transform coding algorithm avoids these problems because it relies on scalar quantization. If uniform quantizers are used, then the variability in the number of bits does not change the computation or storage requirement of the algorithm. Instead, these requirements are dictated by the transform.

The use of the DCT as the quantization transform results in good decorrelation of the harmonic magnitudes, while maintaining reasonable computation and storage requirements. As mentioned in a previous section, the DCT offers good performance for stationary sequences. A block which has large time and frequency dimensions could be used to provide good coding efficiency. Unfortunately, several factors limit the allowable block size. The first centers around the issue of coding delay. The size of a block in the time direction corresponds to the number of frames which must be analyzed before the block can be transmitted. Since a frame is analyzed every 20 ms., an increase in the time dimension of a block corresponds to an additional 20 ms. of delay. A delay less than 200 ms. is tolerable in typical real time applications, however larger delays limit the applicability of a system. Since the analysis and synthesis algorithms have a combined delay of 100 ms., the coding system must not significantly increase this delay. Another constraint on the block size is caused by the non-stationarity of the harmonic magnitudes along the frequency axis. The variance of low frequency harmonic magnitudes is generally much higher than for the high-frequency magnitudes. The DCT can actually cause a reduction in coding efficiency in this case. One solution to this problem is to limit the size of the blocks in the frequency direction, such that all the elements have nearly equal variance. Ideally the frequency dimension of the block should vary with the fundamental frequency, since the variance is a function of absolute frequency rather than harmonic number.

A block size which has a frequency dimension of 8 and a time dimension of 1 was chosen in light of the aforementioned problems. For small fundamental frequencies, the variance over 8 harmonic magnitudes is fairly constant. For large fundamental frequencies the variances may be considerably different, since there may be only 10 or 12 harmonics with the 4 kHz. bandwidth of the system. Fortunately the small total number of harmonic magnitudes decreases the need for very efficient quantization. Any loss in efficiency from the DCT is negated by the large number of bits per harmonic. This choice of block size also limits the coding delay to the 100 ms. imposed by the analysis and synthesis algorithms. Rather than increasing the time dimension of the block, and incurring additional delay, a hybrid approach was adopted.

The idea behind a hybrid coding system is to use different coding methods along different directions in the data. A common approach to the coding of a video sequence is to calculate the transform coefficients for each frame in the sequence. Each transform coefficient is then differentially quantized along the temporal direction [9]. This technique uses the transform to reduce the spatial redundancy in the sequence, and it uses a differential, or predictive, approach to reduce the temporal redundancy. One advantage of predictive coding is that it is based only on previously transmitted data, and therefore it does not add any delay to the system. In addition predictive coders can actually out-perform transform coders when the block size is very small. This fact arises because of the presence of blocking boundaries in a transform coder [9].

Hybrid coding can be applied to the quantization of the harmonic magnitudes in a very similar manner. First the differences between the log magnitudes of the current frame and the log magnitudes of the previous frame are found. These temporal differences are then grouped into blocks of 8 and transformed with the DCT. The coefficients are then quantized and transmitted. A block diagram of this coding algorithm is shown in Figure 5.3. Since all of the operations are invertible, the decoding algorithm can perform the inverse procedure to calculate the magnitudes of the current frame.

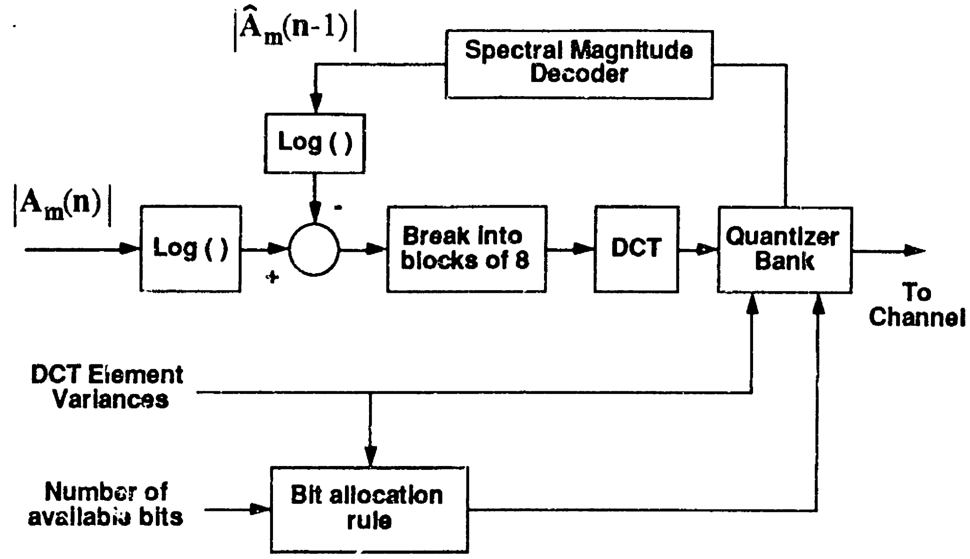

Figure 5.3: Block Diagram of Harmonic Magnitude Encoder

The logarithm function is used in the quantization algorithm for several reasons. First it provides a convenient method for insuring that the quantized magnitudes are always positive. A DCT with scalar quantization cannot guarantee that the decoded magnitudes remain positive. However, the exponential fu ction always produces a positive value, and therefore the quantized magnitudes are guaranteed to be positive. Another desirable property of the logarithm and exponential pair is that it causes the signal to quantization noise ratio to remain constant, regardless of the absolute level of the signal. This is beneficial because of the ear's masking ability. Essentially, the perceived effect of narrowband noise is dependent on the signal-to-noise ratio in that band, rather than on the absolute noise level. The use of the logarithm function forces quantization errors to occur in such a manner that their perceived effects are lessened. A final reason for including this function is that the difference between log magnitudes is not effected by a scale change in those magnitudes. This means that the quantization algorithm is not effected by the average signal level of the speech. Very quiet speech and very loud speech can be quantized, without the need for any variation in the quantizer characteristics.

The decorrelation which is performed by the DCT results in some transform coefficients having much higher variance than others. In order to take advantage of this fact a bit allocation algorithm is required. Equation (5.2) provides an optimal bit allocation rule for the quantization of any set of parameters having the same probability distribution. This rule is dependent on the variance of each member in the set. It is applied to the quantization of the harmonic magnitudes in the following manner. First, the number of available bits is divided among all of the blocks which occur in that frame. This is accomplished by integrating a bit density curve similar to that shown in Figure 5.1. Next, the number of bits which have been allocated to each block are divided among the transform coefficients according to equation (5.2). A "water-filling" algorithm is used to satisfy the positive-integer constraint as discussed in [9]. After all of the bits have been allocated to the transform coefficients, each is scalar quantized with the appropriate number of bits. The quantizer for each coefficient is tuned to the characteristics of that coefficient.

In order to do the bit allocation and the quantization it is necessary to have an estimate of the mean and variance of each transform coefficient. These estimates are obtained by calculating the sample-mean and sample-variance over a large ensemble of speech. From the variance information, the bit allocation rule follows directly. In addition the step size and offset of each quantizer can be found if the probability distribution of the transform coefficients is known. Tests have shown that the system is fairly insensitive to the assumed distribution, and so, for simplicity, the transform coefficients are assumed to be Gaussian. The optimal step size as a function of variance for this distribution is given in [9]. The quantizers for each coefficient use this step size and an offset equal to the estimated mean.

Chapter 6

Performance Evaluation

The 4.8 kbps speech coding system described in this thesis was evaluated using several different procedures. Initial comparisons between this system and the previously developed 8 kbps system, showed the two to have nearly equal performance. More extensive listening tests were then done to form a general opinion of the system. A collection of speech material was processed by the system. This material consisted of a variety of different speakers and noise conditions. The general impression was that the system maintained a high quality level across the test ensemble. The system showed impressive robustness for a number of traditional problem cases such as multiple speakers, and speech in the presence of harmonic noise. The primary artifact of the coded speech is a reverberant sound which is due to a lack of enough coded phase information. Experiments showed that although this reverberance could be reduced through small variations in bit allocation, the block sizes and other system details, it was still noticeable. Fortunately, this effect is much more masked in a normal listening environment than it is under close listening with headphones.

In order to get more quantitative results on the performance of the system, a series of Diagnostic Rhyme Tests (DRT) were conducted. The DRT is a standardized test used to measure the intelligibility of a speech processing system [15]. The test consists of a sequence of words, each of which is one of two rhyming choices. The word pairs differ only in the first consonant, thereby eliminating the effects of context information. Three different tests were performed on the 4.8 kbps speech coding system. Each test consisted of two DRT tapes, the first being unprocessed, and the second being the output of the speech coding system. The use of two tapes per test allows an estimate to be made of the loss of intelligibility caused by the coding system. The first test was performed on clean speech; the second test added 26 dB. white gaussian noise; and the third test added 20 dB. of simulated aircraft noise. For the latter two tests the noise level was chosen such that projected intelligibility score for the unprocessed speech would be close to 80 percent. This level was chosen as representative of a typical noisy environment. The results for the three tests are given in Tables 6.1, 6.2 and 6.3, respectively. The mean and standard deviation for each DRT tape is presented as an estimate of intelligibility and the accuracy of this estimate.

System	Mean Score	Std. Dev.
Original Clean Speech	97.53	0.31
4.8 kbps Clean Speech	95.01	0.42

Table 6.1: DRT Results for Clean Speech

The results of the DRT show that the system has very high intelligibility for clean speech. The unprocessed clean speech has a mean intelligibility of 97.5% while the coded speech has a mean intelligibility of 95.0%. These high scores indicate that even in the absence of context information the speech is almost completely intelligible. For the noisy speech the intelligibility scores were much lower. The unprocessed speech with additive gaussian noise received a mean score of 83.2%

System	Mean Score	Std. Dev.
Original Noisy Speech	83.2	0.99
4.8 kbps Noisy Speech	71.7	0.96

Table 6.2: DRT Results with 26 dB. Gaussian Noise

System	Mean Score	Std. Dev.
Original Noisy Speech	90.62	0.66
4.8 kbps Noisy Speech	75.3	1.18

Table 6.3: DRT Results with 20 dB. Simulated Aircraft Noise

and the unprocessed speech with simulated aircraft noise received a mean score of 90.6%. The mean scores for the coded noisy speech were 71.7% and 75.3% for the gaussian noise and the simulated aircraft noise, respectively. Several conclusions can be drawn from the noisy speech test data. First the presence of the noise has substantially reduced the intelligibility of the unprocessed speech. The noise masks some important cues which the ear uses to decipher the speech. Another finding is that the coding system yields a much greater loss in intelligibility for noisy speech than it did for clean speech. One of the principal attributes of the MBE speech model, as reported in [2], is that it produces less degradation for noisy speech than conventional speech models. However, these DRT results seem to indicate that there still is a substantial degradation. These findings can be explained in several ways. First DRT data was not obtained for a conventional coder at 4.8 kbps, therefore it is not known how this system would perform in comparison to the MBE coder. A conventional coder probably would have yielded even greater degradation in the presence of noise. Also it is possible that the loss in intelligibility for the MBE coder is not due to the underlying model, but is instead related to the parameter quantization method. Since clean and noisy speech have different characteristics, the 4.8 kbps coding system may be adjusted to perform better for clean speech at the expense of the noisy speech performance. Retuning the quantization algorithms may yield better overall performance.

Chapter 7

Conclusion

A 4.8 kbps speech coding system has been presented which offers high quality speech capability. The system can be implemented as a cascade of several algorithms. The first element of the system estimates the MBE model parameters. These parameters are then quantized and transmitted across some channel. A decoder then reconstructs the quantized parameter values, from which the synthesis algorithm produces the synthesized speech. The primary focus of this thesis has been the quantization of the model parameters. Initial experiments showed that there is a substantial amount of redundancy which exists among these parameters. Quantization techniques have been developed which utilize this redundancy in order to achieve higher quantization efficiency. A coding system based on these techniques has been shown to produce high quality speech at 4.8 kbps, while attempts to achieve equal quality with less efficient techniques have not been successful.

Several features of this system make it particularly attractive for use in a number of speech coding applications. Because the system is based on the MBE speech model, it is extremely robust to the presence of background noise. In addition the use of the MBE speech model reduces the sensitivity of the system to errors in the pitch estimation algorithm. The quantization methods which were chosen for the system also enhance its applicability. These techniques result in high quality speech without inducing substantial computational penalties. The storage requirements and coding delay for the system are also within acceptable bounds.

7.1 Suggestions For Improved Quantization Efficiency

Although this speech coding system has a number of desirable features, it can still be improved. Efforts to improve the quality of the system or lower the bit rate may profit from the use of different quantization algorithms which achieve better quantization efficiency. The time-frequency framework which has been presented for the quantization of the harmonic magnitudes can accommodate a number of quantization algorithms. Replacement of the DCT with either a Karhunen Loéve Transform or a vector quantizer may result in more efficient quantization. Since the formant structure in the harmonic magnitudes can result in both linear and non-linear dependencies, vector quantization holds the most promise for performance improvements. Unfortunately, the incorporation of these ideas are likely to cause a substantial increase in the computational and storage requirements of the system. One interesting possibility is to use a tree structured vector quantizer [10]. This technique preserves many of the advantages of vector quantization without incurring the associated computational penalties. A structured vector quantizer also has the advantage that the quantization of each block can be halted at any stage of the tree. This property allows the quantizer to easily accommodate variations in the number of available bits per block.

Quantization efficiency may also be improved by allocating bits over multiple speech segments. Rather than allocating 96 bits to each frame, it may be desirable to allocate more bits to some frames and less to others, while keeping the total number of bits constant. The non-stationarity in the speech model parameters results in some frames being more difficult to quantize than others. This problem can be lessened by adapting the bit allocation to correct for these non-stationarities. One disadvantage of this approach is that the coding delay will increase.

Various other concepts may be used to improve the quantization efficiency of the system. One possibility is to use a higher order predictor for the hybrid coding of the harmonic magnitudes. This may allow the system to remove more redundancy in the parameters. System performance may also gain from the use of non-uniform quantization for the DCT coefficients, and variations in the magnitude and phase block sizes.

7.2 Suggestions for Improved Speech Modeling

One problem with this and other model based speech coders is that even with perfect quantization, there is some degradation in speech quality. Errors in the modeling process and in the estimation of the model parameters result in artifacts in the speech. In very high bit-rate speech coding or in applications such as time-scale modification of speech, these artifacts can be the limiting factors in system performance. One of the principal advantages of the MBE speech model is that it results in substantially fewer model induced degradations. Unfortunately, the modeling and estimation process is still noticeable for most speech material. In particular the presence of pitch errors and voiced/unvoiced errors results in small artifacts in the speech.

One extension to the MBE speech model which may result in an increase in speech quality is to replace the voiced/unvoiced decision with a dual voiced and unvoiced representation. In the current MBE speech model a set of harmonic magnitudes and phases are estimated under the assumptions that the speech is voiced. The validity of this assumption is then determined for each harmonic by comparing the actual spectrum with the estimated voiced spectrum. If the error between these spectra is small then the region is declared voiced. However, if the error is large, then the region is declared unvoiced, and the voiced magnitude and phase are discarded in favor of an unvoiced magnitude estimate. A better speech model may result if each harmonic is allowed to be a combination of voiced and unvoiced energy. This could easily be done by retaining the voiced magnitude and phase estimate for each harmonic. The unvoiced portion of the spectrum could then be obtained from the error between the original spectrum and the estimated voiced spectrum. In practice the unvoiced spectrum could be parameterized by a small number of values which correspond to the average magnitude over some frequency region. Because this idea incorporates both a voiced and unvoiced spectrum, there is no longer a need for voiced/unvoiced decisions. This information would now reside in the relative amplitude of the voiced and unvoiced spectra.

This dual excitation speech model has several advantages in terms of both the quality and the intelligibility of the synthesized speech. First it removes the problem of determining a voiced/unvoiced threshold. This should improve the robustness of the model to widely varying noise and speech conditions. In addition this extension should improve the accuracy of the speech model in low SNR conditions. The reason for this is that at most frequencies noisy speech contains both voiced and unvoiced energy. For the harmonics which are declared voiced, the current model eliminates the unvoiced energy, resulting in an effect similar to that produced by comb-filtering. In contrast, the harmonics which are declared unvoiced are principally filled with background noise, and the voiced speech cues are eliminated. There are a number of other ways in which this extension may improve the speech modeling process. However, the applicability of this dual excitation speech model has not yet been shown.

Bibliography

[1] B. Gold and J. Tierney, "Vocoder Analysis Based on Properties of the Human Auditory System," *M.I.T. Lincoln Laboratory Technical Report*, TR-670, December 1983.

[2] Daniel W. Griffin, "Multi-Band Excitation Vocoder," *Ph.D. Thesis*, E.E.C.S. Department, M.I.T., 1987.

[3] Daniel W. Griffin and Jae S. Lim, "A High Quality 9.6 kbps Speech Coding System," *Proc. of IEEE Int. Conf. on Acoustics, Speech and Signal Proc.*, pp. 125-128, Tokyo, Japan, April 13-20, 1986.

[4] Daniel W. Griffin and Jae S. Lim, "A High Quality 8 Kbps Multi-Band Excitation Vocoder," *Int. Symposium on Signal Proc. and it's Applications*, Brisbane, Australia, Aug. 24-28, 1987.

[5] Daniel W. Griffin and Jae S. Lim, "A New Model-Based Speech Analysis/Synthesis System," *Proc. of IEEE Int. Conf. on Acoustics, Speech and Signal Proc.*, pp. 513-516, Tampa, Florida, March 26-29, 1985.

[6] Daniel W. Griffin and Jae S. Lim "A New Pitch Detection Algorithm," *International Conference on Digital Signal Processing*, Florence, Italy, Sept. 5-8, 1984.

[7] Daniel W. Griffin and Jae S. Lim, "Signal Estimation From Modified Short-Time Fourier Transform," *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. ASSP-32, no. 2, pp. 236-243, April 1984.

[8] J. N. Holmes, "The JSRU Channel Vocoder," *IEEE Proc.*, Vol. 127, Pt. F, No. 1, Feb. 1980, pp. 53-60.

[9] N. S. Jayant and Peter Noll, *Digital Coding of Waveforms*, Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1984.

[10] J. Makhoul, S. Roucos, and H. Gish, "Vector Quantization in Speech Coding," *Proc. of the IEEE*, Vol. 73, No. 11, Nov. 1985, pp. 1551-1588.

[11] R. J. McAulay and T. F. Quatieri, "Mid-Rate Coding Based on a Sinusoidal Representation of Speech," *Proc. of IEEE Int. Conf. on Acoustics, Speech and Signal Proc.*, pp. 945-948, Tampa, Florida, March 26-29, 1985.

[12] C. S. Myers and L. R. Rabiner, "Connected Digit Recognition Using a Level-Building DTW Algorithm," *IEEE Trans. on Acoustics, Speech and Signal Processing*, vol. ASSP-29, no. 3, pp. 351-363, June 1981.

[13] L. R. Rabiner and R. W. Schafer, *Digital Processing of Speech Signals*, Prentice-Hall Inc., Englewood Cliffs, New Jersey, 1978.

[14] M. R. Schroeder and B. S. Atal, "Code-Excited Linear Prediction (CELP): High-Quality Speech at Very Low Bit Rates," *Proc. of IEEE Int. Conf. on Acoustics, Speech and Signal Proc.*, pp. 937-940, Tampa, Florida, March 26-29, 1985.

[15] W. D. Voiers, "Evaluating Processed Speech Using the Diagnostic Rhyme Test," *Speech Technology*, Jan./Feb. 1983.