DESIGN OF A HIGH QUALITY 2400 BIT PER SECOND ENHANCED MULTIBAND EXCITATION VOCODER

By

WALTER D. ANDREWS

Bachelor of Science in Electrical Engineering

Stillwater, Oklahoma

1994

Submitted to the Faculty of the
Graduate College of the
Oklahoma State University
in partial fulfillment of
the requirements for
the Degree of
MASTER OF SCIENCE
December, 1994

DESIGN OF A HIGH QUALITY 2400 BIT PER

SECOND ENHANCED MULTIBAND

EXCITATION VOCODER

Thesis Approved:

Thesis Advisor

George Scheets

R. Ramakumar

Thomas C. Collins

Dean of the Graduate College

PREFACE

This study describes the design of a high quality 2,400 bit per second enhanced MultiBand Excitation vocoder.

A study such as this is not possible without help. First, I would like to thank the Department of Defense for funding this project. A big thanks is given to Tom Tremain and Vanoy Welch for their support and advice.

I would also like to thank the members of my advisory committee, Dr. Keith Teague, Dr. George Scheets, and Dr. Ramakumar. A special thank you is extended to Dr. Keith Teague for his guidance and friendship during this project and previous semesters. Also, I am very grateful to the School of Electrical and Computer Engineering for the employment which made this study possible.

A very special thanks goes to my wife, Ronda, and my children, Chuck and Caleb for their patience and understanding throughout this long year. I need to also thank my Mother, Kay Parker, who helped at the most desperate times and my Grandparents Woodrow Lorene Lawerence for their support and encouragement. A thanks to my Father, Walt Andrews, for his support.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I

INTRODUCTION

Purpose and Motivation

The purpose of this paper is the design of an Enhanced MultiBand Excitation (MBE) [1] voice coder (vocoder) to operate at low bit rates, while maintaining high quality. For this study, the low bit rate is 2,400 bits per second (bps) and high quality is defined to be the equivalence of telephone (toll) quality. The general idea is to characterize, mathematically, a digitized speech signal that has been bandlimited to 4 kHz and sampled at 8 kHz. This mathematical model will be used to generate a set of parameters in the analyzer (transmitter). These parameters will be encoded in the analyzer, using a quantization scheme, and then decoded in the synthesizer (receiver). The decoded parameters will be used to generate a synthetic speech signal. The goal is to synthesize high quality speech, while reducing the bit rate.

The speech model used in this study is MultiBand Excitation, with enhancements. In brief, the MBE model separates the spectrum of a frame of speech into bands. Each of these bands is determined to be either voiced or unvoiced as opposed to the traditional method of one voicing decision per frame. The enhancements will be made in the estimation of the parameters such as: pitch, voiced/unvoiced decisions and spectral representation. The MBE model will be discussed in greater detail in Chapter II. A number of speech models were considered but MBE has shown the greatest potential in being able to reproduce toll quality speech at bit rates of 4,800 bits per second and below [2].

The motivation for this study on speech coding can be contributed to the demand for voice communication, the new generation of technology for cost-effective implementation of digital signal processing algorithms, the need to conserve bandwidth, and the need to conserve disk space in speech storage [2]. Low bit rate speech coding, although not a new topic, has become an important area for research in recent years. There are several reasons for this occurrence. These include advances in microprocessor technology, the sharp decrease in cost of computation and memory, the increased emphasis on providing high-quality communication services, and the improvement in speech models [3],[4]. The result of the recent surge in research has been the development of a number of good speech coding systems at bit rates of 4,800 bps and below. Many of these systems will be discussed in Chapter II.

Speech coding has application in several areas. Three of these areas are the wired telephone network, the wireless network, and voice security. The reader is reminded that each of these areas has numerous specific applications, too many to mention in this introduction. Only three areas are listed because this study is mostly concerned with telephone bandlimited speech. Speech coding does have many applications in voice processing. For a more in depth discussion of speech coding applications the reader is referred to the papers by Gersho [2] and Rabiner [3].

The sections following will outline the procedure used to accomplish the goal stated and list some of the limitations associated with this study. Chapter II will provide some history of speech coding and discuss the different methods and models. The actual design and implementation of the Enhanced MultiBand Excitation (EMBE) vocoder will be detailed in Chapter III. Results and conclusions will be discussed in Chapter IV. Any derivations needed will be supplied in the Appendix.

Overview of Procedure

The design of a vocoder involves choosing a specific speech coding algorithm, a speech model, and estimating the specific parameters for a given speech model. Other parameters such as sampling frequency, frame rate, window size, and number of bits used for quantization will be discussed in more detail in Chapter III. The selection of the speech coding algorithm, speech model, and parameter estimation are discussed in the following sections.

Selection of Speech Coding Algorithm

Speech coding algorithms can be divided into two general categories, vocoders and waveform coders. Vocoders, sometimes referred to as source coders and parametric coders [2], use a mathematical model to characterize the original speech waveform. That is vocoders include a speech production model. On the other hand waveform coders are judged by their ability to preserve the original speech waveform. For this study, the vocoder algorithm was chosen over the waveform coder because waveform coders do not perform well at bit rates below 4,000 bps [2]. Refer to Chapter II for a more detailed discussion of these algorithms.

Selection of Speech Model

A number of different speech models exist for vocoders. Some of the "popular models" are linear-predictive coding (LPC) based models, wavelet coding, sinusoidal transform coding, time frequency interpolation, and MultiBand Excitation coding. For this study, the MBE speech model was chosen over the other models. The main reason is that several studies have shown that more efficient quantization can allow the MBE coder to operate, with little drop in quality, at rates of 2,400 bps and below [2],[25]. Refer to Chapter II for a more detailed discussion of these speech models.

Parameter Estimation

After choosing a speech coding algorithm and a speech model, the parameters to be encoded and decoded must be estimated. For an MBE vocoder, the estimated parameters are pitch (fundamental frequency), voiced/unvoiced (V/UV) decisions, spectrum representation, and spectrum gain. The methods for obtaining these parameters and encoding them are considered to be the enhancements of this study. The following is an overview of the parameter estimation.

The pitch is estimated using the AMPEX pitch detector algorithm developed by Van Immerseel and Martens [5]. Although another pitch detector might provide similar results, the AMPEX algorithm was available in C source code and has been reported to have very good performance over a wide range of input conditions [6]. The V/UV decisions are made using a squared error measure between the original signal and a synthetic version on a harmonic-by-harmonic basis. A varying threshold is used to determine if a harmonic is either voiced or unvoiced. The spectrum is represented using ten linear prediction coefficients (LPC). The linear prediction coefficients provide a smooth representation of the frequency spectrum. A gain value is calculated from the spectrum of the original signal and the spectrum of the LPC coefficients. The estimated parameters are then encoded. The pitch, which falls between 65 Hz and 380 Hz in this vocoder, is encoded with 8-bits. The voiced/unvoiced decisions are encoded using 10-bits. The gain is coded with 5-bits using logarithmic companding. Vector quantization, using 25-bits, is used to code the LPC spectrum. The reader can obtain a more detailed discussion of each of these parameters in Chapter III. A short discussion of the limitations of this design follows.

Limitations

As with any project, there exist some limitations. The limitations for this study are that time did not allow for a real time implementation or for bit error control. A real time implementation is a goal. As a result, bit error control will be needed.

The MBE speech model also has limitations. The speech models and algorithms either used or described are being applied to speech in the narrowband region, 0 - 4,000 Hz.

Vocoders to date still retain remnants of an unnatural or artificial sound with a buzzy characteristic. Recent research has improved the quality of synthetic speech, significantly at low bit rates. The following section provides background material and a discussion of different types of waveform coders and vocoders.

CHAPTER II

REVIEW OF LITERATURE

Overview

This chapter will provide information on the history of voice coding and the different types and methods of voice coders used. Generally, voice coders are separated into two categories: waveform coders and speech coders (vocoders) or parametric coders. For this study three speech coding algorithm categories are used. The speech coding algorithms are separated into waveform coders, *classical* vocoders, and *modern* vocoders. This distinction is made because some of the material used in this study is relatively new and much more advanced than that usually associated with classical vocoders.

The following history section is intended to be only an overview of voice coding. The more interested readers are referred to the book by Linggard [7].

Background

History

The first attempts at synthesizing speech occurred in the 1700's [8]. These early devices were mechanical and had severe limitations. Some of the earliest devices, around 1791, could only produce imitations of the vowel sounds a, e, i, o, and u while others could possibly speak either words or phrases [8]. The next mechanical speech synthesizer of significance was built in 1937. This device used keys to vary the shape of a mechanical vocal tract and could produce connected speech [8].

In 1938 the first electric speech synthesizer, called the Voder, was built [8],[9]. The development of electric speech synthesizers led to the demise of the mechanical devices. In the hands of a skilled person the Voder could produce intelligible speech.

Desires to reduce the transmission bandwidth led to the invention of the Vocoder by Homer Dudley in 1939. The Vocoder used 10 bandpass filters covering the speech frequency range to continuously update the parameters being sent, such as pitch, voiced/unvoiced decisions, and amplitudes [8],[9]. Shortly after the Vocoder's introduction in 1939, the first real-time secure voice coder was introduced. This device, referred to as SIGSALY, was used by Roosevelt and Churchill during the "D-Day" invasion [9].

All the vocoders discussed up to this point have been analog. The developments of digital computing in the 1960's allowed research in speech coding to take a new direction [8]. The next several decades produced numerous approaches to voice coding. One approach was Linear Predictive Coding (LPC). In the 1970's, the U.S. Government's

Senior Speech Scientist Tom Tremain collaborated with Bell Labs to develop LPC vocoders [9]. A number of other approaches were made in the time and frequency domains, which have ultimately led to the ability to produce high quality low bit rate speech.

The following discussions are meant to be an overview, not a detailed description of each type of coder. The reader is encouraged to review the referenced texts and papers for a more in depth discussion.

Waveform Coders

Waveform coders are used to represent and reconstruct accurately a digital speech signal on a sample-by-sample basis. This is accomplished by two methods. The first method exploits the redundant properties of the speech signal and is referred to as time domain waveform coding [10],[11]. The second method, frequency domain waveform coding, exploits the non-uniform distribution of speech information across the frequency spectrum [10],[11]. A discussion of the two methods follows.

Time Domain Waveform Coding

Pulse Code Modulation (PCM) [10],[11] is the most common method of time domain waveform coding. This method quantizes each sample of a speech signal to a specific amplitude. The number of bits, B, used to represent each sample determines the amplitude. The number of quantization levels is computed from B, as 2^B [12]. The bit rate of a PCM coder is found by multiplying the number of bits used to represent a sample by the sampling frequency.

A variation on PCM is differential PCM (DPCM) [10],[11]. The method of differential PCM takes advantage of the fact that samples (amplitudes) do not change much from one to another, resulting in a lower variance. This means that the difference between two samples can be quantized instead of quantizing a specific sample. Since the difference between two amplitudes is smaller on average than a specific amplitude, fewer bits will be needed thus lowering the overall bit rate of the coder.

A simpler version of DPCM is delta modulation (DM) [10],[11]. This coder uses a two level quantizer and a first-order predictor (usually linear prediction) to determine the quantized output signal with an associated quantization error. This results in a linear staircase function as the output signal.

There have been numerous attempts at refining PCM, DPCM, and DM. The most common approach is to make the quantization step size and the predictor adaptive. This resulted in a number of coders such as APCM, ADPCM, and ADM, that produced toll quality speech and communication quality speech at bit rates between 9.6 kbps and 16 kbs respectively [10],[11]. Also, vector waveform quantization, which takes advantage of the redundancies in speech signals, could be applied to the above methods [32]. Vector waveform quantization, while an integral part of many speech coders, will not be discussed in detail.

Frequency Domain Waveform Coding

The filter bank analyzer-synthesizer, usually a research coder, is representative of frequency domain waveform coding [10],[11]. This coder consists of a bank of bandpass filters that cover the entire frequency spectrum of interest. The speech signal is applied to the bank of bandpass filters and the outputs are then decimated for coding efficiency. The decimated outputs are quantized for transmitting. In the synthesizer, the transmitted signal is interpolated and input to the same bank of bandpass filters. The outputs are then summed producing synthesized speech. This coder does not provide better coding efficiency compared to the time domain methods [10].

An improvement on the filter bank analyzer-synthesizer is referred to as subband coding (SBC) [10],[11]. This method uses a bank of filters as in the previous method, but not as many. The frequency spectrum is divided non-uniformly into four to eight subbands and each of these bands is encoded using APCM. This is done because the low end of the spectrum is more important from a perceptual standpoint. Resulting in more bits being needed for coding. The high end of the spectrum does not contain as much information so fewer bits are used for coding. However, for this coder to achieve toll quality speech, four to five bands and 24 kbps is needed to code the entire spectrum.

Adaptive Transform Coding (ATC) segments the speech signal into frames of data, instead of filtering [10],[11]. These frames are pushed into a buffer and then transformed into a another form of representation, usually spectral. The transformed coefficients of the representation are quantized and transmitted to the synthesizer. At the synthesizer, the coefficients are inverse transformed back to the time domain. The bit rate is dependent on the number of bits used to code the coefficients. This type of coder has produced toll and communication quality speech at bit rates of 16 kbps and 9.6 kbps, respectively.

Classical Vocoders

Vocoders differ from waveform coders because they use a mathematical model to represent the speech signal where waveform coders reconstruct the speech signals on a sample-by-sample basis. In vocoders, the speech samples are represented as the output of a time varying linear system as shown in Figure 1. The inputs to the linear system make up the excitation parameters. A single voicing decision modeled as a switch, determines whether the excitation is either periodic or random. Voiced speech is modeled as a periodic impulse train with period equal to the fundamental frequency (pitch). The unvoiced speech is modeled using a psuedo-random white noise sequence. Vocal tract parameters are used to determine the shape of the waveform.

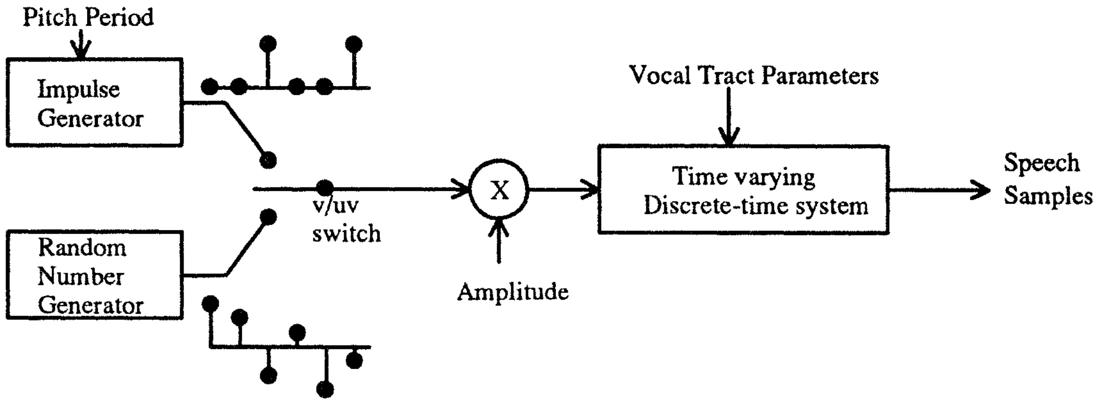

Figure 1. Block Diagram of Classical Vocoder.

In all vocoders, a set of parameters must be estimated and updated periodically by the transmitter (analyzer) for every frame. These parameters are usually the pitch, voiced/unvoiced decision(s), spectral representation, and possibly an associated gain value. The parameters are encoded, using one of the methods described earlier, then transmitted to the receiver. In the synthesizer, the parameters are decoded once for every analysis frame and the speech signal is reconstructed on a frame-by-frame basis using the underlying speech production model.

The vocoders discussed in the sections below target the range of 2.4 kbps to 9.6 kbps. These vocoders have been successful at producing communication quality speech at these bit rates and below. The following discussion pertains to a set of vocoders deemed classical. These classical vocoders are discussed to outline a basis for the modern vocoders, discussed in a later section.

Channel Vocoders

The channel vocoder uses a set of filters, covering the frequency spectrum nonlinearly, to estimate the slowly varying characteristics of the frequency spectrum [10],[11]. A single spectral amplitude is determined from each filter for each frame. The channel vocoder also has a single voicing decision and an estimated pitch for every analysis frame.

In the synthesizer the amplitudes are input into a set of matching filters and multiplied by a psuedo-white noise sequence or a periodic pulse train, with the period equivalent to the pitch, depending on the voicing decision. The outputs of the filters are then summed producing a frame of reconstructed speech. The channel vocoder can produce low quality intelligible reconstructed speech at 2.4 kbps.

Phase Vocoders

The phase vocoder like the channel vocoder uses a set of filters to estimate the magnitude spectrum. However, the phase vocoder does not use a pitch detector. This vocoder estimates the derivative of the phase for each filter output [10],[11]. The amplitudes and derivatives are encoded and transmitted to the synthesizer. At the synthesizer, the phase derivative is integrated and weighted by the corresponding amplitude.

The advantage of coding the phase derivative, instead of the phase, is lower bit rate. Using PCM for coding the amplitudes and a linear quantizer for the phase derivative, bit rates around 7.2 kbps can be achieved [10],[11]. The disadvantage for coding the phase derivative is the loss of timing information about the relative phase for the various harmonics in a speech signal. This disadvantage results in a substantial loss in quality. This loss in quality is so great that this vocoder has not been widely implemented [10],[11].

Cepstral Vocoders

The cepstral vocoder tries to exploit the difference between the excitation and the vocal tract response [10],[11]. This is accomplished by computing the cepstrum and estimating both the vocal tract response and the excitation spectrum. A short time window is used to determine the vocal tract response and a long time window is used to determine the pitch and voicing. These parameters are coded and transmitted to the synthesizer.

At the synthesizer, the reverse of the above process is used. The reconstructed speech is generated by convoluting either a noise sequence or a periodic pulse train, depending on the voicing decision, with the output of a vocal tract filter. The filters input is the transmitted cepstrum representation.

The cepstral vocoder is a computationally intensive algorithm, as two DFT's are required in the transmitter and receiver. Other problems exist in the coding of the cepstral parameters, such as noise induced by the logarithm operation. Some of the problems can be avoided by computing the complex cepstrum but this adds even more computation to the algorithm.

The cepstral vocoder has been shown to yield good quality speech. The problems mentioned in the previous section along with additional difficulties in needing an adaptive window for pitch prediction and loss of emphasis on low level noise because of the logarithm operations have been major drawbacks for wide spread use of the cepstral vocoder.

Formant Vocoders

As the name suggests, the formant vocoder attempts to code the formant frequencies of a frame of speech. The analyzer estimates the first three or four formants using either linear prediction or cepstral analysis [10],[11]. The formants, corresponding bandwidths, and the pitch are the parameters that are coded and transmitted to the synthesizer.

The output speech is reconstructed using a bank of two-pole filters with adjustable gain values. A sequence, either noise or periodic pulses, is generated based on the excitation parameters. This sequence is passed through the appropriate bank of two-pole filters and weighted by the gain.

This method of coding uses very few parameters, which results in bit rates around 1 kbps. The disadvantage of formant vocoders occurs when two formants are close together. Estimating formants that are close together is very difficult. This hinders the use of formant vocoders practically.

Modern Vocoders

The vocoders discussed in this section are defined in the same manner as the classical vocoders. The same set of parameters must be estimated: pitch, vocal tract response, voicing and gain. These vocoders target the bit rate range of 4.8 kbps and below.

Modern vocoders warrant a section separate of the classical vocoders because of new research in the area of speech coding. This new research consists of speech production models that allow for mixed excitation and more efficient methods of quantization, which results in lower bit rates with higher quality. The modern vocoders discussed below are representative of the different approaches which are being pursued by government, industry, and universities for the development of the next generation of vocoders.

The line drawn between classical and modern vocoders is slight. This paper draws the distinction in the following manner. The classical vocoders generally made a single voiced/unvoiced decision for a frame whereas most modern vocoders allow for mixed excitation to exist in a given frame. The block diagram shown in Figure 1 is representative of a classical vocoder. A switch is used to determine whether the excitation should be either voiced or unvoiced.

An example of a modern vocoder is one that allows for mixed excitation such as MBE. Shown in Figure 2a is the original spectrum for a frame of speech. As can be seen, the spectrum is voiced at the low end, then becomes unvoiced, and then voiced again.

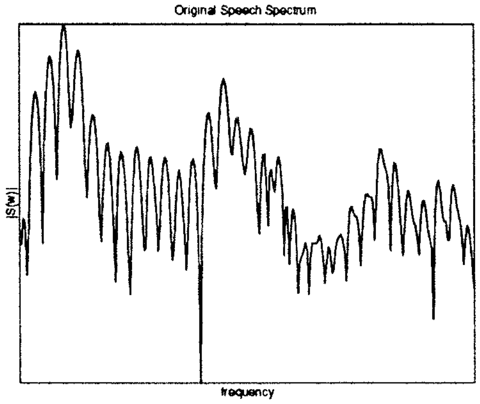

Figure 2a. Original Spectrum.

The modern vocoder estimates the same parameters as the classical vocoder. The vocal tract response is shown in Figure 2b. A voiced spectrum corresponding to a given pitch is shown in Figure 2c and a unvoiced spectrum is given in Figure 2e. The difference between the classical and modern vocoders can be seen in Figure 2d in that both voiced and unvoiced excitation exist in the same frame.

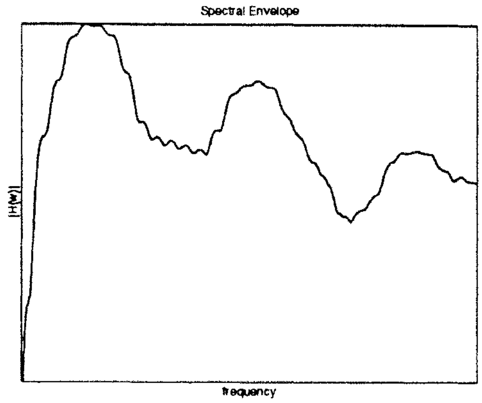

Figure 2b. Spectral Envelope.

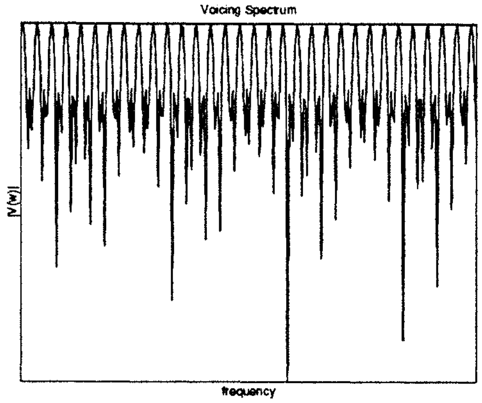

Figure 2c. Voiced Spectrum.

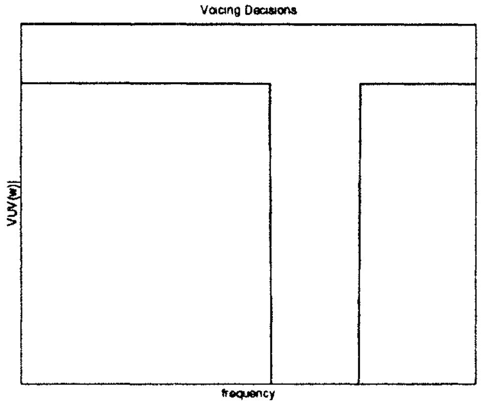

Figure 2d. V/UV Decisions.

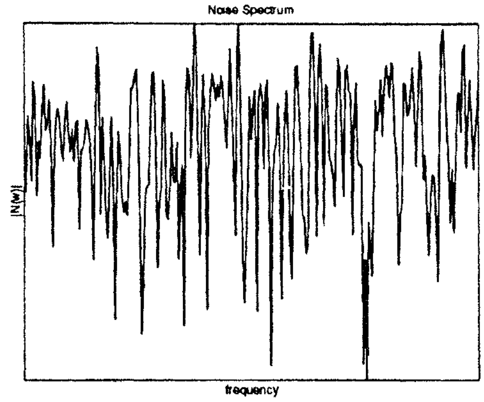

Figure 2e. Unvoiced Spectrum.

The mixed excitation spectrum shown, in Figure 2f, is generated by applying the voiced spectrum over the ranges where the spectrum was declared voiced. The unvoiced spectrum is applied over the range where the spectrum was declared unvoiced. The two spectrums are added together to produce the mixed excitation spectrum.

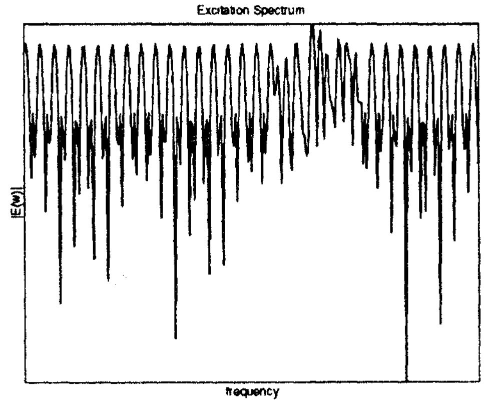

Figure 2f. Mixed Excitation Spectrum.

To generate the synthetic spectrum shown in Figure 2g, the mixed excitation spectrum is multiplied by the spectral envelope. This gives a frame of speech where the spectrum matches the original spectrum better than the method given for the classical vocoder. The classical method would have given a spectrum that was either all voiced or all unvoiced which would be incorrect.

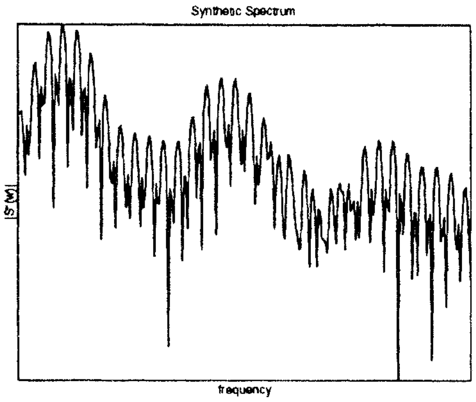

Figure 2g. Synthetic Spectrum.

Linear Predictive Coding

The most widely used method for speech coding is Linear Prediction Coding (LPC). Two specific coders will be discussed in this section. The first is Federal Standard 1015, also generally known as LPC-10e, which operates at 2,400 bps [13]. The second is Federal Standard 1016 or Code Excited Linear Prediction (CELP) operating at 4,800 bps [14],[15]. These speech coders are the basis for many other vocoders. The numerous variations will be mentioned but not discussed in detail.

LPC-10 is characterized by a pitch, a single voicing decision, and linear prediction coefficients [2]. The pitch is determined using a method based on the average magnitude difference function, while the voicing decision is based on zero crossing measures. A 10th order linear predictor is used to estimate the spectrum if the frame is voiced. If the frame is unvoiced then a 4th order linear predictor is used to represent the spectrum.

The receiver synthesizes the speech using a linear prediction synthesis filter. The excitation for this filter is determined by the pitch and voicing decision. If the frame is voiced then the excitation is an impulse train with period equal to the pitch. If the frame is unvoiced then the excitation is a random noise sequence.

There are numerous problems with this type of vocoder. The synthesized speech tends to have an artificial characteristic. As a result, the identity of the speaker becomes difficult to determine particularly at low bit rates. The quality of this type of coder degrades quickly in the presence of background noise. An algorithm which helps eliminate some of these problems is CELP.

Code Excited Linear Prediction (CELP) is an analysis by synthesis system [14]. The analyzer uses a codebook that contains zero-mean Gaussian excitation data along with a perceptually weighted error to determine an index into a codebook, which is transmitted to the receiver. Along with the index of the excitation vector, information about the pitch and spectrum is transmitted.

The receiver uses the same zero-mean Gaussian codebook plus two all-pole filters to synthesize the speech. The first filter is the pitch synthesis filter, which is updated periodically by the transmitted pitch data. The second filter, a linear prediction filter, is used to synthesize the spectrum. This filter is also updated periodically by the transmitted spectrum information.

CELP has been shown to produce toll quality speech at 16 kbps [2]. As the bit rate drops so does the quality of the synthesized speech. The lower bit rates exhibit a harshness in the output speech [10],[11].

CELP is computationally expensive because of the exhaustive searches performed on the codebook. There have been other coders which addressed the search problem and the quality problem mentioned above. This has resulted in CELP based coders achieving toll quality at 4.8 kbps [15]. Some of these coders are listed below along with a reference:

i) Residual Excited Linear Prediction (RELP) [10],[11].

ii) Binary Code Excited Linear Prediction (BCELP) [16].

ii) Vector Sum Excited Linear Prediction (VSELP) [17].

iv) Pitch Synchronous Excited Linear Prediction (PSELP) [18].

v) Low Delay Code Excited Linear Prediction (LD-CELP) [19].

All of the coders listed use CELP as a basis. For a more in depth discussion of each coder the reader is referred to the papers referenced.

Wavelets

The use of wavelets in a 2.4 kbps speech coder was introduced by John S. Baras and Edmund Butler [20]. The basis of this coder is CELP FS 1016. This vocoder was implemented with improvements to the original codebook and the Dyadic Wavelet Transform is used to determine the pitch [20].

As mentioned, the pitch is determined using a wavelet based pitch detector. If the pitch is determined with a high confidence level then the LPC coefficients and pitch gain are determined simultaneously. When the pitch is determined with a low confidence level, the parameters are computed as in CELP FS 1016. The speech reconstruction is also computed as CELP FS 1016.

The advantages are improved speech quality because of a more stable pitch. Improvements in the codebook reduce computational overhead. A reduction in the bit rate will occur by encoding the true pitch instead of the adaptive codebook approximation. The disadvantages are the possibility of long delays in this vocoder and that this vocoder has not been proven at the present time.

Sinusoidal Transform Coding

A speech analysis/synthesis system based on a sinusoidal representation was introduced by Robert J. McAulay and Thomas F. Quatieri [21],[24]. This speech coding technique is referred to as Sinusoidal Transform Coding (STC). The speech waveform is characterized by amplitudes, frequencies, and phases. There are no voicing decisions to make in this algorithm.

The analyzer takes a windowed segment of the original speech and performs a DFT. Using the magnitude of the DFT, the frequencies and amplitudes of each harmonic for the frame are estimated. The phase is estimated using the DFT and the estimated frequencies. These parameters are coded and transmitted to the receiver.

In the synthesizer, a sinusoid is generated for each harmonic in the current frame. This sinusoid is generated using the transmitted frequency and phase, then amplitude modulated. Adjacent frames are smoothed by using a birth and death approach for each harmonic. All the harmonics for the frame are added together producing synthetic speech.

STC is claimed to produce high quality speech for various types of signals such as: quiet speech, multispeaker waveforms, music, speech with background noise, and marine biological signals. The STC vocoder was a candidate for the Federal 4.8 kbps standard, but finished well behind the other candidates in all tests [25].

Time-Frequency Interpolation Coding

The Time-Frequency Interpolation (TFI) vocoder was introduced by Yair Shoham [22]. This vocoder is an efficient implementation of a method referred to as prototype waveform interpolation (PWI) [2].

The analyzer for TFI proceeds in the following manner. First, LPC analysis is performed on the input speech to produce a spectrum estimate. The LPC parameters are quantized and transmitted to the receiver. Next, a pitch and voicing algorithm is needed to estimate the pitch for each frame. If the frame is unvoiced, the base CELP model is used to generate the unvoiced excitation. If the frame is voiced, the pitch, voicing, LPC spectrum, and original speech signal is propagated to the TFI algorithm.

In the TFI algorithm, a DFT is computed, weighted, and vector quantized. The quantized spectrum is then interpolated at sub samples of the pitch. The voiced excitation is found from the interpolation. This excitation is coded and transmitted to the receiver.

The synthesizer consists of an all-pole LPC filter. All the transmitted data is input to the filter and synthetic speech is output. The reconstruction is actually similar to STC although the implementation is not the same.

This vocoder has been reported to produce high quality speech at low bit rates. The bit rates tested were 4.05 kbps and 2.5 kbps. These vocoders were compared to two higher bit rate standards: the 13 kbps European standard GSM coder and the 8 kbps North American cellular standard IS54 [22]. Both TFI coders performed approximately equal to one another and to the two standards.

MulitBand Excitation Coding

MultiBand Excitation (MBE) Coding is the model of most interest in this study. This speech model was introduced in 1988 by D.W. Griffin and J.S. Lim [1].

The MBE model for speech follows the same general scenario as the other vocoders discussed in this section. The speech is modeled with the same three basic parameters: pitch, voiced/unvoiced decisions, and vocal tract response. The MBE model assumes that both voiced and unvoiced excitation can exist at the same time in the same analysis frame. This creates the need for multiple voiced/unvoiced (V/UV) decisions [4].

Mixed excitation is modeled in the frequency domain, rather than the time domain. The spectrum of the speech is split into non-overlapping bands, and each band is modeled as being either voiced or unvoiced. In the MBE synthesizer, the reconstructed frame is formed by combining the appropriate voiced or unvoiced reconstruction over each frequency band such that the entire spectrum is covered. The aggregate synthetic signal spectrum can thus exhibit mixed excitation.

With MBE, the voiced bands are synthesized using sinusoidal oscillators tuned to harmonics of the estimated pitch. Each harmonic which was declared voiced by the analyzer is reconstructed this way. Each unvoiced band is reconstructed using bandpass filtered noise. In both cases, the reconstruction for each harmonic is weighted by a spectral amplitude estimated by the analyzer for the corresponding harmonic frequency.

The MBE speech model has demonstrated that high quality speech can be achieved at low bit rates. In fact, an improved version of MBE, known as IMBE, was adopted by INMARSAT for satellite voiced communications and by APCO [28],[33]. Other advantages of using MBE are the robustness to additive noise and the ability to implement a real time system [23].

CHAPTER III

DESIGN

Overview

The description of the Enhanced MultiBand Excitation (EMBE) vocoder has been separated into three main topics: analyzer, synthesizer, and quantizer. Before discussing these three sections, an overview of the MultiBand Excitation speech model and the associated parameters is provided [1].

A vocoder (speech coder) consists of two operational parts: a transmitter and a receiver. A block diagram of a typical speech coder is shown in Figure 3.

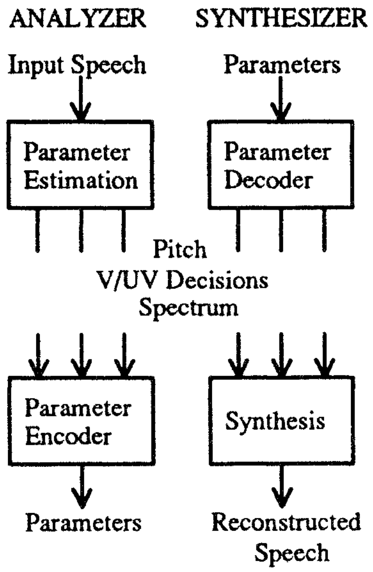

Figure 3. Functional Diagram of a Typical Vocoder.

The analyzer is responsible for determining on a short time basis certain characteristics of the input speech signal. The appropriate characteristics are determined by the specific speech model which is being used. In most cases pitch (fundamental frequency), a single voicing decision, and vocal tract spectrum comprise the desired parameters to be estimated from the input speech signal. These parameters form a vector which describes the current input speech frame. Once estimated, the parameter vectors are properly coded and transmitted to the synthesizer for reconstruction.

At the synthesizer, the analysis model is applied in reverse. The goal of the synthesizer is to produce the best sounding speech signal without regard for how the sample to sample values correspond to the actual input waveform.

The MBE speech model follows the general scenario described above, modeling speech with the same three basic parameters. However, the MBE model differs by assuming that both voiced and unvoiced excitation can exist at the same time in the same analysis frame. This creates the need for multiple voiced/unvoiced (V/UV) decisions.

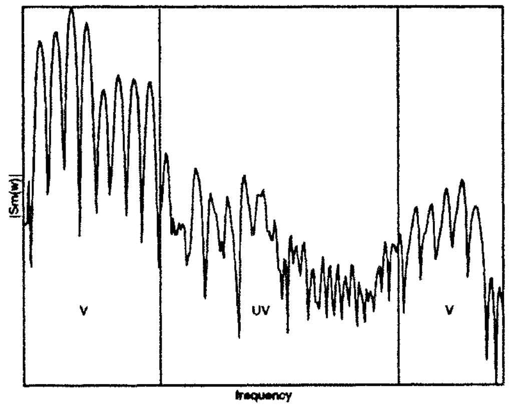

Figure 4. A Typical MBE Speech Spectrum Showing Voiced and Unvoiced Bands.

Figure 4 shows graphically what the spectrum of a frame of speech with mixed excitation might look like with the MBE model. The voiced and unvoiced regions are clearly visible. For example, observe the voiced band at the low end of the spectrum, followed by a wide unvoiced band, then another narrow voiced band.

Mixed excitation is modeled in the frequency domain, rather than the time domain. The spectrum of the speech is split into non-overlapping bands, and each band is modeled as being either voiced or unvoiced. In the MBE synthesizer, the reconstructed frame is formed by combining the appropriate voiced or unvoiced reconstruction over each frequency band such that the entire spectrum is covered. The aggregate synthetic signal spectrum can thus exhibit mixed excitation.

The voiced bands, with MBE, are synthesized using sinusoidal oscillators tuned to harmonics of the estimated pitch. Each harmonic which was declared voiced by the analyzer is reconstructed this way. Each unvoiced band is reconstructed using bandpass filtered noise. In both cases, the reconstruction for each harmonic is weighted by a spectral amplitude estimated by the analyzer for the corresponding harmonic frequency.

The following sections describe in detail the Enhanced MBE speech coder. The analyzer (transmitter) is described first, followed by the synthesizer (receiver) and the quantizer.

Analyzer

A block diagram of the analyzer is shown in Figure 5 below. The analyzer consists of several distinct stages: pre-processing and framing, pitch estimation, spectrum modeling, V/UV decisions, and parameter quantization. The input speech signal is assumed to be 16-bit data sampled at 8,000 samples per second. Each of these stages will be discussed separately in the following sections.

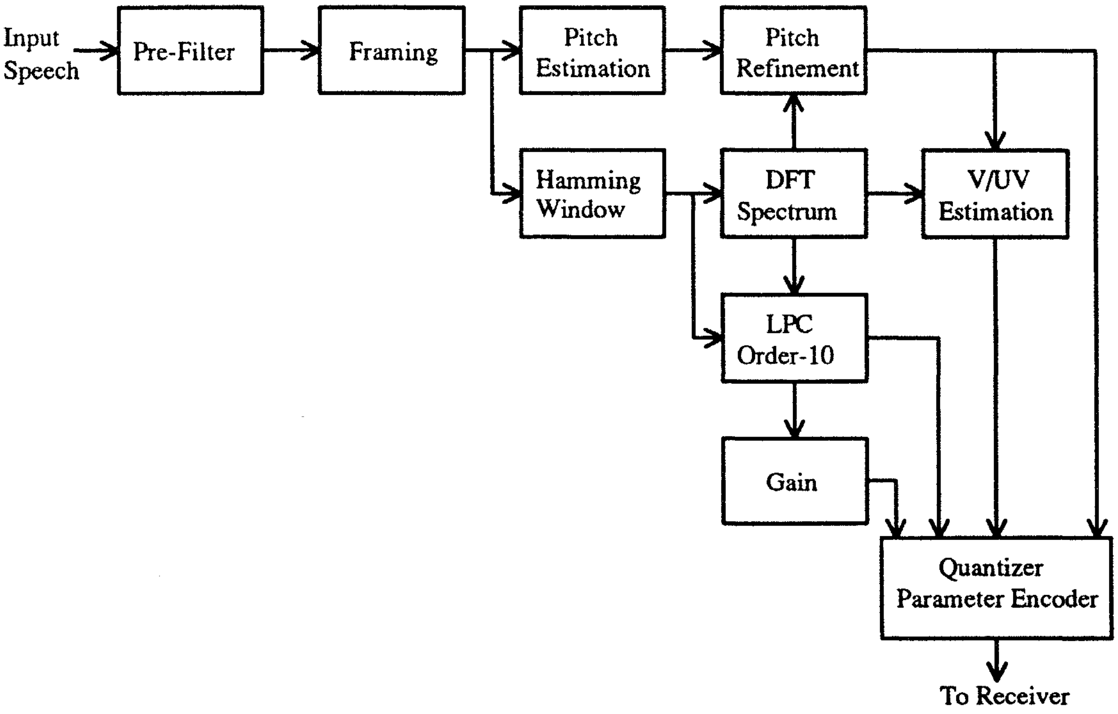

Figure 5. Block Diagram of the Enhanced MBE Analyzer.

Pre-Processing and Framing

Once the speech model and bit rate have been chosen then a method for pre-processing, framing, and windowing is considered. Pre-processing consists of input filtering, framing, and windowing of the raw speech data stream.

In this study, the raw input signal is filtered prior to framing by a simple first-order recursive highpass filter as in equation (1) below. This filter reduces or eliminates signal energy below about 10 Hz.

$$H(z) = \frac{\frac{1}{2} - \frac{1}{2}z^{-1}}{1 - 0.99726z^{-1}} \tag{1}$$

After filtering, the data needs to be windowed to generate a frame of data for processing. This is accomplished with a Rectangular window of some specified length. The frame size (window length) needs to be chosen to correspond to the bit rate and the basic frame increment. If the window is going to overlap, the amount of previous and future data to be included in estimating the parameters should be considered when choosing a window length. If the data is to be windowed again before processing, a window type and length corresponding to the analysis frame size needs to be chosen. The frame size and window lengths are discussed in the following section.

The Enhanced MBE vocoder operates on a basic frame increment of 20 ms, or N=160 samples. This frame increment corresponds to 50 frames per second. Each frame is designed to overlap slightly the immediate previous and subsequent frames to increase smoothness of the parameter estimates. The analysis frame length is 240 points, consisting of an overlap of 40 points ahead and behind the current frame.

The filtered analysis frames are passed directly to the pitch detector. Analysis frames headed for pitch refinement, V/UV estimation, and spectrum modeling are first windowed with a Hamming window per equation (2). The results are then passed to the

$$h(n) = \begin{cases} 0.54 - 0.46\cos\left(\frac{2\pi n}{N-1}\right) & 0 \le n \le N-1 \\ 0 & otherwise \end{cases} \tag{2}$$

respective blocks. Because the frames are overlapping, there needs to be a way of emphasizing the basic frame increment (non-overlap) and de-emphasizing the overlap. A Hamming window is one method for accomplishing this task. The Hamming windowed (analysis) speech frames are referred to as $s(n)$, without regard to frame number, in the following discussion.

Pitch Estimation

The current implementation of the Enhanced MBE vocoder uses a pitch detector based on the AMPEX algorithm by Van Immerseel and Martens [5]. A detailed discussion of the AMPEX algorithm can be found in their referenced paper. This pitch detector is reported to have very good performance over a wide range of input conditions [6]. Further, an implementation in C was already available.

The table below shows the results of three different generations of pitch detection algorithms. The first generation pitch detector is PPR. This is a parallel processing algorithm developed by B. Gold and L. Rabiner [6]. The second generation algorithm is a subharmonic summation (SHS) pitch detector developed by D.J. Hermes [6]. The third generation pitch detector is AMPEX developed by Van Immerseel and Martens. The corr column is a reference pitch contour.

The values in the table represent the number of correct or incorrect frames. AMPEX performed quite satisfactorily compared to PPR and SHS. In each set of data, AMPEX appears to combine the best of both PPR and SHS when making voicing decisions and estimating the pitch.

Table 1: Results of Three Generations of PDAs to 'clean', 'dirty', 'timit', and 'spont' [6].

	'clean.wav'				'dirty.wav'			
	PPR	SHS	AMPEX	corr.	PPR	SHS	AMPEX	corr.
correctly voiced	176	197	196	198	152	179	149	198
pitch correct	164	197	195	198	114	165	148	198
pitch incorrect	12	0	1	0	38	14	1	0
unvoiced errors	22	1	2	0	46	19	49	0
correctly voiced	141	112	142	157	131	126	155	157
voiced errors	16	45	15	0	26	31	2	0
	'timit.wav'				'spont.wav'			
	PPR	SHS	AMPEX	corr.	PPR	SHS	AMPEX	corr.
correctly voiced	191	204	204	215	29	33	18	41
pitch correct	157	196	201	215	15	29	18	41
pitch incorrect	34	8	3	0	14	4	0	0
unvoiced errors	24	11	11	0	12	8	23	0
correctly voiced	165	136	153	165	149	193	231	237
voiced errors	0	29	12	0	88	44	6	0

A pitch estimate is returned by the AMPEX algorithm every 10 ms or every 80 points (exactly twice per analysis frame). A processing delay of approximately 50 ms is imposed by the algorithm in the current implementation.

Since AMPEX returns a pitch twice per analysis frame a method to determine a single pitch value was needed. Initially, a nonlinear filtering scheme that consisted of a set of rules and thresholds was applied to the estimated pitches and to produce a single smoothed pitch estimate for each complete analysis frame. A set of rules and thresholds was used to determine a single pitch estimate. This method proved to have problems because the testing for high and low frequency pitches is not mutually exclusive. For example, a set of tests which make correct estimates at low frequency pitches would fail at high frequency pitches. These results led to the development of the pitch refinement stage, which is discussed in the following section.

MBE based vocoders require sub-sample pitch resolution for accurate representation of the harmonic spectrum. Unfortunately, the implementation of AMPEX which is being used does not return pitches with acceptable accuracy for this coder. An additional pitch estimation step, which will be referred to as pitch refinement, is necessary to produce sub-sample accuracy. The pitch refinement step is performed on the pitch initially estimated by AMPEX for each analysis frame. The refinement procedure is based on matching the original speech spectrum for the current frame with the spectrum which would be generated if a particular pitch were used in the synthesizer. The newly generated spectrum will be referred to as the *synthetic spectrum*.

The refinement assumes that the original pitch from AMPEX is close to the correct pitch. The pitch is then allowed to vary above and below the originally estimated pitch, and the spectral errors between the original and synthetic spectra are computed over a set of harmonic frequencies. Spectral errors are used to determine the refined pitch.

A number of other subjective criteria are then applied to the pitch which produces the synthetic spectrum with the lowest error. First, the pitch is assumed to lie between 65 Hz and 380 Hz approximately. Any pitch estimate which falls outside this interval is assumed to be invalid. Further, any abrupt change in pitch between frames is considered suspect, and smoothness criteria are applied to limit the amount of change which is possible. The *best* overall pitch is finally chosen as the refined pitch. Note that in the discussion and equations that follow, pitch is in terms of radian frequency rather than Hertz.

Pitch refinement is broken into two steps. The first step involves choosing a single pitch ω, for the current analysis frame, from the two 80-point pitches, ω'_o and ω'_{-1}, returned by AMPEX. Here ω'_o and ω'_{-1} refer to the most recent and next most recent 80 point pitches, respectively, produced by AMPEX. The variables ω_o and ω_{-1} will be used to refer in the same way to the two most recent refined pitches. However, in this last instance ω_o and ω_{-1} are actually full 160-point pitches. (Note: In most instances throughout this discussion, a subscript of "o" refers to something *current,* while a subscript of "-1" refers to something *previous.*) The second step involves refining ω to a single pitch estimate ω_o for the current analysis frame. The following is a discussion of these two steps.

Step 1: In this step, a single candidate pitch ω is selected for further refinement in Step 2. By observation, a frame to frame pitch variation of about 10% is typical, while a variation of more than about 25% is rare [30]. Also known by observation, AMPEX is more likely to return poor pitch estimates when the input speech is less voiced. In fact AMPEX may return poor pitch estimates for both ω'_o and ω'_{-1} in this latter case.

In selecting ω, first compare the two 80-point AMPEX pitch estimates ω'_o and ω'_{-1} to the pitch ω_{-1} selected in the previous frame as shown in the equations below

$$\begin{array}{ll} \frac{\omega_{-1}}{\omega'_o} < 0.75 & \frac{\omega_{-1}}{\omega'_o} > \frac{1}{0.75} \\ \frac{\omega_{-1}}{\omega'_{-1}} < 0.75 & \frac{\omega_{-1}}{\omega'_{-1}} > \frac{1}{0.75} \end{array} \qquad (3a,b,c,d)$$

A variation of more than 25% from ω_{-1} (one or more of these inequalities being true) is used to indicate the strong possibility that one or both of the current 80-point pitch estimates is suspect. Incidentally, this also includes the possibility that the pitch has doubled or halved. In such a case, the previous refined pitch ω_{-1} is included as one of the candidates. Thus, a pitch ω is selected from among the pitch candidates ω'_0 and ω'_{-1} and possibly ω_{-1}.

Once a set of two or three candidate pitches has been chosen, the selection process continues as follows. For each of the candidate pitches, a corresponding pitch period P_o (in samples) is determined as

$$P_o = \tfrac{2\pi}{\Omega} \tag{3}$$

where Ω is some specified candidate pitch selected from the set of candidate pitches under investigation.

Since MBE requires approximately quarter-sample accuracy to properly represent the highest harmonics in the spectrum, a new set of candidate pitches is generated from each of the original candidates. The new sets each consist of eleven candidates at integer multiples of one quarter sample above and below the original pitch periods, or within ±1.25 samples. For example, with an original candidate pitch of period P_o, a new set of candidates take the form

$$\{P_{P_o-1.25} \ldots\ldots, P_{P_o-0.25}, P_{P_o}, P_{P_o+0.25}, \ldots\ldots P_{P_o+1.25}\} \tag{4}$$

Now, for each candidate pitch period P with corresponding radian pitch Ω, a synthetic voiced spectrum $S_{\Omega}(m)$ is generated as the product of a unity amplitude harmonic function $H(m)$ with the DFT magnitude spectrum $S(m)$ of the current analysis frame as

$$S_{\Omega}(m) = S(m)H(m) \qquad 0 \le m < 256 \tag{5}$$

where

$$S(m) = \left| \sum_{n=0}^{511} s(n) e^{-j\frac{2\pi mn}{512}} \right| \tag{6}$$

$$H(m) = \sum_{r=0}^{L_o-1} \hat{h}(m - rP_o) = \sum_{r=0}^{L_o-1} h(m - \tfrac{P_o}{2} - rP_o) \tag{7}$$

$$h(m) = \begin{cases} 0.54 - 0.46\cos\left[\frac{2\pi m}{upper-lower}\right] & 0 \le m \le (upper - lower) \\ 0 & otherwise \end{cases} \tag{8}$$

$$L_o = \left\lfloor \frac{0.975\pi}{\Omega} \right\rfloor \tag{9}$$

and

$$lower = \left\lceil (\tfrac{512}{2\pi})(l - 0.5)\Omega \right\rceil \tag{10a}$$

$$upper = \left\lceil (\tfrac{512}{2\pi}(l + 0.5)\Omega \right\rceil \qquad 1 \le l \le L_o \tag{10b}$$

The equations above describe how each harmonic in the synthetic spectrum is built by generating a Hamming window centered in the spectrum at integer multiples of the pitch with period P_o to form the desired ideal harmonic spectrum as shown in Figure 6.

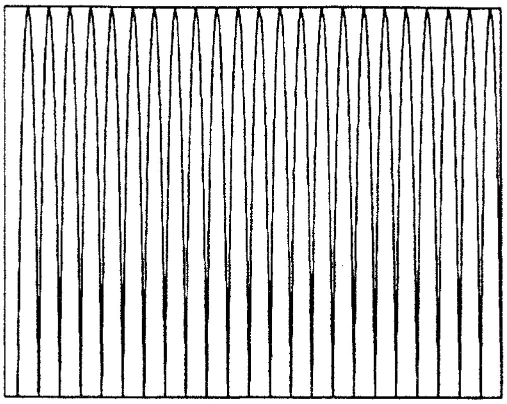

Figure 6. Spectrum Built Using Replicas of a Hamming window.

The number of harmonics, L_o, in the synthetic spectrum is a function of the pitch and is computed as shown above. The *lower* and *upper* limits refer, respectively, to the lower and upper band edges of the l^{th} harmonic. These limits must be computed for every harmonic in the synthetic spectrum. Once every voiced harmonic has been computed, the entire ideal synthetic harmonic spectrum corresponding to the current candidate pitch will have been formed.

After the synthetic spectra are built for all the candidate pitches, the overall squared error between each synthetic spectrum and the original spectrum is computed as shown in equation (11).

$$e_\Omega(j) = \sum_{m=start}^{end} [S(m) - S_\Omega(m)]^2 \qquad 0 \le j < 11 \qquad (11)$$

The spectral error in equation (11) is intentionally computed over a very limited range of frequencies. In particular these limits, given below, correspond to the fourth through the ninth harmonic of the pitch.

$$start = \lceil (\tfrac{512}{2\pi})(4 - 0.5)\Omega \rceil \tag{12a}$$

$$end = \lceil (\tfrac{512}{2\pi})(9 + 0.5)\Omega \rceil \tag{12b}$$

These limits were chosen somewhat arbitrarily, but they correspond generally to the range of harmonics where the *correct* synthetic spectrum (the one corresponding to the best pitch) is most likely to match the original spectrum as shown in Figure 7. This is because the lower portion of the spectrum is the most likely part to be voiced (if there is any voicing present at all). The lowest harmonics are not considered since they may be missing if the original speech is bandlimited. A poor match (incorrect pitch) between the original spectrum and synthetic spectrum is shown in Figure 8. In both Figure 7 and 8 the dashed line is the synthetic spectrum.

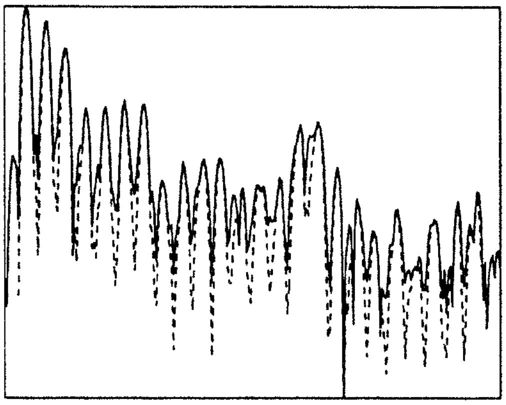

Figure 7. A Close Match Between the Original Spectrum and the Synthetic Spectrum (--) for Step 1.

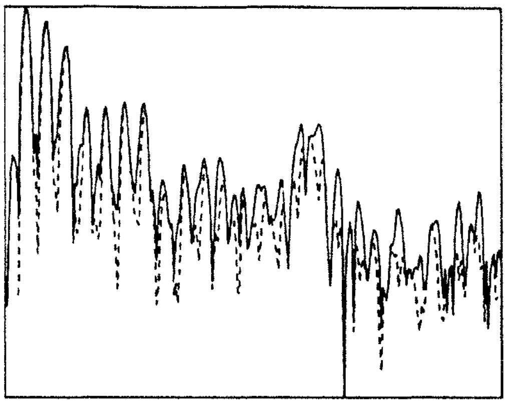

Figure 8. A Poor Match Between the Original Spectrum and the Synthetic Spectrum (--) for Step 1.

The candidate pitch corresponding to the synthetic spectrum with the lowest overall error $e_{\omega}(j)$ is chosen as the best pitch of the set under consideration. This pitch is passed to Step 2 for final refinement.

It is important to note that this procedure (Step 1) is not intended to return a final pitch value; it is designed to return a pitch which is only *close* to the correct value. Obtaining an initial close match allows us to eliminate most *bad* pitches (doubled, halved, or generally erroneous pitches from AMPEX) from further consideration without having to worry at this stage about achieving quarter-sample accuracy. Further, this procedure functions to consolidate two 80-point pitches from AMPEX, and possibly the previous 160-point refined pitch, into a single 160-point candidate pitch ω for the current analysis frame.

Step 2: The second stage of the algorithm takes the final 160-point candidate pitch ω from Step 1 and performs a refinement operation on it alone to one-quarter sample accuracy. The candidate pitch ω is assumed to be a valid pitch which is within ±1.75 samples of the correct pitch period.

As in Step 1 a set of candidate pitches at one-quarter sample increments and bracketing the input pitch is formed. In this step, the set consists of 15 candidate pitches including the pitch ω. Using the previous notation, the final set of candidate pitches might be written as

$$\{P_{P_o-1.75}\ldots\ldots, P_{P_o-0.25}, P_{P_o}, P_{P_o+0.25}, \ldots\ldots P_{P_o+1.75}\} \qquad (13)$$

This refinement step is further modified in that the synthetic harmonic spectra are built more accurately than before. In the first step, the harmonic spectra are built as (time domain) replicas of a Hamming window. In this stage replicas of a 16,384-point DFT of the Hamming analysis window are used to build the harmonic spectra, shown in Figure 9.

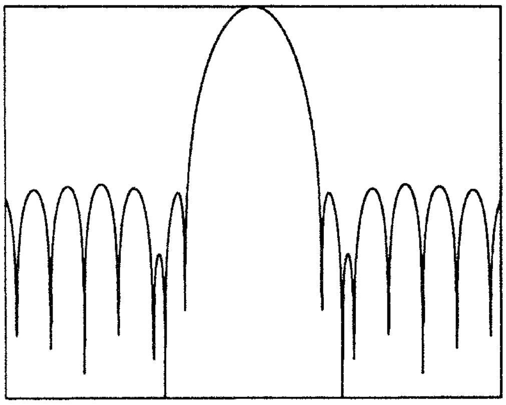

Figure 9. Spectrum of a Shifted Hamming Window (16,384 DFT).

Frequency domain interpolation is performed using this over-sampled window to achieve the desired accuracy. This refinement procedure is similar to the one documented in INMARSAT IMBE [28], except for the number of pitch candidates that are selected and the way the method computes the error criterion.

The synthetic harmonic spectrum can be written as

$$S_{\omega}(m) = A_l H\left(\left\lfloor 32m - \frac{16384\omega l}{2\pi} + 0.5 \right\rfloor\right) \qquad \mathit{lower} \le m < \mathit{upper} \quad (1 \le l \le L_o) \tag{14}$$

where

$$A_l = \frac{\sum_{m=\mathit{lower}}^{\mathit{upper}-1} S(m) H\left(\left\lfloor 32m - \frac{16384\omega l}{2\pi} + 0.5 \right\rfloor\right)}{\sum_{m=\mathit{lower}}^{\mathit{upper}-1} \left| H\left(\left\lfloor 32m - \frac{16384\omega l}{2\pi} + 0.5 \right\rfloor\right) \right|^2} \tag{15}$$

Figures 10 and 11 below show how the synthetic spectrum matches the original spectrum for a correct pitch estimate and for a poor pitch estimate. When compared to the spectral matches in Step 1, the spectral matches of Step 2 tend to match more of the original spectrum.

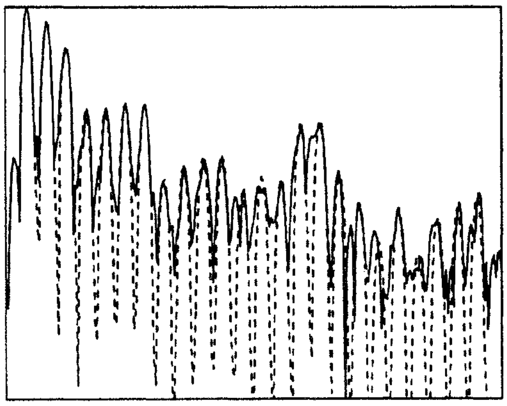

Figure 10. A Close Match Between the Original Spectrum and the Synthetic Spectrum (--) for Step 2.

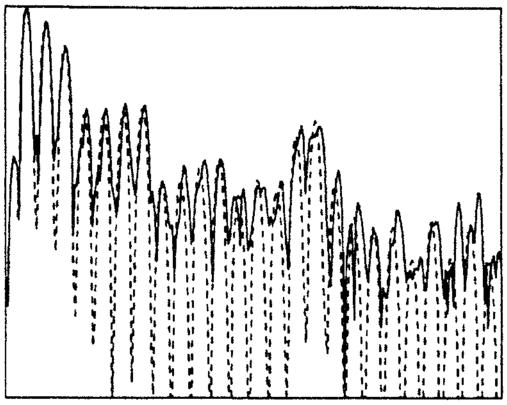

Figure 11. A Poor Match Between the Original Spectrum and the Synthetic Spectrum (--) for Step 2.

As before, an error measure is computed for each synthetic spectrum as shown in equation (16). This error measure differs from that of equation (11) because the range is

$$e_{\omega}(j) = \sum_{m=39}^{\lfloor\lfloor \frac{.9\pi}{\omega} - 0.5\rfloor \frac{512\omega}{2\pi}\rfloor} [S(m) - S_{\omega}(m)]^2 \qquad 0 \leq j < 15 \qquad (16)$$

now a function of the frequency ω and represents a larger portion of the spectrum. The larger range of spectrum for error measure is a result of the closer synthetic spectrum matches in Step 2.

The pitch corresponding to the lowest error is chosen as the final refined pitch, ω_o. The synthetic harmonic spectrum corresponding to the refined pitch and the refined pitch are passed to the voiced/unvoiced routine. The refined pitch is also passed to the quantizer for transmission.

Voiced / Unvoiced Decisions

The V/UV estimate is at the heart of the MBE model. The MBE model assumes that the speech spectrum can be composed of both voiced and unvoiced bands. This is equivalent to considering the excitation to contain both periodic and aperiodic components simultaneously. Simple observation of short time speech spectra will reveal that this mixed mode occurs regularly in real speech, leading to the conclusion that it should be included in the vocoder model if natural sounding speech is desired.

Grouping harmonics into threes is a common method for making V/UV decisions in MBE based vocoders. This band structure allows for a single V/UV decision to be made for each band of three harmonics. This reduced the number of V/UV decisions that must be transmitted for each frame of speech. Grouping into bands is a reasonable thing to do as there is high probability that a single voiced harmonic will have at least one neighboring harmonic which is also voiced.

The initial method used for making V/UV decisions was computing a ratio of the energy in a spectral harmonic to a smoothed spectral energy in the band around the harmonic. If the ratio exceeded a threshold, then the harmonic was declared voiced. Otherwise, the harmonic was declared unvoiced.

As in pitch estimation, accurate estimation of the voicedness of speech is a difficult task. The initial method proved to have significant errors, especially if the pitch estimate was in error. If the pitch error was significant, locating the harmonics became a difficult task. This error prompted the design of a band-based varying threshold technique presented in the following section.

The varying threshold technique uses the current analysis frame and a synthetic (voiced) harmonic spectrum, corresponding to the refined pitch and formed during the last phase of pitch refinement, for making voiced/unvoiced decisions. Specifically, harmonic error terms (based on the degree of match between the two spectra) together with the refined pitch provide the required information.

The voiced/unvoiced decisions are made on a harmonic-by-harmonic basis using an adaptive threshold technique [27]. The threshold function, $T(\omega_o,e(l),L_o)$, is based on several observed properties of quiet speech and extensive listening tests. Relevant observations which were used to develop the threshold function include the following:

i) The voiced spectrum of a high pitched speaker matches closely the synthetic spectrum over most of the frequency range of interest.

ii) The voiced spectrum of a low pitched speaker matches the synthetic spectrum best at lower frequencies.

iii) Voiced speech tends to be strongly voiced at low frequencies, falling off, often rapidly, at higher frequencies.

iv) Frames of speech that are completely unvoiced typically do not match the synthetic spectrum except at a few random harmonics.

v) The spectrum of typical speech contains more low frequency energy than high frequency energy.

The first observation is due to the fact that error in the synthetic spectrum is multiplicative with harmonic number. Making robust V/UV decisions requires that each harmonic of the pitch be located accurately. A small pitch error becomes much more pronounced in the synthetic spectrum as the number of harmonics increases. For example, a 5 Hz error with a 100 Hz actual pitch will produce a 50 Hz error when trying to locate the tenth harmonic. Since higher pitches have fewer harmonics, the multiplicative frequency error at the highest harmonic typically remains low. Since the actual DFT spectrum is being compared with a synthetic harmonic spectrum, this generally results in better accuracy (lower error) with higher pitches as can be observed in Figure 12.

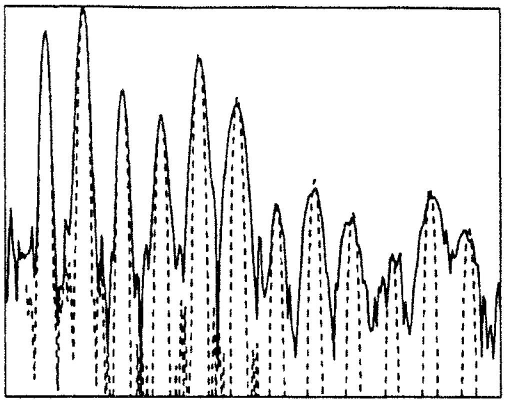

Figure 12. Match Between the Original Spectrum and Synthetic Spectrum (--) at High Pitch.

The second observation follows from the first. The synthetic harmonic spectrum constructed is generally more accurate at lower frequencies. In practice, the upper harmonics can exhibit large errors when the pitch is low since there are many harmonics and a potential for large multiplicative errors. This can be observed in Figure 13 below. The synthetic spectrum matches close in the low frequency range but in the higher frequency range the harmonics are a poorer match. This leads to the conclusion that the error at higher harmonics may be due to errors in pitch estimation. Voiced/Unvoiced decisions are thus highly dependent on an accurate pitch estimate, particularly when the pitch is low. The threshold is designed to make it more difficult to declare harmonics voiced when the pitch is low, thus becoming increasingly more difficult at the higher frequencies.

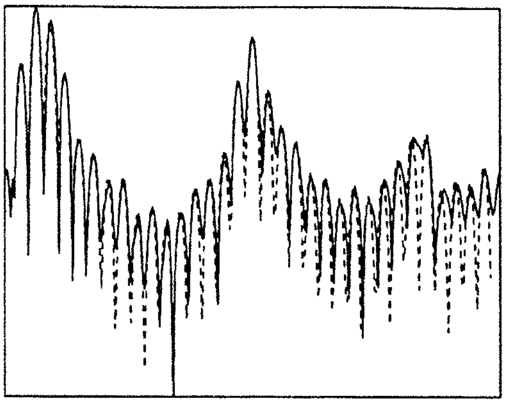

Figure 13. Match between the Original Spectrum and Synthetic Spectrum (--) at Low Pitch.

The third observation reinforces the first two. If a frame is going to be voiced at all, it is most likely that the lower harmonics will be voiced first. For frames with mixed excitation, as presented in Figure 4, the lower harmonics are more likely to be present. As the harmonic number increases, the probability that any particular harmonic will be present decreases. This decrease in likelihood often occurs quite rapidly above a certain point in the spectrum, with most or all of the harmonics above this point being unvoiced. The threshold is designed to make a voiced decision more difficult above a certain variable frequency in the spectrum.

The fourth observation simply supports the concept being used to determine voicedness. In particular the spectrum of a voiced frame exhibits a large amount of harmonic structure while the spectrum of an unvoiced frame exhibits little if any harmonic structure. Figure 14 shows a spectrum of an unvoiced frame being compared to the original spectrum for that frame. Observe that some harmonics could randomly be declared voiced.

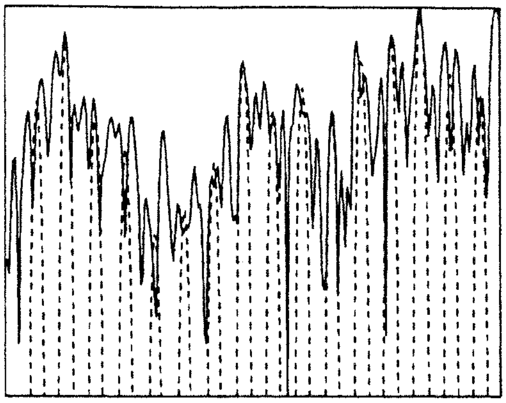

Figure 14. Original Unvoiced Spectrum with Synthetic Spectrum (--).

The last observation regarding the frequency balance in typical speech can be used to make more or less blanket voicing decisions about a frame. This observation is used specifically to declare frames unvoiced when an apparently valid pitch is returned by mistake. If the spectral balance of the frame is not typical of speech, the entire frame is declared unvoiced. This occurs most often when a pitch is returned for a low energy unvoiced or *silent* frame. In either case, declaring any harmonics voiced would be incorrect, so the condition is avoided. An example of a frame of speech with a spectral imbalance is shown in Figure 15. Observe that the energy in the low frequency range is lower than the energy in the high frequency range.

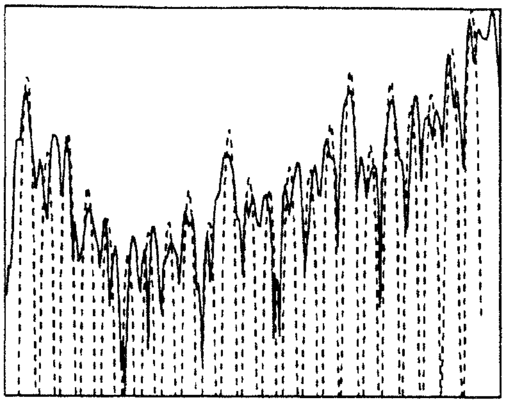

Figure 15. Original Spectrum with Imbalance and the Synthetic Spectrum.

These observations led to a threshold function of the form shown graphically in Figure 16 below.

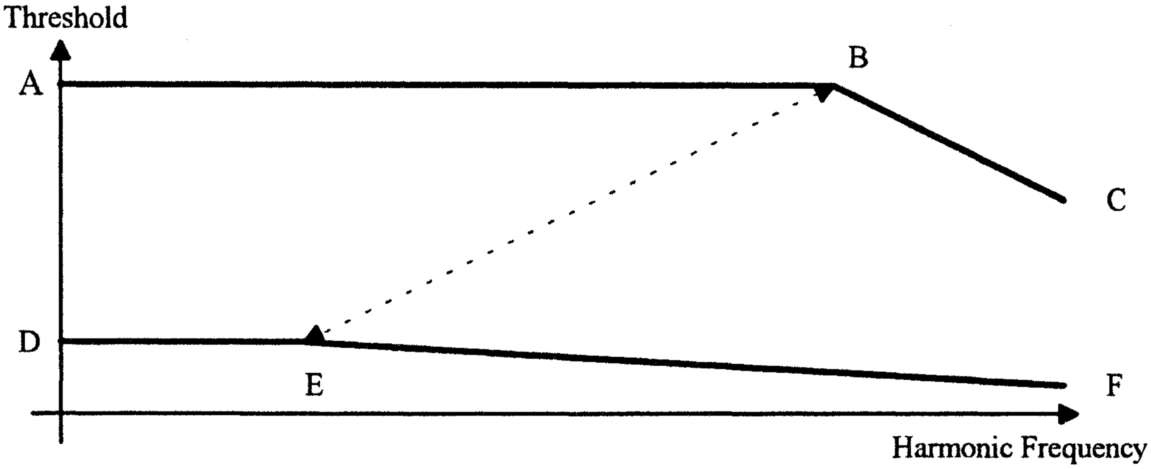

Figure 16. The Adaptive Voiced/Unvoiced Threshold Function.

Along with the adaptive threshold, an error term, $e(l)$, is computed for each harmonic. If $e(l) < T(\omega_o, e(l), L_o)$, the harmonic is declared voiced. Otherwise the harmonic is declared unvoiced. This error term is not the same as that computed in the pitch refinement section. This error is computed harmonic-by-harmonic referenced to the refined pitch, while the previous error was computed over part of the spectrum (over most harmonics together) for each pitch candidate. The harmonic error terms $e(l)$ are defined as

$$e(l) = \frac{\sum_{m=lower}^{upper} [S(m) - S_{\omega_o}(m)]^2}{\sum_{m=lower}^{upper} S(m)^2} \tag{17}$$

$$upper = \left\lceil \frac{512(l+\frac{1}{2})\omega_o}{2\pi} \right\rceil \tag{18a}$$

$$lower = \left\lceil \frac{512(l-\frac{1}{2})\omega_o}{2\pi} \right\rceil \tag{18b}$$

The threshold function is computed on a per-frame basis. The bounds on $T(\omega_o,e(l),L_o)$ are shown graphically in Figure 16 above for a single frame of speech. The points A and D show two possible extreme initial values of the threshold as computed from the pitch. Segment AB shows a typical initial threshold limit for a high pitched speaker. Segment DE shows a typical initial threshold limit for a low pitched speaker. The important point is that the initial threshold is highest for a high pitched speaker, making voiced decisions easier. The initial value (points A and D, for example) of the threshold varies linearly with the pitch and is based on the typical observed quality of the match between the synthetic and actual spectra.

The initial flat portion of $T(\omega_o,e(l),L_o)$ shown in the figure above as segments AB and DE is given by the equation below. The precision with which the constants are represented in these equations is not indicative of their actual accuracy.

$$T_f = \frac{0.15735 + \omega_o\left(\frac{8000}{2\pi}\right)(0.0014245)}{1 + \frac{\sum_{l=1}^{L_o} e(l)}{1.1L_o}} \tag{19}$$

The numerator is linearly dependent on the pitch and sets the starting threshold. The higher the pitch, the higher the initial threshold is set. The denominator can be thought of as adjusting the threshold based on a signal to noise ratio since it is based on the average error between the synthetic and original spectra. A high average error, indicating a poor match (a lower likelihood that the harmonic is voiced), will result in the threshold being lowered thus making it more difficult to declare harmonics voiced.

The breakpoints B and E are computed along the harmonic frequency axis based on the per harmonic error terms. The breakpoint represents the harmonic frequency where the spectral match begins to decay. The breakpoint is established based on the third observation above. The breakpoint occurs at the lowest harmonic frequency where $e(l)$ exceeds the average error. This condition can be written as

$$e(l) > \frac{\sum_{l=1}^{L_o} e(l)}{L_o} \tag{20}$$

Because this breakpoint is allowed to change based on the match between the synthetic voiced spectrum and the original spectrum, the slope of BC (or EF) must be computed for each frame of speech.

From the breakpoint upward into the spectrum, the threshold $T(\omega_o, e(l), L_o)$ is sloped down to its ending value as given below.

$$T_{end} = T_f[0.62059 - 0.008824L_o] \tag{21}$$

The slope of BC (or EF) is based on the ending value of the threshold along with the initial threshold value and the location of the breakpoint.

If the current spectrum is tilted such that the energy in the upper half of the spectrum is at least twice the energy in the lower half of the spectrum there is no need to complete the thresholding process (or the calculation of the harmonic error terms, for that matter). Based on the last observation listed above, the entire frame is arbitrarily declared unvoiced. This, in effect, is a declaration that the pitch is probably invalid and that any voicing decision would be suspect. The energy condition is shown below.

$$\frac{\sum_{m=129}^{255} [S(m)]^2}{\sum_{m=1}^{128} [S(m)]^2} > 2.0 \tag{22}$$

Finally, if the energy condition above is not true, each harmonic error term is compared with the corresponding threshold value. Any harmonic whose error is below the threshold is declared voiced. Any harmonic above the threshold is declared unvoiced.

The harmonic based decisions are grouped into bands. In keeping with the spectral representation chosen for this MBE-based coder, the first eleven bands contain a maximum of three harmonics each. The last band contains all of the remaining harmonics.

For each of the first eleven bands, if two or more harmonics (a majority) are voiced, that entire band is declared voiced. Since the last band may contain more than three harmonics, at least half of the harmonics in the last band must be voiced for it to be declared voiced.

Once the set of voicing decisions has been formed, a final confidence test is performed. Despite all the efforts to produce a good pitch estimate, the possibility exists that the current voicing decisions have been formed based upon a poor pitch estimate. If this is the case, the decisions may be perfectly proper based on the bad pitch but totally invalid for the unknown correct pitch. As a final check on the V/UV decision, the first two bands are tested to see if either band was declared voiced. As noted in the list of observations above, voiced speech tends to be strongly voiced at the low end of the spectrum. Based on this observation, *at least one* of the first two bands must be declared voiced if *any* other bands are to be declared voiced. Otherwise, the entire frame is arbitrarily declared unvoiced.

Spectrum Model and Gain

The vocal tract response is modeled by estimating the smoothed speech spectrum for the frame. There are several possibilities for representing the spectrum. The initial method for representing the spectrum was to compute a 512-point DFT of the current analysis frame. Then a peak picking algorithm was applied to the computed spectrum to determine the locations of the harmonics. The entire spectrum need not be stored, however, only the amplitudes which correspond to harmonics of the pitch. For this case, the refined pitch is used to select the spectral amplitude for each harmonic. The spectral amplitude for each harmonic (voiced or unvoiced) was extracted directly from the DFT spectrum. Based on a minimum possible pitch of about 65 Hz, there could be at most 60 harmonic amplitudes. These amplitudes were used to scale the voiced or unvoiced reconstruction which will be performed by the synthesizer.

The method described above would require extensive coding techniques, since, the maximum number of harmonics is 60. This prompted the need for an alternate spectral model. The spectral amplitudes can also be represented using a linear prediction model. Using a linear prediction will result in a smoother representation for the spectrum than using the spectral amplitudes directly. This will result in a loss of frequency resolution in the spectral representation, however, fewer coefficients are required to represent the spectrum.

For this design, a tenth order LPC (Linear Prediction Coefficients) model is used to represent the spectrum on a frame-by-frame basis. The LPC coefficients are computed using the autocorrelation method for the current analysis frame, $s(n)$, which was Hamming windowed [31]. The tenth order model is generally sufficient to resolve the 3-4 formants, but is too smooth to resolve much else. Figures 17 and 18 show the smooth LPC spectrum matched with the spectrum of a frame of voiced speech and a frame of unvoiced speech.

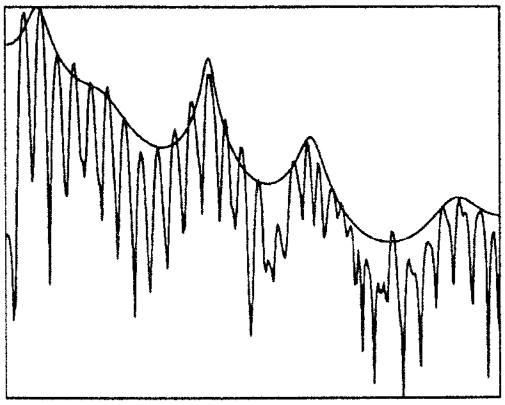

Figure 17. A Match of the LPC Spectrum with the Original Spectrum (mostly voiced).

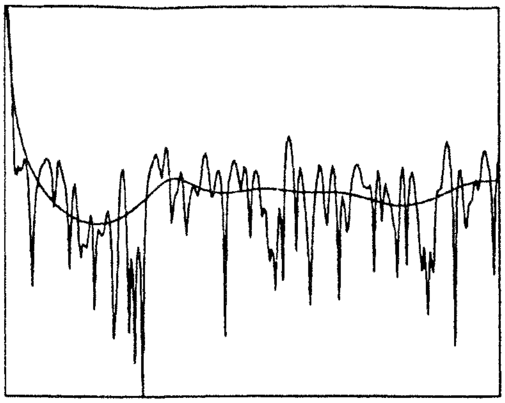

Figure 18. A Match of the LPC Spectrum with the Original Spectrum (mostly unvoiced).

The LPC gain is computed from the DFT magnitude spectrum and the LPC magnitude spectrum for each frame. The gain is computed as shown below in equation (23). $S_{LPC}(m)$ is the LPC magnitude spectrum as defined below in equation (24) which is computed from the LPC coefficients as shown in equation (25).

$$G = \left[\frac{\sum_{l=1}^{L_o} S(\lfloor \frac{512\omega_o l}{2\pi} + 0.5 \rfloor)^2}{\sum_{l=1}^{L_o} S_{LPC}(\lfloor \frac{512\omega_o l}{2\pi} + 0.5 \rfloor)^2} \right]^{\frac{1}{2}} \tag{23}$$

$$S_{LPC}(m) = \left| \frac{1}{\sum_{n=0}^{511} \hat{\alpha}_n e^{-j\frac{2\pi mn}{512}}} \right| \qquad 0 \le m < 512 \tag{24}$$

$$\hat{\alpha}_n = \{1, -\alpha_1, -\alpha_2, \ldots\ldots, -\alpha_{10}, 0, 0, \ldots\ldots 0\} \tag{25}$$

Synthesizer

In the receiver, the reconstructed signal is formed by synthesizing the voiced and unvoiced parts separately and summing the result. A block diagram of the receiver is shown in Figure 19 below.

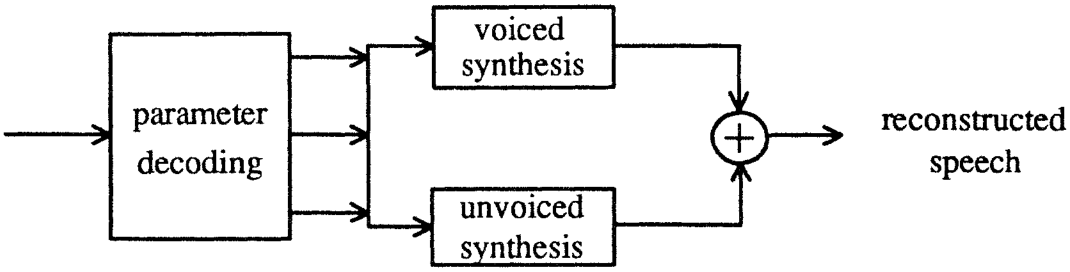

Figure 19. Block Diagram of the Synthesizer.

The synthesizer receives a set of parameters describing each analysis frame. These parameters include pitch, V/UV decisions, LPC parameters, and an associated LPC gain value. These parameters, encoded in the transmitter, are similarly decoded at the receiver prior to use. A synthesis overview is provided in the next section followed by a discussion of the unvoiced synthesis and voiced synthesis.

Overview of Synthesis

As shown in Figure 19, synthesis is divided into two sections: unvoiced and voiced reconstruction. The two time domain components are computed separately, multiplied by a reconstruction window, and summed to produce a 20 ms frame of synthesized speech.

The original method of reconstruction was performed using the INMARSAT IMBE reconstruction [28]. Concern about proprietary issues led to the development of a new method of reconstruction. This method is described in the following sections.

Certain parameters need to be introduced at this point in order to completely understand the voiced and unvoiced synthesis. The reconstruction window is the 229 point tapered overlapping window shown in Figure 20. The current frame of speech is centered on the taper, so the current frame includes a portion of the previous frame and a portion of the new frame. A more in-depth discussion of this window follows along with the discussion of the unvoiced and voiced synthesis procedures.

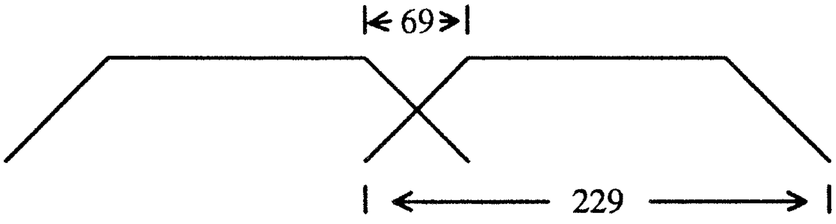

Figure 20. Two Overlapped Reconstruction Windows.

The band structure for the receiver has a maximum of K bands, where $K = 12$, with a maximum of three harmonics in the first eleven bands with any remaining harmonics coded in the last band. The number of harmonics in the current frame is defined by equation (9) in the transmitter. The receiver determines the number of harmonics and bands in the current frame directly from the received pitch value.

The voiced/unvoiced (V/UV) decision for each band is used to determine whether a particular band is to be reconstructed as either voiced or unvoiced. Each band is processed separately. If the current band is declared unvoiced, a noise signal having the appropriate spectral characteristics (bandwidth and amplitude) is generated in the frequency domain. The amplitude characteristics for the current unvoiced band are determined by computing the reciprocal of the DFT of the linear prediction coefficients (LPC) for the current frame weighted by the LPC gain. The bandwidth is determined by the pitch and the number of harmonics in the current band. The magnitude characteristic for the band is then assigned a random phase. This process is repeated for every unvoiced band in the frame.

The time domain waveform corresponding to the unvoiced bands in the current frame is generated by taking the inverse DFT (IDFT) of the unvoiced spectrum just formed. This waveform is saved as the unvoiced component for the current frame.

For each voiced band in the current frame, a sinusoidal oscillator is used to generate a periodic signal corresponding to the appropriate harmonic frequencies in the band. These harmonics are scaled by the appropriate harmonic amplitudes and summed. Each voiced band in the frame is processed similarly. The result is a time domain signal corresponding to the voiced components in the current frame.

After the time domain signals corresponding to the unvoiced and voiced components for the current frame are formed, the signals are summed, producing one complete frame of reconstructed speech. Adjacent frames are overlapped to maintain smoothness in the reconstructed signal.

A technique which is referred to as Linear Frequency Variation is applied to the voiced components to linearly interpolate the pitch and its harmonics across frames and to maintain phase continuity between corresponding harmonics at frame boundaries [26]. A detailed discussion of the algorithm used for reconstruction follows.

Unvoiced Synthesis

The unvoiced synthesis algorithm is used to generate a time domain sequence $u_0(n)$ that contains the unvoiced component of the current frame of reconstructed speech. An initial noise sequence for the unvoiced synthesis is generated directly in the frequency domain instead of the time domain.

The DFT is defined per equation (26) for notation purposes. This discussion assumes that all time domain sequences have been properly zero padded prior to computation of the DFT.

$$S(m) = DFT_{512}[s(n)] = \sum_{n=0}^{511} s(n) e^{-j\frac{2\pi mn}{512}} \qquad (26)$$

The LPC spectrum or smoothed vocal tract response, $\hat{S}_{LPC}(m)$, is computed by taking the reciprocal of the DFT of the LPC coefficients as shown in equation (27), where again proper zero padding is assumed. G is the LPC gain from the analyzer.

$$\hat{S}_{LPC}(m) = \frac{G}{\sum_{n=0}^{511} \hat{\alpha}_n e^{-j\frac{2\pi mn}{512}}} \qquad 0 \le m < 512 \qquad (27)$$

where

$$\hat{\alpha}_n = \{1, -\alpha_1, -\alpha_2, \ldots\ldots, -\alpha_{10}, 0, 0, \ldots\ldots 0\} \qquad 0 \le n < 512$$

The initial noise spectrum, $N_0(m)$, is then computed by assigning a random phase, $\theta(m)$, to the LPC magnitude spectrum. The random phase is simply a uniformly distributed random variable on the interval $[0,2\pi)$. Equation (28) shows this operation in terms of magnitude and phase. Equations (29) and (30) show the equivalent operation using real and imaginary parts. Further, the spectrum $N_0(m)$ must exhibit even symmetry for the real part and odd symmetry for the imaginary part if the time domain noise signal is to be real. This is not shown explicitly in the equations.

$$N_o(m) = \left|\hat{S}_{LPC}(m)\right| e^{j\theta(m)} \qquad 0 \leq m < 512 \qquad (28)$$

$$Re[N_o(m)] = \left|\hat{S}_{LPC}(m)\right| \cos(\theta(m)) \qquad 0 \leq m < 512 \qquad (29)$$

$$Im[N_o(m)] = \left|\hat{S}_{LPC}(m)\right| \sin(\theta(m)) \qquad 0 \leq m < 512 \qquad (30)$$

After the initial noise spectrum, $N_0(m)$, is generated, the unvoiced speech is reconstructed for each harmonic. The upper and lower bounds for the l^{th} unvoiced harmonic are computed as shown in equations (31) and (32).

$$lower = \left\lceil \left(\tfrac{512}{2\pi}\right)(l - 0.5)\omega_o \right\rceil \qquad (31)$$

$$upper = \left\lceil \left(\tfrac{512}{2\pi}\right)(l + 0.5)\omega_o \right\rceil \qquad (32)$$

If the current band k, defined in equation (36), is declared voiced then assignment is made per equation (33) for each harmonic in the band. But, if the current band k is declared unvoiced, then assignment is made per equations (34) and (35) for each harmonic in the band. As before with $N_0(m)$, the symmetry of $U_0(m)$ must be enforced as the spectrum is being built.

$$U_o(m) = 0 \qquad lower \leq m < upper \qquad (33)$$

$$Re[U_o(m)] = Re[N_o(m)] \qquad lower \leq m < upper \qquad (34)$$

$$Im[U_o(m)] = Im[N_o(m)] \qquad lower \leq m < upper \qquad (35)$$

$$k = \left\lfloor \tfrac{l+2}{3} \right\rfloor \qquad (36)$$

After the noise spectrum $U_o(m)$ is built, then the low and high frequency terms are set to zero using equations (37) and (38). The low and high frequency band edges are computed using equations (39) and (40), respectively. The low and high frequency terms are zeroed because the spectrum is not modeled below half of the fundamental frequency or above the last harmonic plus half of the fundamental frequency. Again, the symmetry of the spectrum must be maintained.

$$U_o(m) = 0 \qquad 0 \leq m < left \qquad (37)$$

$$U_o(m) = 0 \qquad right \leq m < 256 \qquad (38)$$

$$left = \left\lceil \left(\tfrac{512}{2\pi}\right)(1 - 0.5)\omega_o \right\rceil \qquad (39)$$

$$right = \left\lceil \left(\tfrac{512}{2\pi}\right)(L_o + 0.5)\omega_o \right\rceil \qquad (40)$$

Next, to get $u(n)$, the actual time domain unvoiced component for the current frame, the IDFT of $U_o(m)$ must be computed. This is easily accomplished by taking the complex conjugate of $U_o(m)$ and then computing the DFT of the result. This is shown in equation (41).

$$u(n) = \tfrac{1}{512} \sum_{m=0}^{511} U_o^*(m) e^{-j\frac{2\pi mn}{512}} \qquad 0 \leq n < 512 \qquad (41)$$

Using the unvoiced reconstructed data, $u_{-1}(n)$, from the previous frame, the data, $u(n)$, from the current frame, and the overlapped tapered reconstruction window, $u_o(n)$ can now be computed as shown in equation (42).

$$u_o(n) = u_{-1}(n)w_d(n) + u(n)w_u(n) \qquad 0 \leq n < 160 \qquad (42)$$

A frame of unvoiced speech generated from equation (42) is presented in Figure 21. The reconstruction windows, $w_d(n)$ and $w_u(n)$, are shown in Figure 23 and discussed in detail in a later section.

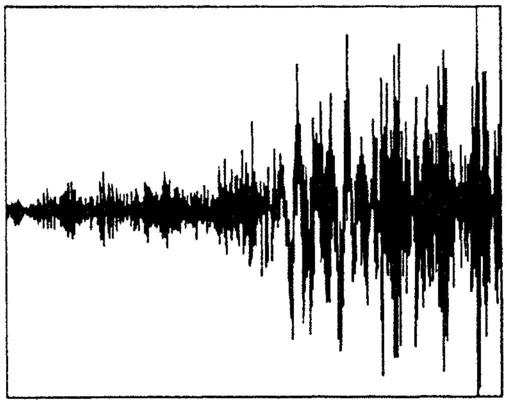

Figure 21. A Frame of Reconstructed Unvoiced Speech.

Finally, the second 160 points of the current unvoiced signal is saved to become $u_{-1}(n)$ for the next frame.

$$u_{-1}(n) = u(160 + n) \qquad 0 \leq n < 160 \qquad (43)$$

Voiced Synthesis

The voiced synthesis algorithm is used to generate a time domain sequence, $v_o(n)$, that contains the current frame of reconstructed voiced speech data. The time domain sequence is generated using equation (44). The sequence $v_l(n)$ is the l^{th} sinusoid with harmonic frequency $l\omega_o$.

$$v_o(n) = \sum_{l=1}^{\max(L_{-1},L_o)} v_l(n) \qquad 0 \le n < 160 \qquad (44)$$

The upper limit on the summation is the maximum of the number of harmonics in the previous frame and the number of harmonics in the current frame, because the number of harmonics between adjacent frames is not always the same.

To maintain smoothness (smoothly varying pitch and phase continuity) of the harmonics from frame to frame, the pitch is continuously interpolated across frames and phase continuity is enforced at frame boundaries. Smoothness across frame boundaries is critical to producing high-quality reconstructed speech. The method of ensuring phase continuity and smoothly varying pitch is referred to as Linear Frequency Variation (LFV) [26]. The derivation of LFV is presented in Appendix A.

The voiced reconstruction procedure varies depending on the V/UV decisions in the previous and current frames. A number of possibilities exists when reconstructing the voiced signal. These possibilities are listed below. A discussion of each instance follows.

i) Previous band is voiced and current band is voiced.

  a) $L_o > L_{-1}$
  b) $L_o < L_{-1}$
  c) $L_o = L_{-1}$

ii) Previous band is voiced and current band is unvoiced.

iii) Previous band is unvoiced and current band is voiced.

iv) Previous band is unvoiced and current band is unvoiced.

Case 1, where the previous band and the current band are voiced, is broken into three sub-cases as shown above. When $L_o > L_{-1}$, there are more harmonics in the current frame than in the previous frame. The goal is to smoothly connect corresponding harmonics from the previous frame to the current frame. Since the number of harmonics in each frame is not the same, additional harmonics must be *born* in the current frame. This condition occurs because the current pitch frequency is sufficiently lower than the previous pitch frequency to produce more harmonics in the current frame.

First, a uniformly distributed random phase $\theta(l)$ is generated between $[0,2\pi)$. Note that this is not the $\theta(m)$ which was defined and used for unvoiced reconstruction. Then $v_l(n)$ can be computed from equation (45).

$$v_l(n) = 2w_u(n)A_o(l)\cos[\omega_o nl + \theta(l)] \qquad \begin{array}{l} 0 \le n < 160 \\ 0 \le \theta(l) < 2\pi \end{array} \qquad (45)$$

Once $v_l(n)$ has been calculated, a new phase value, $\theta(l)$, must be determined in order to keep track of the phase for the l^{th} harmonic across the frame boundary. This is shown in equation (46).

$$\theta(l) = 160\omega_o l + \theta_{-1}(l) \qquad (46)$$

When $L_0 < L_{-1}$, there are more harmonics in the previous frame than in the current frame. This condition occurs when the current pitch frequency is sufficiently higher than the previous pitch frequency so that fewer harmonics are present in the current frame. Harmonics in the previous frame for which there are no corresponding harmonics in the current frame must be allowed to *die* in the current frame. The calculation of $v_l(n)$ is shown in equation (47) where $\theta(l)$ is defined in equation (46).

$$v_l(n) = 2w_d(n)A_{-1}(l)\cos[\omega_{-1}nl + \theta(l)] \qquad 0 \le n < 160 \qquad (47)$$

The final instance for case 1 is when $L_0 = L_{-1}$. For this case each harmonic in the current frame has a corresponding harmonic in the previous frame. The calculation of $v_l(n)$ is shown below.

$$v_l(n) = 2\cos[\omega_{-1}nl + \phi(n) + \theta(l)](w_d(n)A_{-1}(l) + w_u(n)A_o(l)) \qquad 0 \le n < 160 \qquad (48)$$

where

$$\phi(n) = \frac{(\omega_o - \omega_{-1})n(n+1)l}{2(159)} \qquad 0 \le n < 160 \qquad (49)$$

and, again, $\theta(l)$ must be determined, as in equation (50), in order to keep track of the phase across frame boundaries.

$$\theta(l) = 160\omega_{-1}l + \phi(n) + \theta_{-1}(l) \qquad (50)$$

In Case 2 the previous band was declared voiced and the current band is declared unvoiced. The harmonics in the previous voiced band are allowed to die off, becoming unvoiced in the current frame, as shown in equation (51). (Note: the convention is to use the minimum possible value of pitch, 65 Hz, for any frame that is purely unvoiced. This results in unvoiced frames having the maximum possible number of frequency bands.)

$$v_l(n) = 2w_d(n)A_{-1}(l)\cos[\omega_{-1}nl + \theta(l)] \qquad 0 \le n < 160 \tag{51}$$

Any unvoiced bands in the current frame which do not have corresponding voiced harmonics in the previous frame (the case where there are more harmonics in the current frame than in the previous frame) are represented by equation (52).

$$v_l(n) = 0 \qquad 0 \le n < 160 \tag{52}$$

In Case 3 the previous band was declared unvoiced and the current band is declared voiced. New voiced harmonics are born using equation (53). Each new harmonic starts with a random phase θ(l), uniformly distributed on the interval [0,2π), as shown in equation (53). The harmonic phase, θ(l), must be re-calculated in each frame in order to keep track of the phase across frame boundaries. This calculation is given in equation (54).

$$v_l(n) = 2w_u(n)A_o(l)\cos[\omega_o nl + \theta(l)] \qquad 0 \le n < 160 \tag{53}$$

$$\theta(l) = 160\omega_o l + \theta_{-1}(l) \tag{54}$$

Any unvoiced bands in the previous frame which do not have corresponding voiced harmonics in the current frame are reconstructed, as before, using equation (52).

In case 4 the previous band is unvoiced and the current band is unvoiced. For this condition, the voiced segment $v_l(n)$ is identically zero and the band is synthesized as described in the unvoiced algorithm.

Once all the voiced harmonics have been built for the current frame, they are summed to form the complete voiced component for the current frame using equation (44). This process is repeated for subsequent frames.

A frame of voiced speech that corresponds to the unvoiced frame presented in Figure 21 is presented in Figure 22. The sum of the voiced and unvoiced parts are shown in Figure 23. The sum of the voiced and unvoiced speech data can be compared to the original time domain speech waveform shown in Figure 24. Notice the difference in phase of the waveforms in Figures 23 and 24, this result is expected because no information about the phase is being transmitted to the receiver.

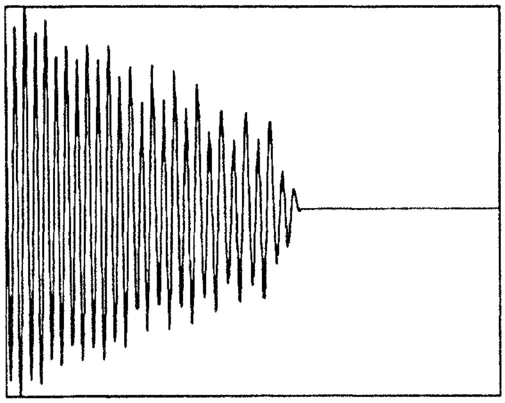

Figure 22. A Frame of Reconstructed Voiced Speech.

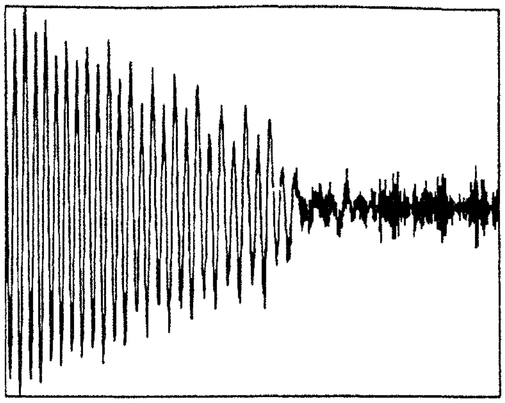

Figure 23. The Sum of the Reconstructed Voiced and Unvoiced Parts.

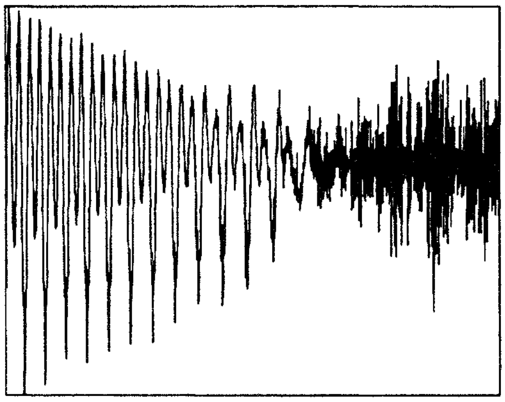

Figure 24. A Frame of the Original Time Domain Speech Waveform.

Overlapping Window

The window used during the reconstruction process is an overlapping tapered window $w(n)$, shown in Figure 20. The specifications of the window $w(n)$ are given in Figure 25.

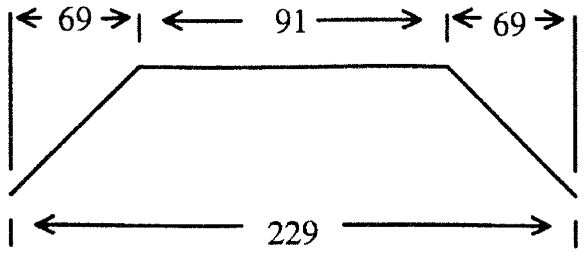

Figure 25. Specifications for the Reconstruction Window, *w(n)*.

The current frame of reconstructed speech includes the taper of the window but is not centered on the taper because the taper has an odd number of points. The taper and overlap are chosen to produce 160 points of windowed speech data, giving the same frame rate as used in the analyzer. The current frame is shown in Figure 26.

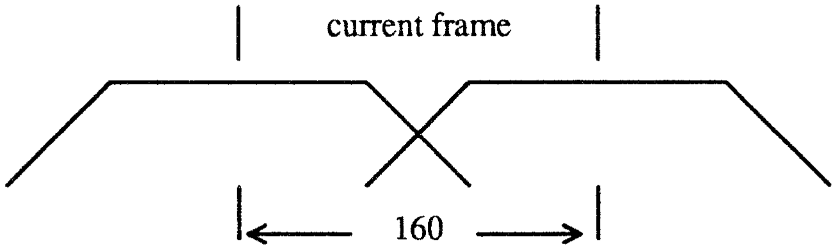

Figure 26. The Current Frame of Reconstructed Speech (160 points).

Since the current frame includes the taper, the window $w(n)$ can be separated into two sections, $w_d(n)$ and $w_u(n)$. The window $w_d(n)$ is used to let the harmonics *die* in the previous frame. This is referred to as the *down window*. The other window $w_u(n)$ is used when harmonics are *born* in the current frame. This is referred to as the *up window*. These two windows and how they overlap to give 160 points of windowed data are shown in Figure 27.

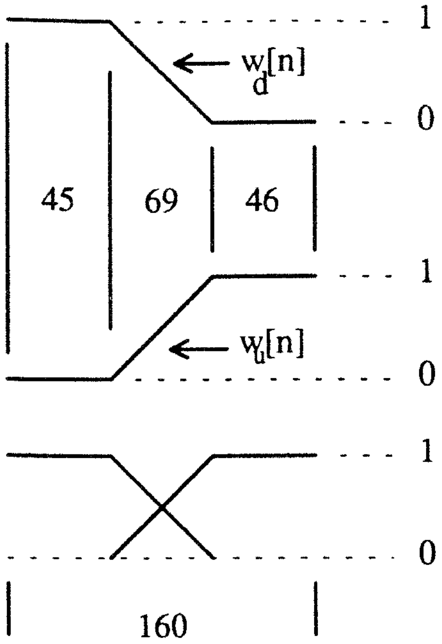

Figure 27. Diagram of $w_d(n)$ and $w_u(n)$ and Detail of the Overlap.

The taper of the window can be calculated using equations (55) and (56). The actual values used in this algorithm are listed in Appendix B.

$$w_d(n) = \begin{cases} 1 & 0 \le n \le 44 \\ 1 - \frac{1}{68}(n-45) & 45 \le n \le 113 \\ 0 & 114 \le n \le 159 \end{cases} \qquad (55)$$

$$w_u(n) = \begin{cases} 0 & 0 \le n \le 44 \\ \frac{1}{68}(n-45) & 45 \le n \le 113 \\ 1 & 114 \le n \le 159 \end{cases} \qquad (56)$$

Quantizer

After the parameters have been estimated some form of quantization must be performed. The bit rate goal for this study is 2,400 bits per second. The Enhanced MBE vocoder assumes a sampling frequency of 8,000 Hz. and operates on 160 samples per frame (20 ms). This results in 50 frames per second which allows for 48 bits per frame to represent the input data. This data consists of pitch, voicing, spectrum representation, and gain. The pitch is quantized with 8 bits using an optimum quantizer. The voicing decisions are compressed into 10 bits. The gain is compressed into 5 bits. The LPC spectrum is vector quantized using a split-2 25 bit vector quantizer.

Pitch

The pitch is quantized, using Lloyd-Max quantization, into 8 bits [29]. Thousands of frames of pitch data were computed and then Lloyd-Max was used to optimize the data into 256 possible values. These values are stored in a codebook.

In the transmitter, the estimated pitch is compared to each value in the codebook. The index with the smallest least square error is transmitted to the receiver. The receiver then uses the index to find the quantized pitch value in the codebook.

Voiced/Unvoiced Decisions

The voiced/unvoiced decisions are determined for a maximum of twelve bands. In order to get to 2,400 bits per second the voiced/unvoiced decisions need to be coded using 10 bits. By observation, the voicing decisions in the upper bands (9-12) tended not to contain a set pattern. This is especially true for the last band, where the number of harmonics range from 0 to 27. When tested using only the first ten bands, no considerable difference could be heard. So, the first ten bands are transmitted to the receiver and the upper two bands are never transmitted.

Gain

The gain, which is the relationship between the amount of energy in the original DFT spectrum and the smoothed LPC spectrum, is quantized into 5 bits. The gain is quantized using logarithmic companding as shown in the equation below.

$$\log[1+G] * \frac{2^B}{\log[G_{\max}]} \tag{57}$$

where G is the gain, B is the number of bits, and $G_{\max}$ is the maximum value of the gain. This result is transmitted to the receiver and the formula is applied in reverse to attain a quantized gain value.

LPC Coefficients

The LPC spectrum is quantized using vector quantization. The vector quantizer produces a 12 bit upper codebook index and a 13 bit lower codebook index, which results in a total of 25 bits for representing the spectrum.

The LPC spectrum representation is converted to line spectral frequencies to match the codebook. The codebook is searched for the best match using a simple squared error criterion on the line spectral frequencies. The codebook can be searched using a full search or a tree search. The full search computes an error value for each entry in the codebook. The tree search computes an error value and based on that error value propagates down the tree until the best match is found. The tree search does not find the most optimum match only the best match for that segment of the tree.

The 13-bit index codebook is used to represent the lower four values of the line spectral frequencies. The values of the codebook are stored in an array of size 8,192 by 4.

The 12-bit index codebook is used to represent the upper six values of the line spectral frequencies. The values of the codebook are stored in an array of size 4,096 by 6.

A best match is found for each codebook independently. This is shown in the Figure 28 below. The two indexes together represent the line spectral frequencies corresponding to the LPC coefficients for a specific frame.

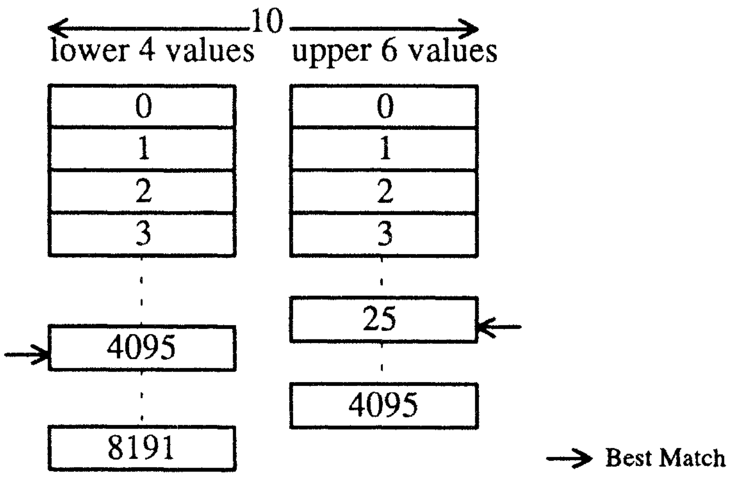

Figure 28. Diagram of Split-2 25 bit Vector Quantizer Codebook.

After, a best match is found the indexes are transmitted to the receiver. The receiver uses the same codebooks, that the transmitter used, to lookup the quantized line spectral frequencies. The receiver then converts the line spectral frequencies back to LPC coefficients.

CHAPTER IV

RESULTS AND CONCLUSIONS

Summary

This study was the design and development of a 2,400 bit per second Enhanced MultiBand Excitation vocoder. The design is based on the MBE speech model developed by Daniel Griffin and Jae Lim. Enhancements were made to the parameter estimation, with particular attention being paid to pitch and voiced/unvoiced decisions in the transmitter and reconstruction in the receiver. Overlapping frames and windows were used to help smooth the parameter estimates and the reconstructed speech. These enhancements represent improvements to the MBE speech model and helped make implementing Enhanced MBE at 2,400 bits per second possible.

This design used a three step pitch refinement procedure, using the AMPEX algorithm for the initial pitch estimate, to produce sub-sample accuracy pitch estimates. The voiced/unvoiced decision algorithm presented was a new band-based method using a variable threshold technique. A tenth order linear predictor was used to represent the vocal tract response. Included in this design was a new method for reconstructing speech samples. The reconstructed speech was synthesized using a continuous pitch interpolation technique to smoothly vary the pitch across and within frames.

The base model was implemented with no quantization to ensure that the model alone was capable of producing high quality reconstructed speech. Next, quantization was added to lower the bit rate to 2,400 bits per second. This was accomplished using the following methods. The pitch was quantized to 8 bits using a Lloyd-Max quantization technique. Ten bits were used to represent the voiced/unvoiced decisions. The linear prediction coefficients were quantized using a split-2 25 bit vector quantizer. The gain was quantized to 5 bits using logarithmic companding. This totaled 48 bits per frame and resulted in a frame rate of 50 frames per second. The following sections will discuss the results and conclusions of this implementation.

Results

As presented in the summary, a number of enhancements were made to improve on the MBE speech model. The results of these enhancements are presented.

The three step pitch refinement procedure returned sub-sample pitch accuracy provided that AMPEX made correct initial estimates. The voiced/unvoiced decision algorithm proved to be a robust voicing algorithm, by making correct V/UV decisions in both quiet and noisy enviroments. The reconstruction algorithm provided smooth synthesized speech samples, i.e., no audible chirps. As a result of quantization, there were only a few audible artifacts.

Preliminary results obtained from extensive listening of the Enhanced MBE vocoder have been encouraging. The subjective quality of the reconstructed speech is reasonably good, even when a relatively low order LPC model is used to represent the spectrum. Both intelligibility and naturalness of the reconstructed speech are good.

The Enhanced MBE vocoder was submitted for testing against other 2,400 bit per second vocoders. Early results of this test have shown the Enhanced MBE vocoder to be competitive although at this time the final results have not been returned.

Conclusions

One major conclusion developed during the design is that the pitch estimate and V/UV decisions are crucial to the success of a vocoder such as one based on the MBE model. In this regard, a pitch estimate with sub-sample resolution is necessary. This is true because the model parameters represent only harmonics of the pitch, and, if the pitch is in error the ability to accurately locate higher harmonics in the spectrum becomes impossible. This is a result of the error being multiplicative at higher harmonics as the pitch estimate degrades. This error also affects the V/UV decisions, since voicing is dependent on accurately determining which harmonics in the spectrum are present and which are absent.

Future Research

A future goal is to implement the Enhanced MBE vocoder in a real time system. Implementing this design in real time will pose a new set of questions.

The first topic is, can AMPEX be implemented in real time? The implementation of AMPEX in this study is complex and can probably not be implemented in real time.

This opens the door for a new pitch detector. A method using autocorrelation will probably work because of the extra pitch refinement included in this design.

Another topic related to real time is the searching of the vector quantizer codebooks for the best match. The codebooks tend to be large in size so a full search method is probably not desirable. Therefore, a tree search method will be needed and is included in this design. There are other questions concerning the tree search method such as: tree size and comparison method. The tree size can range from 1 bit to 7 bits, in this design. A 1 bit tree is approximately equal to searching half of the codebook and a 7 bit tree only searches 1/128th of the codebook. The 7 bit tree is much faster but the result is not as reliable as the 1 bit tree. The comparison methods are a squared error value and a cepstral (perceptual) match. Extensive listening needs to be performed on both methods to determine which is more applicable.

Other topics of concern are LPC model, framing, and voiced/unvoiced band structure. The LPC model being used is tenth order. Since LPC provides a smoothed version of the spectrum, using a higher order model such as 12 or 14 should provide a better spectral model. The current tenth order LPC model uses a split-2 25 bit codebook. A higher order LPC model will require a larger codebook resulting in more bits needed to represent the spectrum accurately.

The framing, 20ms, used in this design is typical of other MBE based vocoders. Under consideration is the use of 30ms superframes (240 samples). A superframe would contain three subframes (80 samples each). This method would transmit the spectral model once per superframe. The spectral model could then be updated on a subframe by using the fact that line spectral frequencies can be linearly interpolated to form an updated spectral model for each subframe. Speech model parameters such as pitch, V/UV decisions and gain could be updated during each subframe instead of just sending the parameters once every 20ms. This would allow the parameters to be updated more often and still retain the 2,400 bit per second implementation.

The V/UV band structure for this design follows the structure presented in the INMARSAT IMBE standard. A couple of methods being considered are non-linear bands and using more than three harmonics per band in the current implementation. Non-linear bands would use fewer harmonics per band for the low end of the spectrum and then increase the number of harmonics per band at the higher end of the spectrum. The other method would allow more harmonics per band linearly throughout the spectrum. The non-linear band structure would probably be the preferred method. Since the ear is non-linear, errors are more audible at lower frequencies than at higher frequencies. Perceptually the non-linear band structure may be better because the lower frequencies will be emphasized more and the higher frequencies will be deemphasized.

REFERENCES

[1] D.W. Griffin and J.S. Lim, "Multiband Excitation Vocoder," *IEEE Trans. ASSP*, Vol. 36, No. 8, August 1988.

[2] Allen Gersho, "Advances in Speech and Audio Compression," *Proceedings of the IEEE*, Vol. 82, No. 6, June 1994.

[3] Lawrence R. Rabiner, "Applications of Voice Processing to Telecommunications," *Proceedings of the IEEE*, Vol. 82, No. 2, February 1994.

[4] Keith Teague, Bryce Leach, and Walter Andrews, "Development of a High-Quality MBE based Vocoder for Implementation at 2400 bps," *Proceedings IEEE Wichita Conference on Communications, Networking, and Signal Processing*, April 1994.

[5] Luc M. Van Immerseel and Jean-Pierre Martens, "Pitch and Voiced/Unvoiced Determination with an Auditory Model," *J. Acoustics. Soc. Am.* 91 (6), June 1992.

[6] Dik J. Hermes, "Pitch Analysis," in *Visual Representation of Speech Signals*, ed: M. Cooke, et al, Wiley, 1993.

[7] R. Linggard, "*Electronic Synthesis of Speech*," Cambrigde University Press, Cambridge, 1985.

[8] F.J. Owens, "*Signal Processing of Speech*," McGraw Hill, New York, 1993.

[9] Joe P. Campbell and Richard A. Dean, "A History of Voice Coding," *Digital Signal Processing*, Vol. 3, 1993.

[10] John R. Deller, John G. Proakis, and John H.L. Hansen, "*Discrete-Time Processing of Speech Signals*," Macmillan Publishing Company, New York, 1993.

[11] Douglas O'Shaughnessy, "*Speech Communication: Human and Machine*," Addison-Wesley, New York, 1987.

[12] L.R. Rabiner and R.W. Schafer, "*Digital Processing of Speech Signals*," Prentice Hall, New Jersey, 1978.

[13] T. Tremain, "The Government Standard Linear Predictive Coding Algorithm: LPC-10," *Speech Technology Magazine*, April 1982.

[14] Manfred R. Schroeder and Bishnu S. Atal, "Code-Excited Linear Prediction (CELP): High Quality Speech At Very Low Bit Rates," *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing*, March 1985.

[15] J.P. Campbell, T.E. Tremain, and V.C. Welch, "The DOD 4.8 KBPS Standard (Proposed Federal Standard 1016)," *Advances in Speech Coding*, B.S. Atal, V. Cupperman, and A. Gersho, Eds. Norwell, MA: Kluwer, 1991.

[16] R.A. Salami, "Binary Code Excited Linear Prediction (BCELP): new approach to CELP coding of speech without codebooks," *Electron. Lett.*, vol. 25, no. 6, pp. 401-403, March 1989.

[17] I.A. Gerson and M.A. Jasiuk, "Vector Sum Excited Linear Prediction," *Advances in Speech Coding*, B.S. Atal, V. Cupperman, and A. Gersho, Eds. Norwell, MA: Kluwer, 1991.

[18] B. Fette, C. Bergstrom, C. Jaskie, S. You, R. Pattison, and C. Wood, "Pitch Synchronous Excited Linear Prediction (PSELP)," 2400 BPS Speech Workshop, February 1994.

[19] J-H. Chen, "A Robust Low-Delay CELP Speech Coder at 16 kb/s," *Advances in Speech Coding*, B.S. Atal, V. Cupperman, and A. Gersho, Eds. Dordrecht, The Netherlands: Kluwer, 1991.

[20] J.S. Baras and E. Butler, "Wavelets for Speech Processing," 2400 BPS Speech Workshop, February 1994.

[21] R.J. McAulay and T.F. Quatieri, "Speech Analysis/Synthesis Based on a Sinusoidal Representation," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-34, August 1986.

[22] Yair Shoham, "High-Quality Speech Coding at 2.4 to 4.0 KBPS Based on Time-Frequency Interpolation," *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing*, April 1993.

[23] M.S. Brandstein, P.A. Monta, J.C. Hardwick, and J.S. Lim, "A Real-Time Implementation of the Improved MBE Speech Coder," *Proceedings IEEE*

International Conference on Acoustics, Speech, and Signal Processing, April 1990.

[24] Bishnu S. Atal, Vladimir Cuperman, and Allen Gersho, "*Advances in Speech Coding*," Kluwer Academic Publishers, Boston, 1991.

[25] D.P. Kemp, R.S. Sueda, and Tom Tremain, "Evaluation of 4800 BPS Voice Coders,"

[26] Walter Andrews and Keith Teague, "High Quality Reconstruction of voiced Speech Using Linear Frequency Variation," submitted to *IEEE International Conference on Acoustics, Speech, and Signal Processing - 1995*, Detroit, Michigan, May 1995.

[27] Keith A. Teague, Walter Andrews, and Bryce Leach, "An Enhanced Procedure for Multiple V/UV Estimation," submitted to *IEEE International Conference on Acoustics, Speech, and Signal Processing - 1995*, Detroit, Michigan, May 1995.

[28] Inmarsat Satellite Communications Services, "Inmarsat-M System Definition, Issue 3.0-Module 1: System Description," November 1991.

[29] J. Max, "Quantizing for Minimum Distorttion," *IRE Transactions on Infromation Theory*, Vol. IT-6, pp7-12, March 1960.

[30] Y. Medan, E. Yair and D. Chazan, "Super Resolution Pitch Determination of Speech Signals," *IEEE Trans. Signal Processing*, Vol. 39, No. 1, January 1991.

[31] IEEE Press, "*Programs for Digital Signal Processing*," John Wiley and Sons, New York, 1979, pp.4.1-1 - 4.2-1.

[32] N.S. Jayant and Peter Noll, "*Digital Coding of Waveforms*," Prentice Hall, New Jersey, 1984.

[33] APCO, "NASTD Federal Project 25 Vocoder: Version 1.0," December 1992.

APPENDIX A

Linear Frequency Variation (LFV)

Linear Frequency Variation

This section presents a brief derivation of the pitch interpolation procedure which we refer to as Linear Frequency Variation (LFV) and which is used extensively in the voiced synthesis procedure.

In order to smooth two sinusoids of different frequency together and not produce any "chirps" the sinusoids must be smoothed linearly over the frame with continuity of phase maintained. To accomplish this we developed a technique based on frequency modulation (FM) as defined in equation (A-1). The lower limit of the integral is not specified, since to do so would require knowledge of an initial condition, which we will instead assume for now to be identically zero.

$$x_c(t) = A_c \cos\left[\omega_c t + 2\pi f_d \int^t m(\alpha)\,d\alpha\right] \qquad \text{(A-1)}$$

For the LFV algorithm, we have defined the output signal $x(n)$, as shown in equation (A-2). The gain value A, shown in equation (A-2), will be assumed to be unity.

$$x(n) = A\cos[\omega n + \phi(n) + \theta] \qquad \text{(A-2)}$$

This stems from the fact that in the implementation A will be a spectral amplitude. The $\phi(n)$ term is defined below in equation (A-3).

$$\phi(n) = \left(\frac{2\pi f_d}{f_s}\right)\sum_{i=0}^{n} m(i) \qquad 0 \le n < N \qquad \text{(A-3)}$$

The variable f_d is the *frequency deviation* and f_s is the sampling frequency. A closed form solution exists for $\phi(n)$. The closed form solution can be found by making a substitution for *m*(*n*) in equation (A-3). The resultant equation is shown in equation (A-4).

$$\phi(n) = \left(\frac{2\pi f_d}{f_s}\right) \sum_{i=0}^{n} \left(\frac{1}{N-1}\right) i \qquad 0 \leq n < N \qquad \text{(A-4)}$$

Now using the identity shown in equation (A-5) and substituting, $\phi(n)$ can be written as shown in equation (A-6).

$$\sum_{k=0}^{n} k = \frac{n(n+1)}{2} \qquad \text{(A-5)}$$

$$\phi(n) = \left(\frac{2\pi f_d}{f_s}\right)\left(\frac{n(n+1)}{2(N-1)}\right) \qquad 0 \leq n < N \qquad \text{(A-6)}$$

The message signal, *m*(*n*), is defined to be the straight line shown in Figure A-1. A straight line was chosen because our intent is to vary the pitch linearly from one sample to the next. Also, the line must be the same length as the analysis frame size; in this case the frame is specified to be *N* points long. The line can be computed using equation (A-7), shown below.

$$m(n) = \left(\frac{1}{159}\right) n \qquad 0 \leq n < N \qquad \text{(A-7)}$$

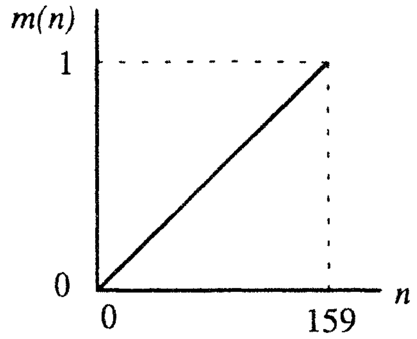

Figure A-1. Plot of the Message Signal, *m(n)*.

Now, the θ term in equation (A-2) can be specified to be the phase of the sinusoid at the n^{th} point (the initial condition at sample n). When a new harmonic is *born*, we set this value initially to be a uniformly distributed random number on $[0,2\pi)$.

To keep track of the phase at frame boundaries, θ is computed per equation (A-8). This equation gives the value of the phase N points beyond where it was last specified. The new phase term is calculated at the N^{th} point because we want to know where the start of the next frame must occur.

$$\theta = \omega N + \phi(N) + \theta_{-1} \qquad N = 160 \qquad \text{(A-8)}$$

A few modifications must be made to implement LFV for voiced reconstruction. In equation (A-6), f_d is defined to be the difference between the current pitch ω_o and the previous pitch ω_{-1}. Also, since the voiced synthesis is built on a harmonic-by-harmonic basis, $\phi(n)$ must be proportional to the l^{th} harmonic. The implemented version of this equation is shown in (A-9). This is shown in equation (49).

$$\phi(n) = \frac{(\omega_o - \omega_{-1})n(n+1)l}{2(N-1)} \qquad 0 \le n < N \qquad \text{(A-9)}$$

In equation (A-2), ω is chosen to be equal to the previous pitch ω_{-1}. The previous pitch was chosen because the implementation calls for the frequency to vary from the previous pitch to the current pitch. Also, ωn must be proportional to the l^{th} harmonic. This is shown below in (A-10).

$$\omega n = \omega_{-1} n l \qquad \text{(A-10)}$$

In equation (A-8), θ is defined independent of ω. For implementation in the coder, θ must be defined as a function of the l^{th} harmonic. This is shown in equation (A-11) below. This corresponds to equation (50) in the previous discussion.

$$\theta(l) = \omega N l + \phi(N) + \theta_{-1} \qquad N = 160 \qquad \text{(A-11)}$$

APPENDIX B

Reconstruction Window

Reconstruction Window

Down Window, $w_d(n)$

n	$w_d(n)$	n	$w_d(n)$	n	$w_d(n)$	n	$w_d(n)$	n	$w_d(n)$
0	1	32	1	64	0.72	96	0.25	128	0
1	1	33	1	65	0.71	97	0.24	129	0
2	1	34	1	66	0.69	98	0.22	130	0
3	1	35	1	67	0.68	99	0.21	131	0
4	1	36	1	68	0.66	100	0.19	132	0
5	1	37	1	69	0.65	101	0.18	133	0
6	1	38	1	70	0.63	102	0.16	134	0
7	1	39	1	71	0.62	103	0.15	135	0
8	1	40	1	72	0.6	104	0.13	136	0
9	1	41	1	73	0.59	105	0.12	137	0
10	1	42	1	74	0.57	106	0.1	138	0
11	1	43	1	75	0.56	107	0.09	139	0
12	1	44	1	76	0.54	108	0.07	140	0
13	1	45	1	77	0.53	109	0.06	141	0
14	1	46	0.99	78	0.51	110	0.04	142	0
15	1	47	0.97	79	0.5	111	0.03	143	0
16	1	48	0.96	80	0.49	112	0.01	144	0
17	1	49	0.94	81	0.47	113	0	145	0
18	1	50	0.93	82	0.46	114	0	146	0
19	1	51	0.91	83	0.44	115	0	147	0
20	1	52	0.9	84	0.43	116	0	148	0
21	1	53	0.88	85	0.41	117	0	149	0
22	1	54	0.87	86	0.4	118	0	150	0
23	1	55	0.85	87	0.38	119	0	151	0
24	1	56	0.84	88	0.37	120	0	152	0
25	1	57	0.82	89	0.35	121	0	153	0
26	1	58	0.81	90	0.34	122	0	154	0
27	1	59	0.79	91	0.32	123	0	155	0
28	1	60	0.78	92	0.31	124	0	156	0
29	1	61	0.76	93	0.29	125	0	157	0
30	1	62	0.75	94	0.28	126	0	158	0
31	1	63	0.74	95	0.26	127	0	159	0

Up Window, $w_u(n)$

n	$w_u(n)$	n	$w_u(n)$	n	$w_u(n)$	n	$w_u(n)$	n	$w_u(n)$
0	0	32	0	64	0.28	96	0.75	128	1
1	0	33	0	65	0.29	97	0.76	129	1
2	0	34	0	66	0.31	98	0.78	130	1
3	0	35	0	67	0.32	99	0.79	131	1
4	0	36	0	68	0.34	100	0.81	132	1
5	0	37	0	69	0.35	101	0.82	133	1
6	0	38	0	70	0.37	102	0.84	134	1
7	0	39	0	71	0.38	103	0.85	135	1
8	0	40	0	72	0.4	104	0.87	136	1
9	0	41	0	73	0.41	105	0.88	137	1
10	0	42	0	74	0.43	106	0.9	138	1
11	0	43	0	75	0.44	107	0.91	139	1
12	0	44	0	76	0.46	108	0.93	140	1
13	0	45	0	77	0.47	109	0.94	141	1
14	0	46	0.01	78	0.49	110	0.96	142	1
15	0	47	0.03	79	0.5	111	0.97	143	1
16	0	48	0.04	80	0.51	112	0.99	144	1
17	0	49	0.06	81	0.53	113	1	145	1
18	0	50	0.07	82	0.54	114	1	146	1
19	0	51	0.09	83	0.56	115	1	147	1
20	0	52	0.1	84	0.57	116	1	148	1
21	0	53	0.12	85	0.59	117	1	149	1
22	0	54	0.13	86	0.6	118	1	150	1
23	0	55	0.15	87	0.62	119	1	151	1
24	0	56	0.16	88	0.63	120	1	152	1
25	0	57	0.18	89	0.65	121	1	153	1
26	0	58	0.19	90	0.66	122	1	154	1
27	0	59	0.21	91	0.68	123	1	155	1
28	0	60	0.22	92	0.69	124	1	156	1
29	0	61	0.24	93	0.71	125	1	157	1
30	0	62	0.25	94	0.72	126	1	158	1
31	0	63	0.26	95	0.74	127	1	159	1

VITA

Walter D. Andrews

Candidate for the Degree of

Master of Science

Thesis: DESIGN OF A HIGH QUALITY 2400 BIT PER SECOND ENHANCED MULTIBAND EXCITATION VOCODER

Major Field: Electrical Engineering

Biographical:

Personal Data: Born in Clovis, New Mexico, March 18, 1965, the son of Kay Parker and Walter Andrews Jr.; married to Ronda Andrews and have two children Chuck and Caleb.

Education: Graduated from Sapulpa High School, Sapulpa, Oklahoma, in May, 1983; received Associates of Applied Science Degree in Electrical-Electronics Technology from Oklahoma State University Technical Branch Okmulgee, Okmulgee, Oklahoma, in September, 1986; received Bachelor of Science Degree in Electrical and Computer Engineering from Oklahoma State University in December, 1993; completed requirements for the Master of Science Degree at Oklahoma State University, Stillwater, Oklahoma, in December, 1994.

Professional Experience: Engineering Intern, Los Alamos National Lab, Los Alamos, New Mexico, June, 1991 to August, 1991; Engineering Intern, REDA Pump, Bartlesville, Oklahoma, June, 1992 to August, 1992; Research Assistant, Department of Electrical and Computer Engineering, Oklahoma State University, June, 1993 to present.

Professional Memberships: IEEE, Eta Kappa Nu, and Tau Beta Pi.