

ENHANCED SPECTRAL MODELING
FOR SINUSOIDAL SPEECH

CODERS

By

BUDDY J. WALLS

Bachelor of Science in Electrical Engineering

Stillwater, Oklahoma

1995

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, 1996

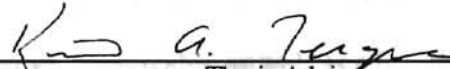
ENHANCED SPECTRAL MODELING

FOR SINUSOIDAL SPEECH

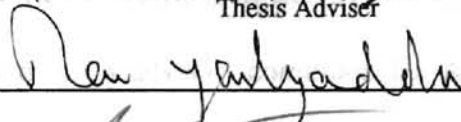
study was to determine ways to better model the spectral
CODERS

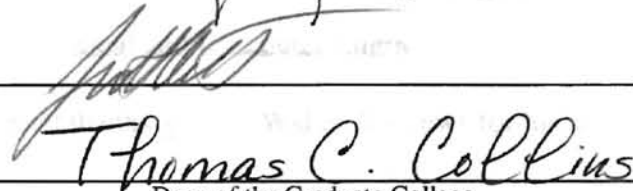
This study would not have been possible without
people who made it possible to do this research.
I would like to thank those people.

Thesis Approved:



Thesis Adviser





Dean of the Graduate College

ACKNOWLEDGMENTS

The purpose of this study was to determine ways to better model the spectral peaks for low bit rate sinusoidal coders. This study would not have been possible without the support and guidance of a number of people. I would like to take this opportunity to briefly acknowledge and thank some of those people.

First, I would like to thank my advisor, Dr. Keith Teague, for his guidance, suggestions, support and friendship throughout this study. I would especially like to thank Dr. Teague for the opportunity to work on this project. I can say without a doubt, that the last two years have been very enjoyable as well as educational. Additionally, I wish to thank the other members of my advisory committee, Dr. Acton, and Dr. Yarlagadda, as well as the School of Electrical and Computer Engineering for their support during the last two years. A very special thanks goes to Walter Andrews for his suggestions and friendship.

I would be very remiss if I didn't take this opportunity to thank the Department of Defense for funding the project that this study is based upon. A special thanks goes to Tina Kohler, Ron Kohn, and the late Tom Tremain for their guidance, and suggestions.

Finally, I would like to thank my parents, Ray and Polly Walls, as well as my Aunt and Uncle, Peggy and Jimmy Wells, for their help and understanding throughout my years in school. None of this would have ever been possible without their support.

TABLE OF CONTENTS

Chapter	Page
I. INTRODUCTION	1
Purpose.....	1
Implementation	3
Thesis Outline	4
II. BACKGROUND.....	6
Speech Coding	6
Waveform Coders	7
Analysis-by-Synthesis	9
Vocoders	12
Sinusoidal Coders	14
III. CURRENT SPECTRAL MODELS.....	20
Scalar Quantization	20
Cepstral Modeling.....	21
Linear Prediction.....	26
IV. SPLINES AND SPECTRAL WARPING	35
Spectral Interpolation.....	35
Spectral Warping	43
V. POSTFILTERING	50
VI. RESULTS.....	59
Computational Considerations.....	65
Perceptual Results.....	68
VII. CONCLUSION	70
Future Research	72

REFERENCES 74

LIST OF TABLES

Page

iv

LIST OF TABLES

Table	Page
I. Spectral distortion measures for increased LP orders	60
II. Spectral distortion measures for LP improvements	61
III. Spectral distortion for warped frequency scale	63
IV. Spectral distortion introduced by white noise correction	68

LIST OF FIGURES

Figure	Page
1. Performance of classes of speech coders	8
2. Analysis-by-synthesis structure.....	9
3a. Original speech signal.....	10
3b. MPLPC synthetic speech.....	11
3c. CELP synthetic speech.....	11
4. Simple vocoder speech production model	13
5a. Voiced speech spectrum.....	15
5b. Unvoiced speech spectrum.....	15
6a. Original speech spectrum	17
6b. All voiced synthetic speech spectrum	17
6c. Voicing decisions for speech signal.....	18
6d. Reconstructed speech spectrum	18
7. Magnitude spectrum using 40 cepstral coefficients	24
8. Magnitude spectrum using 12 cepstral coefficients	25
9a. 10th order LPC fit to voiced frame.....	29
9b. 10th order LPC fit to unvoiced frame.....	29
10. 14th order LPC fit to a voiced frame.....	30

11. 18th order LPC fit to a voiced frame	31
12. Gain based on sampled spectrum.....	32
13. Interpolation procedure for voiced speech.....	38
14a. Spline envelope for voiced frame, male speaker	40
14b. Spline envelope for voiced frame, female speaker	41
15a. Spline enhanced LP model, male speaker.....	42
15b. Spline enhanced LP model, female speaker.....	42
16. Mel warping function.....	44
17. Mel warped spline and spectrum	46
18. LPC fit to warped spectrum	48
19. Warped LPC vs. Non-Warped LPC	48
20. Effect of an all pole postfilter on a voiced spectrum.....	51
21. Effect of pole-zero postfilter on voiced spectrum	53
22. Postfilter for sinusoidal coders	56
23. Postfiltering with adaptive highpass filter	57
24. Whitened LP spectral fit (1/256)	66
25. Whitened LP spectral fit (1/24576)	67

CHAPTER I

INTRODUCTION

Purpose

This paper details the development of an improved method for representing the spectrum for sinusoidal speech coders. For voiced speech, sinusoidal coders require a very accurate representation of the underlying sinusoid amplitudes. An extremely popular class of sinusoidal coders is harmonic coders that model the harmonics as sinusoids tuned to interger multiples of the fundamental frequency. Accurate representation of the harmonic amplitudes is usually not compatible with the available bit rate in low rate coders. The method presented in this paper uses linear prediction in conjunction with a preprocessing stage to better model the harmonic amplitudes.

In recent years, speech coders based on sinusoidal models of speech production have received increased attention, particularly at relatively mid and low bit rates (below 8,000 bps). These sinusoidal coders belong to a class of speech coders known as vocoders. Vocoders use a parametric model to attempt to reproduce the sound of the original signal. Waveform coders represent another class of speech coders. These coders attempt to accurately reproduce the shape of the original waveform. As a general rule, at bit rates above 4,800 bps waveform coders outperform vocoders in the quality of the resulting synthetic speech. For bit rates below 4,800 bps, vocoders, particularly those based upon sinusoidal models, outperform waveform coding methods. In the recent testing by the United States Department of Defense Digital Voice Processing Consortium

(DDVPC) for a new 2,400 bps speech coding standard, five of the eight test coders were based on sinusoidal models [1].

Sinusoidal coders require a fairly accurate model of the spectrum, particularly for the underlying harmonic structure of voiced speech. In the past, spectral representations for harmonic coders have required large numbers of bits to achieve the desired quality. Inaccuracies in the modeling of the harmonic amplitudes are known to increase the reverberation and mechanical quality of the synthetic speech. The traditional method for spectral representation is to quantize and code the individual spectral amplitudes either individually or in a block wise fashion [2 and 3]. While this method is acceptable for bit rates at or above 4,800 bps, it is not efficient enough for low rate coding. This paper details a modified spectral representation that allows for the necessary accuracy in representing the harmonic amplitudes, while using only a fraction of the bits previously used.

The modified representation is based upon the well known linear predictive model. Linear prediction (LP) has a number of very desirable properties. The spectral envelope can be represented using a relatively small number of coefficients (typically 10-18). Using an alternate representation known as line spectral frequencies (LSF's), the model can be coded efficiently using either scalar or vector quantization (VQ) techniques. Essentially, the LP model attempts to match the spectral envelope in an overall minimum mean squared sense. While this fit follows the general shape of the speech spectrum well, it is somewhat lacking in the representation of the individual harmonic amplitudes. This type of fit is generally sufficient for waveform coders which are more interested in the overall

spectral shape and formant structure than the individual harmonic amplitudes. This paper presents a technique by which LP can model the harmonic amplitudes more accurately. The use of linear prediction for only the harmonic amplitudes is problematic, as will be shown in a later chapter. A much better method for emphasizing the harmonics is by interpolating the amplitudes between each harmonic, thus producing a slower varying spectral envelope that LP can more accurately model. In addition, methods to increase the perceptual quality of the resulting speech through spectral manipulation, such as spectral warping, and adaptive postfiltering are also included in this paper.

The proposed enhancements in spectral modeling are not limited to a single type of sinusoidal speech coder. Any coder that relies on an accurate estimate of the amplitudes of the spectral peaks can use the methods developed in this paper. The postfilter presented later is also applicable to a wide range of speech coders, at numerous bit rates.

Implementation

The improvements in the spectral modeling have been incorporated into a test coder to determine their validity and perceptual improvement over traditional methods. The test coder used is the Enhanced MultiBand Excitation Coder (EMBE) [4], developed at Oklahoma State University. The EMBE coder was a recent candidate in DDVPC tests for a new 2,400 bps voice coder. This coder is based upon improvements to the MultiBand Excitation (MBE) model, and enhanced methods of model parameter estimation. The MBE model will be explored in greater detail in a later chapter. In essence, the MBE model attempts to model speech with a combination of voiced and

unvoiced components for each frame. The input speech frame is divided into a set of frequency bands, with the voiced/unvoiced determination being made for each band.

Once the voicing decisions have been made for the speech frame the unvoiced portions are synthesized by using bandpass white noise, while voiced components are synthesized using a bank of sinusoidal oscillators tuned to harmonics of the fundamental frequency of the frame. The goal of the spectral model in MBE is to accurately represent the amplitudes of these sinusoidal oscillators and to fit the unvoiced segments in a mean squared sense. Thus the MBE model is an ideal candidate to test the proposed spectral model.

Thesis Outline

The remainder of this paper details the development of the improved method of representing the harmonic amplitudes. The use of postfiltering to improve the perceptual quality is also addressed in detail. A breakdown for the rest of this paper is presented in the following paragraphs.

Chapter 2 provides the reader with a brief background on speech coding, including traditional waveform coders and vocoders. A more thorough discussion of sinusoidal coding with particular emphasis on MBE, is presented.

Chapter 3 examines the various methods for representing the harmonic amplitudes. These methods include direct quantization along with various parametric models, such as cepstral modeling and linear prediction. The limitations and coding issues of these

methods will also be discussed. Particular emphasis is placed on the linear predictive model.

Chapter 4 explores the use of an interpolation function to improve the spectral fit obtained through linear prediction. The cubic spline interpolation function is discussed in detail. The complete process of computing a linear predictive model based on an interpolated spectral envelope is presented. Additionally, the use of spectral warping to improve the spectral match in perceptually significant areas is discussed.

Chapter 5 focuses on the use of adaptive postfiltering to improve the perceptual quality of the synthetic speech signal. The discussion focuses on all-pole, as well as, pole-zero postfilters. A discussion on the adaptation of these filters to the speech signal is also included.

Chapter 6 presents the results of the various proposed enhancements to the spectral model. Spectral distortion data is presented to evaluate the various improvements. Additionally, the computation considerations of linear prediction are addressed. Finally, perceptual results based on the incorporation of the changes into the EMBE speech coder are also discussed.

Chapter 7 provides the conclusion to this paper. A brief summary of the process of spectral interpolation and the effects on performance are presented. Finally, various suggestions for future research are also addressed.

CHAPTER II

BACKGROUND

Speech Coding

The goal of speech coding is, simply, to produce the highest quality speech using the least amount of data. The roots of speech coding can be traced back to the work of Dudley in the late 1930's [5]. In the last twenty years there has been a dramatic increase in the field of speech coding. This increase has been fueled by the demand for higher quality speech at succeeding lower bit rates. This chapter attempts to present a brief description and background of speech coding. The two major classes of speech coders, waveform coder and vocoders, will be examined. The sinusoidal model for speech production is also presented in some detail.

Let us first get a feel for the breadth of the speech coding field. Speech coding has many diverse and numerous applications. These range from standard telephone applications to compression and encryption. The most obvious of these applications is in the telephone industry. By coding a speech signal prior to transmission, the required bandwidth for the signal is dramatically reduced. This allows a larger number of calls to be placed on a given communication channel. This is especially important in the cellular telephone industry as channel congestion is already a problem in some areas. As cellular telephones increase in popularity in the next few years, this congestion will only worsen. Speech coding offers the promise of improved speech quality along with reduced bandwidth for these applications. To ensure interoperability among various users, speech

coding standards are being established. Digital cellular standards are already in place in Japan, the United States, and Europe.

The use of speech coding for storage and compression is also increasing dramatically. Digital answering machines employ speech coding algorithms to store both incoming and outgoing messages. This reduces the amount of memory needed by the answering machine, thus reducing their cost. Multimedia applications for home personal computers also use speech coding to efficiently store voice data, thus decreasing the amount of storage space needed by these applications.

The most important area that speech coding is being applied to, however, is in achieving secure communications. Digital data, in general, lends itself readily to encryption for security. A person can easily envision the use of digital speech for secure communications in a tactical environment, between embassies, in banking, or even for home use to prevent eavesdropping on personnel phone calls.

These are only a few of the examples of the use of speech coding. A more thorough discussion of these applications is found in [6]. We will now explore two of the primary classes of speech coders: Waveform coders and Vocoders.

Waveform Coders

Most speech coders can be broken down into two categories: Waveform coders and Vocoders. Waveform coders attempt to match the actual speech waveform, while vocoders try to only preserve the waveforms essential perceptual qualities. Figure 1 [7] illustrates the traditional performance characteristics of waveform coders and vocoders.

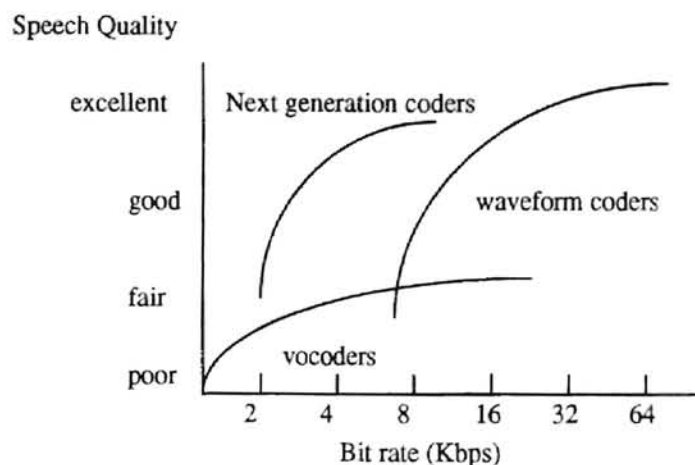


Figure 1 Performance of classes of speech coders

It can be seen that at higher bit rates waveform coders deliver superior performance compared to vocoders. However as the bit rate drops, particularly below 4,800 bps, vocoders significantly out perform waveform coders. The next generation of speech coders is expected to fill the gap between vocoders and waveform coders.

Early waveform coders were the result of research into efficient methods of quantizing the speech signal. The earliest type is Pulse Code Modulation (PCM). PCM basically maps each speech sample to a discrete set of quantization levels. PCM is a speech coding standard at 64 Kbps (ITU G.711) [8]. Another early waveform coder is Delta Modulation (DM) [9]. DM oversamples the signal using a 1 bit quantizer to quantize the samples. The samples are then integrated to recover the desired sample value. DM is used extensively in such applications as compact disc players. Adaptive Delta Pulse Code Modulation (ADPCM) is yet another early waveform coding technique. ADPCM uses a low order predictor to reduce some of the redundancies in the speech signal and an adaptive quantizer to quantize the residual. ADPCM is a speech coding standard at 32 Kbps (ITU G.721) [10].

Adaptive Predictive Coding, or APC, is a broad category of early waveform coders that uses a combination of long and short term predictors to code the speech signal. The long term predictor represents the pitch or the fundamental frequency of the vocal chord vibration, while the short term predictor models the shape of the vocal tract. APC forms the basis of the next class of waveform coder that will be discussed, analysis-by-synthesis coders.

Analysis-by-Synthesis

Analysis-by-synthesis coders attempt to match a synthetic (or coded) signal to the original signal by means of an iterative process. This iterative process typically attempts to minimize the mean squared error between the synthetic and a perceptually weighted version of the original signal. Perceptual weighting is an attempt to reduce the quantization noise in the synthetic speech by shifting it into regions in which it will be masked by the speech signal. The entire process is illustrated in Figure 2.

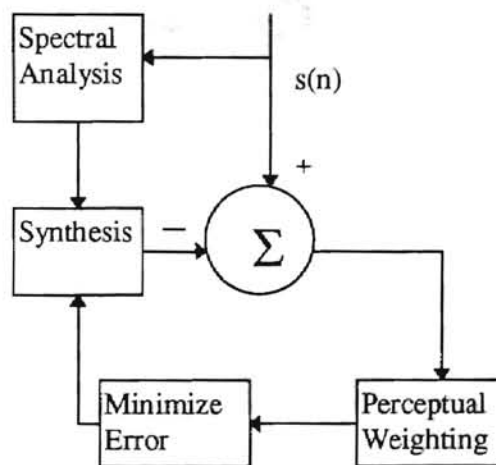


Figure 2. Analysis-by-synthesis structure

One of the first major analysis-by-synthesis coders was Multipulse Linear Predictive Coding (MPLPC) [11]. Multipulse attempts to model the spectral envelope through linear prediction (LP), while modeling the excitation as a sequence of pulses. The position and amplitude of the pulses are chosen so to minimize the error criterion. Typically four to eight pulses per 5 ms subframe are sufficient to produce high quality speech. An example of a speech waveform generated by MPLPC is shown in Figure 3b, with the corresponding original displayed in Figure 3a.

The limited number of pulses needed for mid to low rate coders poses problems for higher pitched speakers. These higher pitches require a larger number of pulses for the excitation than is typically available at these bit rates. The next type of analysis-by-synthesis coder, Codebook Excited Linear Prediction or CELP, overcomes this through the use of a table of excitations[12].

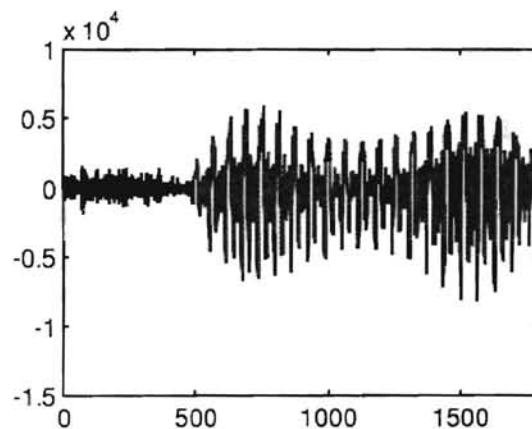


Figure 3a Original speech signal

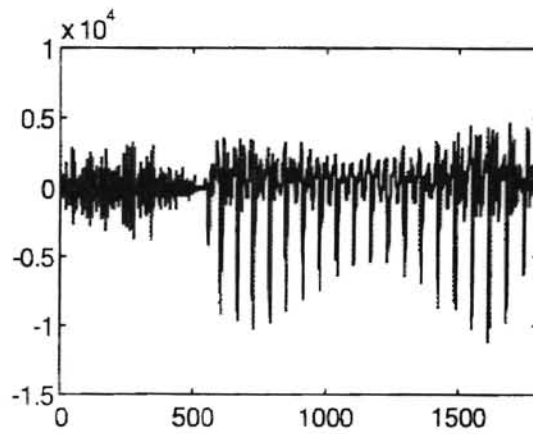


Figure 3b MPLPC synthetic speech

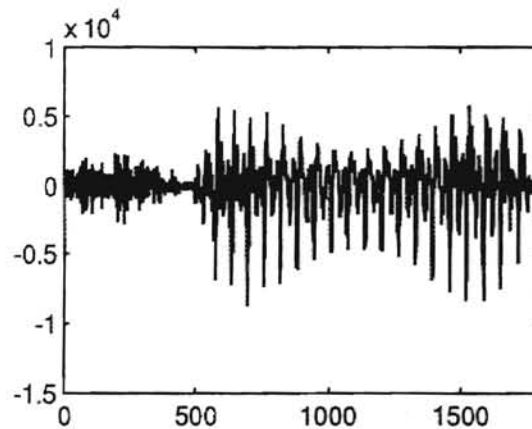


Figure 3c CELP synthetic speech

The CELP speech coding algorithm represents a major advancement in speech coding. CELP is similar in structure to MPLPC and can be viewed as a somewhat natural evolution of it. Instead of choosing a set of pulse amplitudes and locations to represent the excitation, CELP uses a codebook of excitations and simply chooses the best entry and transmits the corresponding codebook index to the synthesizer. As in MPLPC, a set of linear prediction coefficients is used to represent the short term structure, i.e. the spectrum, of the speech signal. The long term periodic structure of the speech signal, is modeled using a simple one or three tap pitch predictor. Some CELP coders, such as the

Federal Standard 1016 [13], incorporate an adaptive codebook to model the long term correlation instead of using a long term predictor. Once the contributions of the pitch predictor, or adaptive codebook, are removed from the signal, the remaining signal is modeled using a stochastic codebook. This codebook typically contains gaussian distributed noise that is used to model the aperiodic structure of the speech signal. Due to the presense of this underlying model, CELP and MPLPC are sometimes thought of as quasi-hybrid waveform coders. An example of a speech waveform processed with the CELP algorithm is shown in Figure 3c. Notice that compared to MPLPC shown in Figure 3b, the CELP waveform is closer in appearance to the original.

The success of CELP is apparent from the number of speech coding standards based on it. These include FS1016 at 4,800 bps, the North American TDMA standard and Japanese digital cellular standards, both of which are based on CELP variants. Also international standard ITU G.728 [14], G.723 [15], and G.729 [16] were recently adopted at 16 Kbps, 6.2 Kbps, and 8 Kbps, respectively. All of these standards are based on the CELP model. A much more thorough investigation of these coders as well as speech coding in general can be found in [17,7 and 5].

Vocoders

Vocoders, as mentioned previously, do not try to reproduce the original waveform shape, but instead only its perceptual qualities. Vocoders rely on a parametric model of the speech production process to code the speech signal. The earliest vocoders include the channel vocoder[18] and the phase vocoder[19]. The channel vocoder is basically the

same as the original coder used by Dudley in the late 30's. The channel vocoder separates the input speech signal into frequency bands using a set of bandpass filters. The output of these filters are then quantized and transmitted along with a voicing decision and a pitch value. Thus we can see that the parametric model that forms the basis of the channel vocoder includes a model for the spectrum and excitation.

An extremely popular parametric model for the speech production process is shown below in Figure 4. As can be seen, the speech waveform is modeled as the result of passing an excitation sequence through a time varying filter that represents the shape of the vocal tract. The excitation is chosen to represent either voiced or unvoiced speech. Voiced speech is characterized by the quasi-periodic vibration of the vocal chords, and is modeled as a pulse train. Unvoiced speech is the result of turbulent airflow through a constriction without the vocal chords vibrating, usually modeled as white noise. A voicing decision switch is used to control which excitation sequence is used. The time varying vocal tract filter is most often chosen to be an autoregressive model.

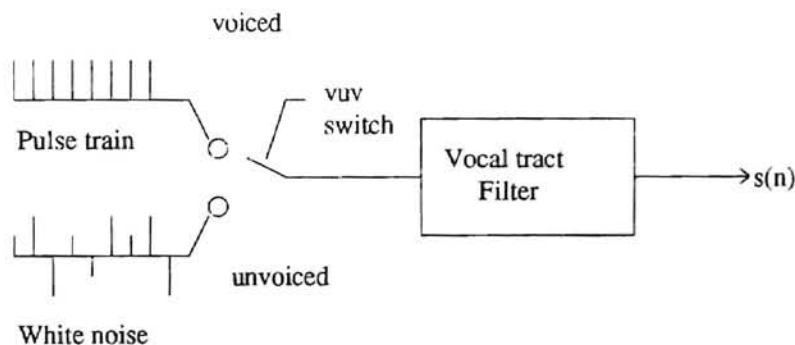


Figure 4. Simple vocoder speech production model

A speech coder based on the above parametric model, known as LPC-10e, is a federal standard at 2,400 bps (FS-1015) [20]. LPC-10e, uses a 10th order linear predictive

model to represent the spectrum, thus giving it its name. LPC-10e has been incorporated into a number of systems, many providing secure communications for military and non-military personnel.

While simple LPC type coders provide very intelligible speech, the quality leaves much to be desired. These coders tend to sound mechanical and artificial. As can be seen from Figure 4, the LPC model makes a single decision as to whether a frame is voiced or unvoiced. This is a gross simplification of the speech production process. It is known that speech, in general, exhibits a combination of both voiced and unvoiced excitations. It is believed that the binary voicing decision present in the LPC-10 model, introduces excess periodicity into the speech, which results in the reverberant, mechanical quality of the synthetic speech. This binary voicing decision is the reason that this type of coder is often referred to as a “buzz/hiss” coder. Additionally, errors in pitch estimation or voicing lead to annoying “anomalies” in the reconstructed speech signal. Numerous models have been proposed to attempt to alleviate this. One of the more recent models is the sinusoidal model, presented below.

Sinusoidal Coders

The development of speech coders based on sinusoidal models for speech production have increased dramatically in the last few years. Sinusoidal coders are based on the assumption that speech can be represented as a sum of sinusoids as given in (2.1).

$$s(n) = \sum_{l=1}^L A(l) \cos(\omega_l n + \theta_l) \quad n=0,1,\dots,M-1 \quad (2.1)$$

In (2.1), $A(l)$ represents the amplitude of each sinusoid, M is the number of samples in the frame, L is the number of harmonics in the frame, ω_l is the frequency of each sinusoid (not necessarily harmonically related), and θ_l represents the phase of each component sine wave. Figure 5a shows the spectrum of a frame of voiced speech. It is easy to see how a sinusoidal model can be intuitively derived from this by letting ω_l be harmonics of the fundamental frequency with $A(l)$ representing the l^{th} harmonic amplitude. Figure 5b shows a frame of purely unvoiced speech. Here it is not intuitive that a sinusoidal model can accurately represent this. By assigning random phases to each sinusoid it is possible, given enough sinusoids, to represent unvoiced speech. In fact spacing the sinusoids 100 Hz apart is sufficient to model unvoiced speech [21].

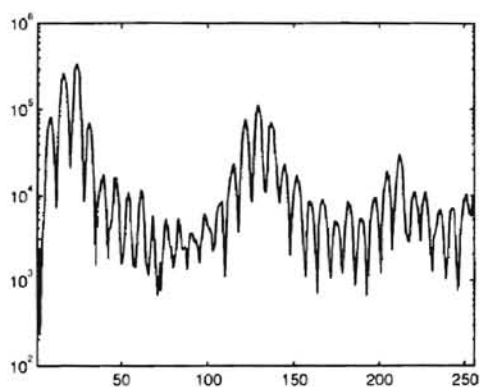


Figure 5a. Voiced speech spectrum

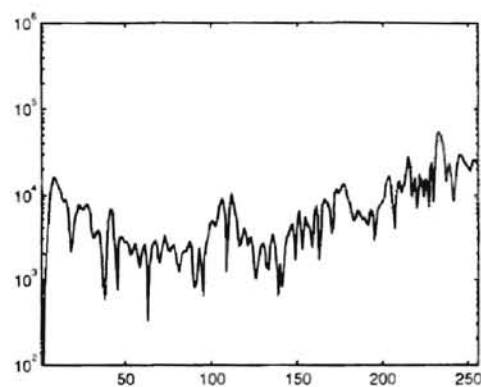


Figure 5b. Unvoiced speech spectrum

The sinusoidal model for speech production has several advantages over the LPC model previously discussed. The sinusoidal model does not constrain the speech signal to a single voicing state. By manipulating the phase function of the individual sinusoids, variable amounts of voicing can be introduced. This allows the model to follow both the

periodic and aperiodic components present in most frames of speech. Sinusoidal coders have been shown to be capable of producing extremely high quality speech.

Two of the most popular sinusoidal coders are sinusoidal transform coding (STC) [21], and the MultiBand Excitation (MBE) model [22]. STC is a direct application of equation 1. The analysis phase in STC consists of a peak picking algorithm that determines the amplitudes, frequencies, and phases of the underlying sinusoids using the short time fourier transform (STFT) of the input frame. The synthesizer in STC reconstructs the speech waveform by generating the resultant sinusoids in the time domain. A cubic interpolation function is used to maintain phase continuity across successive frames. It is known that slightly altering the phase relationship between sinusoids dramatically affects the perceptual quality of the reconstructed waveform. Thus, the interpolation procedure is critical for achieving high quality speech.

MBE, on the other hand, is technically not a pure sinusoidal coder. MBE represents the speech signal as a combination of voiced and unvoiced frequency bands. Voiced frequency bands are generated using a bank of sinusoidal oscillators, while unvoiced frequency bands are generated from bandpass white noise. MBE constrains the sinusoidal model by limiting the sinusoids to be harmonics of a fundamental frequency. The model is further constrained by the introduction of a set of voicing decisions for various frequency bands of the speech signal. MBE, traditionally, has outperformed pure sinusoidal coders, such as STC.

The analysis phase of an MBE coder estimates a pitch, voicing decisions, and spectral model for each frame of speech. The pitch value is typically determined on a

coarse grid and subsequently refined to sub-sample accuracy. The reason for the high degree of accuracy in the pitch estimate is in making the voicing decisions. Once the pitch estimate is obtained, a synthetic, all voiced speech signal is generated. The voicing decisions are made by comparing the match between the original and synthetic spectra on a harmonic by harmonic basis. An error in the pitch estimate will be multiplicative in frequency causing the harmonics at higher frequencies to be farther and farther from the location of the original harmonics, possibly resulting in large voicing errors.

Once the voicing decisions for the harmonics have been made they are grouped together into frequency bands. A binary decision is then assigned for each band. The voicing decision for each band represents the majority of the voicing decisions of the harmonics in the band. As many as 12 bands are used to relate the voicing states. Figures 6a-c illustrates the voicing decisions for a frame of speech. As can be seen from Figure 6c the fixed nature of the band structure has grouped some voiced harmonics into a frequency band in which the overall majority of the harmonics are unvoiced. This is one of the drawbacks of a fixed band structure. This can be partially compensated for by allowing the width of the voicing bands to adjust based on the current pitch of the speech signal [4].

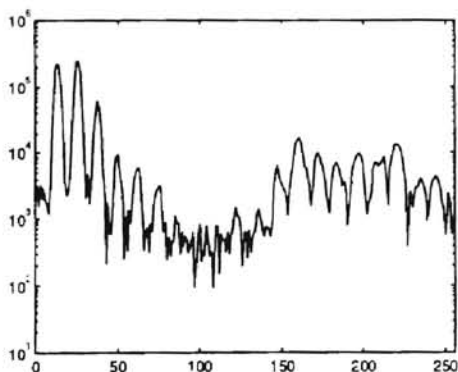


Figure 6a Original speech spectrum

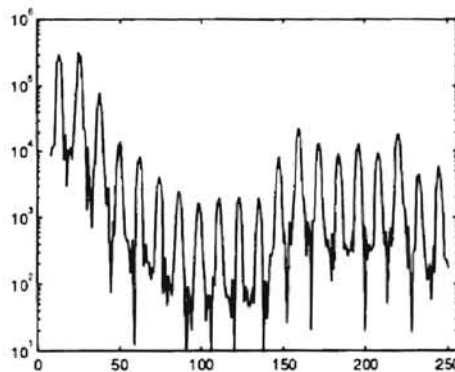


Figure 6b All voiced synthetic speech

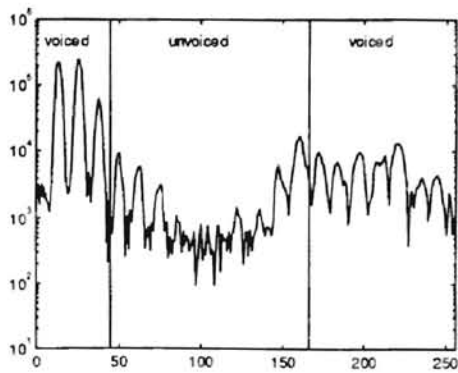


Figure 6c Voicing decisions for speech

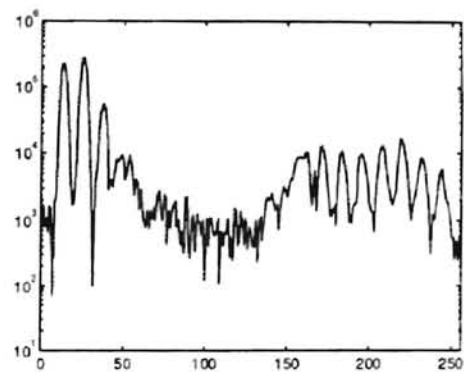


Figure 6d Reconstructed speech spectrum

The final stage of the analysis phase is the estimation of the spectrum. As previously mentioned MBE requires an accurate representation of the harmonic amplitudes for voiced speech, and an average fit to the spectrum, for unvoiced speech. The methods of this estimation and various improvements to it are the subject of the remainder of this paper.

Synthesis of the speech waveform in MBE is accomplished in two separate stages. For voiced frequency bands, a method similar to the one used in STC is employed. Sinusoids are generated in the time domain with the appropriate amplitudes and frequencies needed to represent the harmonics declared voiced in the analysis phase. The transmitted harmonic phases are used as the initial starting phases of the sinusoids. The phase is then continually tracked as long as the harmonics are declared voiced. Again phase continuity is maintained between the frame boundaries. Unvoiced frequency bands are generated using bandpass white noise. The resulting voiced and unvoiced signals are then summed to produce the output speech signal. Figure 6d shows the spectrum of the resulting speech waveform.

A variant of the MBE coder, known as Enhanced Multiband Excitation (EMBE), developed at Oklahoma State University, is discussed in greater detail later in this paper.

As can be seen from equation (2.1), the parameters used to represent speech include the individual sinusoid amplitudes, frequencies and phases. This large number of parameters is not conducive to a low bit rate coding scheme. A method of reducing this information while maintaining the accuracy of the sinusoidal model is the subject of this paper. The next chapter presents a brief overview of current methods of representing and coding the spectrum.

CHAPTER III

CURRENT SPECTRAL MODELS

Scalar Quantization

As was mentioned in the previous chapter, the parameters needed for a sinusoidal coder include the individual sinusoid amplitudes, frequencies and phases. This large number of parameters poses a problem for implementations of these coders at low bit rates. Methods must be devised to reduce the total number of bits required.

The bulk of the parameters in a sinusoidal coder correspond to the amplitudes and phases of the individual harmonics. Using scalar quantization, these parameters can be coded using between 94 and 184 bits per frame [23]. While this number of bits may be acceptable at 8 Kbps and above, at lower bit rates it is not feasible. The phases of the harmonics can be discarded by using a quasi-random initial phase for each harmonic and subsequently tracking the phases across frames to insure continuity. This considerably reduces the number of raw parameters needed to represent the spectrum. The two standards based on MBE, namely APCO [2] and INMARSAT-M [3], use this assumption along with an improved method of coding the harmonic amplitudes to represent the spectrum using only 76 bits per frame. This improved method of coding the harmonic amplitudes involves the use of the Discrete Cosine Transform (DCT) to exploit the redundancies that exist between the harmonic amplitudes in time as well as frequency [23]. Even with these changes, the resulting bit rate is still excessive for low bit rate coders, i.e.,

2,400 bps and below. Typically, a parametric model is used to represent the spectrum at these low bit rates.

Parametric spectral models are capable of representing the spectrum using a small number of coefficients. These models usually fit the spectral envelope using as few as 10 to 20 coefficients, thus allowing them to be quantized very efficiently. In the following sections two such parametric models, namely cepstral modeling and linear predictive coding, will be examined.

Cepstral Modeling

Cepstral modeling represents a way to parametrically represent both the magnitude and phase spectra through the use of the complex cepstrum. The complex cepstrum is an outgrowth of a larger area of signal processing known as homomorphic processing. The goal of homomorphic processing is to apply a generic superposition operator to linear systems in order to provide a linear mapping between input and output signals [24]. For the case of convolution, the log operator is one such function capable of doing this. The general idea is illustrated in (3.1-3.4) below, where $\hat{x}(n)$, $\hat{y}(n)$, and $\hat{h}(n)$ refer to the inverse Fourier transforms of (3.3).

$$y(n) = x(n) * h(n) \quad (3.1)$$

$$Y(\omega) = X(\omega)H(\omega) \quad (3.2)$$

$$\log[Y(\omega)] = \log[X(\omega)] + \log[H(\omega)] \quad (3.3)$$

$$\hat{y}(n) = \hat{x}(n) + \hat{h}(n) \quad (3.4)$$

The usefulness of these concepts can be seen by considering the case of voiced speech. Voiced speech is the result of passing a periodic impulse train through a vocal tract filter, as depicted in (3.1). By taking the log of the various terms, we can convert the convolution operation to a summation. It is known that the vocal tract filter is a low pass filter, thus a simple linear low pass filter could now be used to separate the vocal tract response from the excitation. This linear filtering in the log domain is referred to as liftering, a play on the word filtering. While this example is an oversimplified view, it serves to illustrate the goals of homomorphic processing.

Let us examine the cepstral model in more detail. Assume that the vocal tract response is given by (3.5) below. The complex cepstrum (CC) is defined as the inverse Z transform of the complex log spectrum. The result of the complex log operation is shown in (3.6), where $H_s(\omega)$ corresponds to the vocal tract transfer function, $A_s(\omega)$ is the magnitude response, and $\Phi(\omega)$ is the phase response of the vocal tract.

$$H_s(\omega) = A_s(\omega)e^{j\Phi(\omega)} \quad (3.5)$$

$$\log[H_s(\omega)] = \log[A_s(\omega)] + j\Phi(\omega) \quad (3.6)$$

The complex cepstrum differs from the real cepstrum (RC), or more commonly just the cepstrum, in that the RC is the inverse Z transform of only the magnitude spectrum. Thus we see that the RC doesn't contain any phase information due to the magnitude operation, while the CC preserves the phase information. The computation of the cepstral coefficients is shown in (3.7). This equation applies for both the real and complex cepstral coefficients. The difference is that for the real cepstrum the log operator that is used is the traditional one, and for the complex cepstrum, the log operator used is

the one defined in (3.6). Note that the equation is given for the real cepstral coefficients, designated by $c(n)$, instead of the complex cepstral coefficients $\gamma(n)$, for illustration purposes only. Also note that the amplitude spectrum, $A_l(\omega)$, used in (3.7), is a smoothed amplitude spectrum, obtained by interpolating the data between the harmonic peaks.

$$c(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} \log A_l(\omega) e^{j\omega n} d\omega \quad n=0,1,2,\dots \quad (3.7)$$

For the complex cepstrum, the calculation of the phase information is an involved process. Special care must be taken to “unwrap” the phase to obtain the correct value [24]. If however, the vocal tract system function is assumed to be minimum phase [25], then the computation of the complex cepstral coefficients, $\gamma(n)$, can be simplified greatly. Minimum phase systems contain poles and zeros which all lie within the unit circle. These systems have the property that the phase spectrum can be obtained directly from the magnitude spectrum. This is illustrated below in (3.8) where we see that the complex cepstral coefficients are obtained from the real cepstral coefficients.

$$\begin{aligned} \gamma(n) &= c(n) && \text{for } n = 0 \\ &= 2c(n) && \text{for } n > 0 \\ &= 0 && \text{for } n < 0 \end{aligned} \quad (3.8)$$

The procedure described above forms the basis of the use of cepstral coefficients to represent the spectrum [26]. Equations (3.9-3.13) show the exact form of this representation, where (3.10) results from the minimum phase assumption and the substitution of (3.8), and (3.12-3.13) result from comparing (3.11) to (3.6).

$$\log[H_s(\omega)] = \sum_{m=-\infty}^{\infty} \gamma_m e^{j\omega m} \quad (3.9)$$

$$\log[H_s(\omega)] = c_0 + 2 \sum_{m=1}^{\infty} c_m e^{j\omega m} \quad (3.10)$$

$$\log[H_s(\omega)] = c_0 + 2 \sum_{m=1}^{\infty} c_m \cos(m\omega) - 2j \sum_{m=1}^{\infty} c_m \sin(m\omega) \quad (3.11)$$

$$\therefore \log[A_s(\omega)] = c_0 + 2 \sum_{m=1}^{\infty} c_m \cos(m\omega) \quad (3.12)$$

$$\Phi_s(\omega) = -2 \sum_{m=1}^{\infty} c_m \sin(m\omega) \quad (3.13)$$

The determination of the cepstral coefficients is performed using (3.7). The magnitude and phase spectra are then obtained from (3.12-3.13). It is reported that 40 cepstral coefficients can accurately represent the spectrum [26]. An example of this is shown below in Figure 7, with the resulting amplitude spectrum superimposed upon the DFT.

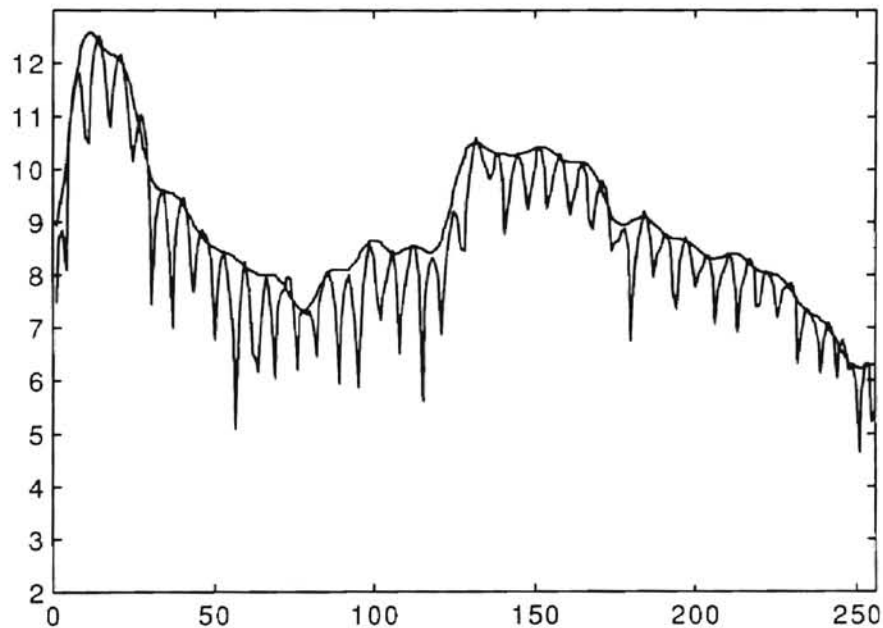


Figure 7. Magnitude spectrum using 40 cepstral coefficients

As can be seen above, the overall fit achieved using 40 cepstral coefficients is fairly accurate in representing the harmonic peaks. However, this number of coefficients is not conducive to a low bit rate application. One alternative is to reduce the number of cepstral coefficients, which reduces the quality of the fit considerably. Figure 8 shows how the fit degrades when the number of cepstral coefficients is reduced to 12.

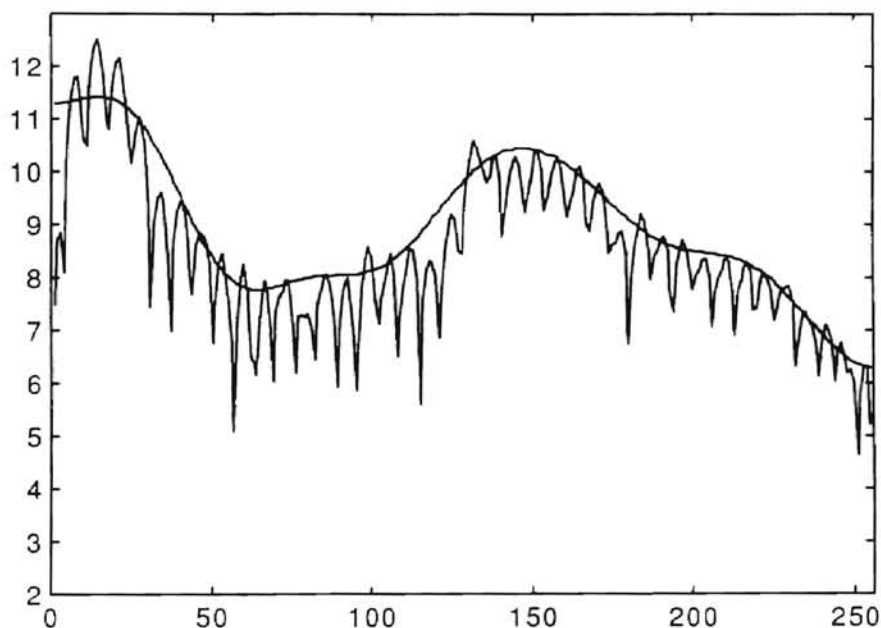


Figure 8. Magnitude spectrum using 12 cepstral coefficients

From Figure 8 it is obvious that 12 cepstral coefficients is only moderately successful in modeling the harmonic amplitudes. The use of other parametric models may be more appropriate for low bit rate coders. The most common method of representing the spectrum at low bit rates is Linear Prediction (LP), the topic of the next section.

Linear Prediction

Linear Prediction (LP) attempts to model the shape of the vocal tract using an autoregressive model. The vocal tract is generally modeled as the concatenation of nonuniform lossless tubes. This tube model is considered a resonant cavity and is approximated using an all pole model, such as LP. These resonances suggest that the speech signal exhibits significant correlation from one speech sample to the next. LP attempts to predict the current speech sample based on a weighted linear combination of past samples. The use of LP modeling in speech coding is well known, with a number of low bit rate speech coders incorporating it in one form or another, i.e. LPC-10e [20] and EMBE [4]. An excellent discussion of LP modeling can be found in [27 and 28].

Linear prediction coefficients can be coded efficiently using either scalar or vector quantization techniques. Typically, however, LP coefficients are first converted to an alternate representation, known as Line Spectral Pairs (LSP's), or Line Spectral Frequencies (LSF's) [29]. LSP's are known to be less sensitive to coding errors than LP coefficient's. An error in one line spectral coefficient affects the spectrum only near the associated frequency. In other words, the LSP's are frequency selective, a trait not directly possessed by LP coefficients.

Typically a fairly moderate order LSP spectrum, between 10th and 18th order, can be coded using a small number of bits. As an example the EMBE coder mentioned in chapter 2, uses a vector quantization scheme to code an 18th order LP model (represented as LSP's), using only 39 bits.

Let us examine how the LP model is obtained. Our derivation will be based upon a frequency domain approach instead of the more common time domain approach. While these two approaches produce identical results, the frequency domain version gives a better intuitive feel of what is happening in the context of spectral modeling. This follows the derivation presented in [28].

Assume that the vocal tract can be represented using an all pole form, given by (3.14)

$$S_{lp}(e^{j\omega}) = \frac{G}{A(e^{j\omega})} = \frac{G}{1 + \sum_{k=1}^P \alpha_k e^{-j\omega k}} \quad (3.14)$$

where G represents a gain factor, α_k is the k^{th} LP coefficient, and P is the model order. The frequency domain approach for computing the LP model attempts to minimize the ratio between the power spectrum of the original signal and that of the model. The power spectrum of the LP model is given by (3.15).

$$P_{lp}(e^{j\omega}) = |S_{lp}(e^{j\omega})|^2 = \frac{G^2}{|A(e^{j\omega})|^2} \quad (3.15)$$

To determine how accurately the model spectrum, $P_{lp}(e^{j\omega})$, matches the original power spectrum, $P(e^{j\omega})$, the error criterion in (3.16) is used. The model coefficients, α_k , can be obtained by minimizing the error with respect to each coefficient. This operation is shown below in (3.17). It can be shown that (3.18) follows from (3.15) and (3.16). R_k represents the k^{th} autocorrelation coefficient, given by (3.19).

$$E = \frac{G^2}{2\pi} \int_{-\pi}^{\pi} \frac{P(\omega)}{P_{lp}(\omega)} d\omega \quad (3.16)$$

$$\frac{\partial E}{\partial \alpha_i} = 0 \quad 1 \leq i \leq P \quad (3.17)$$

$$\frac{\partial E}{\partial \alpha_i} = 2 \left[R_i + \sum_{k=1}^P \alpha_k R_{|i-k|} \right] \quad (3.18)$$

$$R_k = \frac{1}{2\pi} \int_{-\pi}^{\pi} P(\omega) \cos(k\omega) d\omega \quad (3.19)$$

The solution for the LP coefficients and the gain are given by equations (3.20) and (3.21). Efficient methods exist to solve (3.20), such as the Levinson-Durbin recursion [18].

$$\sum_{k=1}^P \alpha_k R_{|i-k|} = -R_i \quad (3.20)$$

$$G^2 = R_0 + \sum_{k=1}^P \alpha_k R_k \quad (3.21)$$

Figure 9a below illustrates the LP spectral fit of a 10th order model to a given voiced frame. Figure 9b shows the resulting fit for an unvoiced frame. As can be seen from these figures, the LP model fits the spectrum in a general sense. While this is sufficiently accurate for unvoiced speech, this type of fit is often not accurate enough for voiced speech. As previously mentioned, sinusoidal coders, in particular, require a very accurate estimate of the harmonic amplitudes. Inaccuracies in these will lead to significant reverberation and an increased amount of buzziness in the reconstructed speech.

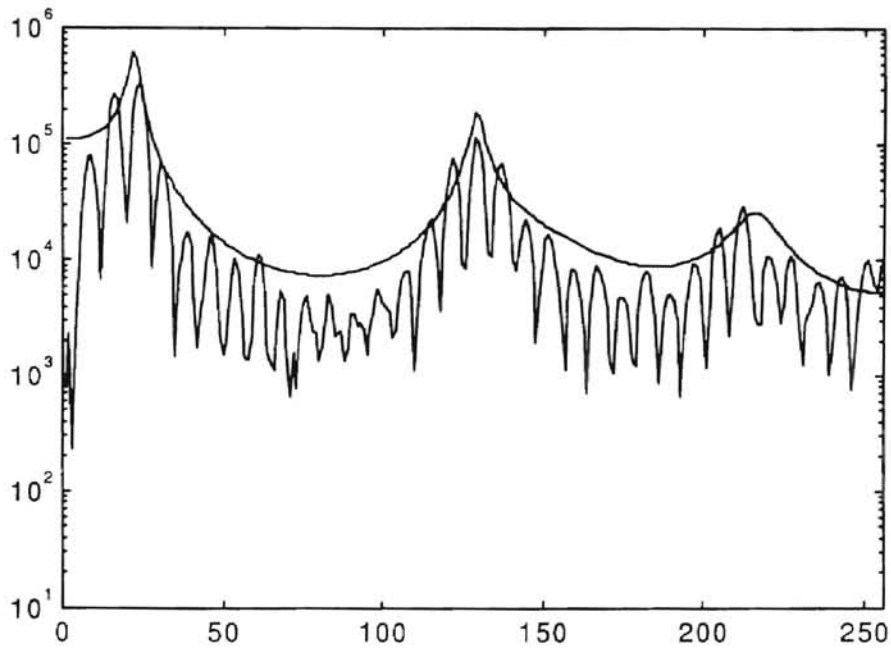


Figure 9a. 10th order LPC fit to voiced frame

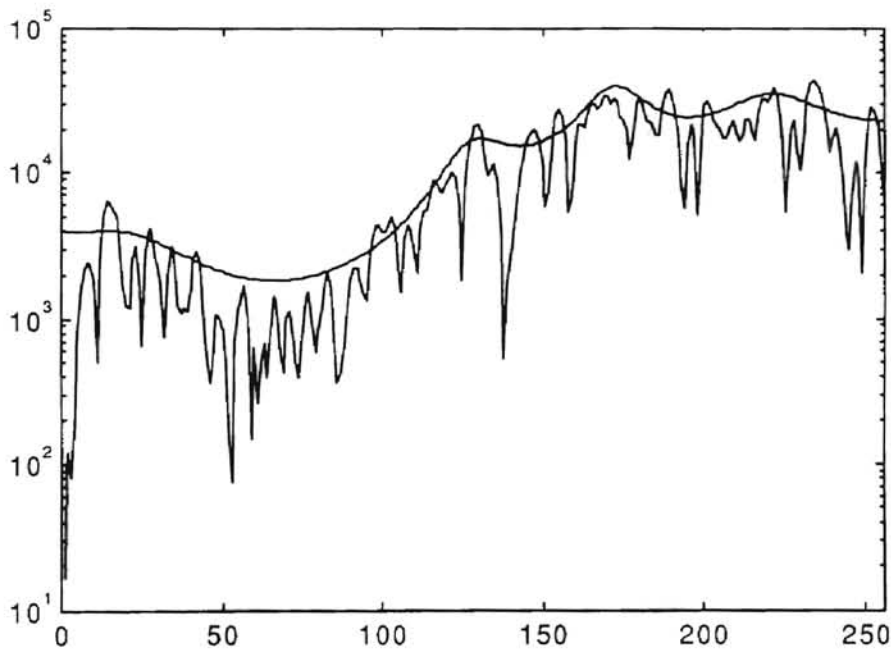


Figure 9b. 10th order LPC fit to unvoiced frame

The overall spectral fit can be improved by increasing the model order. It is well known that the autocorrelation function of a segment of speech, $R(m)$, and the autocorrelation function of the impulse response of the model, $R_s(m)$, are equal for the first $P+1$ values [18]. Thus by increasing the model order P , the autocorrelations of the speech signal and the model will match for a larger amount of data. In fact, any spectrum can be arbitrarily closely approximated by an all pole model simply by increasing the model order. Most sinusoidal coders increase the model order so as to improve the spectral representation. Figures 10 and 11 illustrate the improvements for a 14th order and 18th order model respectively.

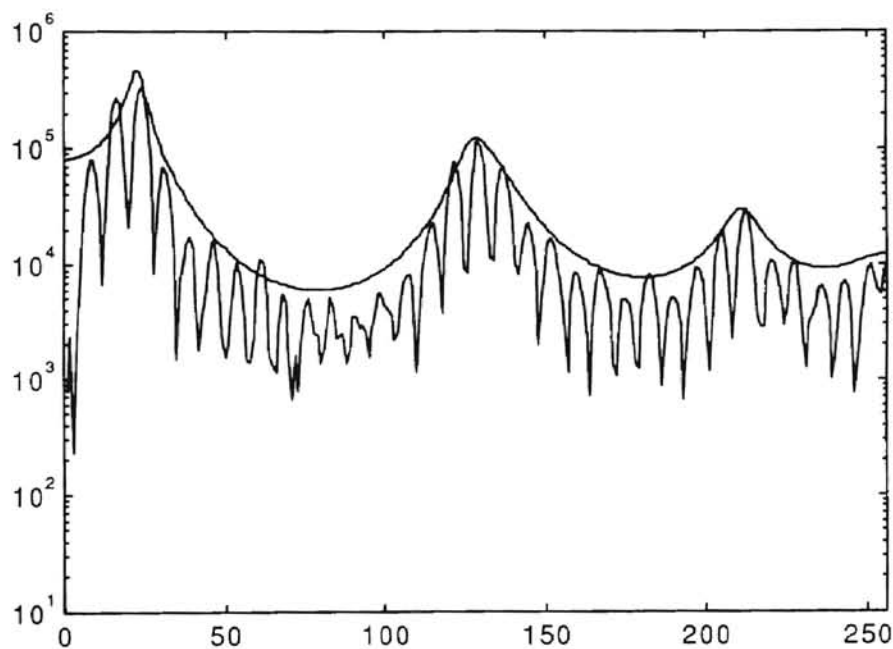


Figure 10. 14th order LPC fit to a voiced frame

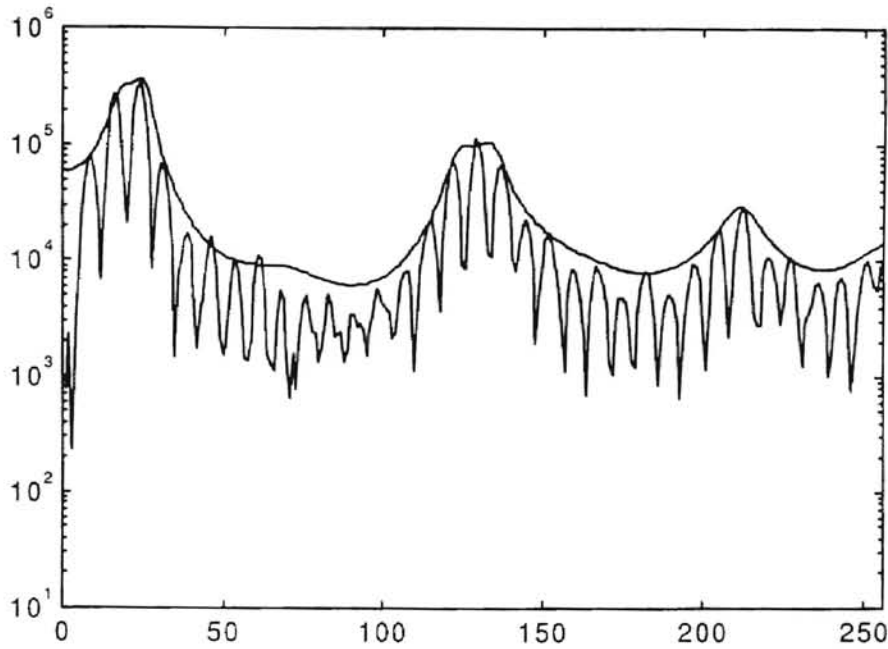


Figure 11. 18th order LPC fit to a voiced frame

Thus we can see that the 14th and 18th order LP fits are far superior to the 10th order LP fit for voiced speech.

The gain value computed in (3.21) is designed to provide an accurate overall match to the energy of the original signal. For harmonic coders, however, the gain should match the harmonic amplitudes primarily, not the overall spectrum. For female speakers in particular, the overall match presented in (3.21) can lead to an inaccurate gain level for the spectra of voiced speech. This is due to the relatively wide spacing of the harmonics and the small amount of inter-harmonic energy. A more accurate gain measure is shown in (3.22) [4], where ω_0 corresponds to the fundamental frequency and $P(k)$ and $P_{lp}(k)$ are the original and model power spectra.

$$G^2 = \frac{\sum_{k=1}^M P^2(\omega_0 k)}{\sum_{k=1}^M P^2_{ip}(\omega_0 k)} \quad (3.22)$$

Here the gain is determined as the ratio of the energy of the spectra, sampled at the harmonics. This provides a better match to the original spectrum for voiced speech. This is illustrated in Figure 12 below, where the dotted line corresponds to the gain calculation from (3.21), and the solid line represents the new gain value from (3.22).

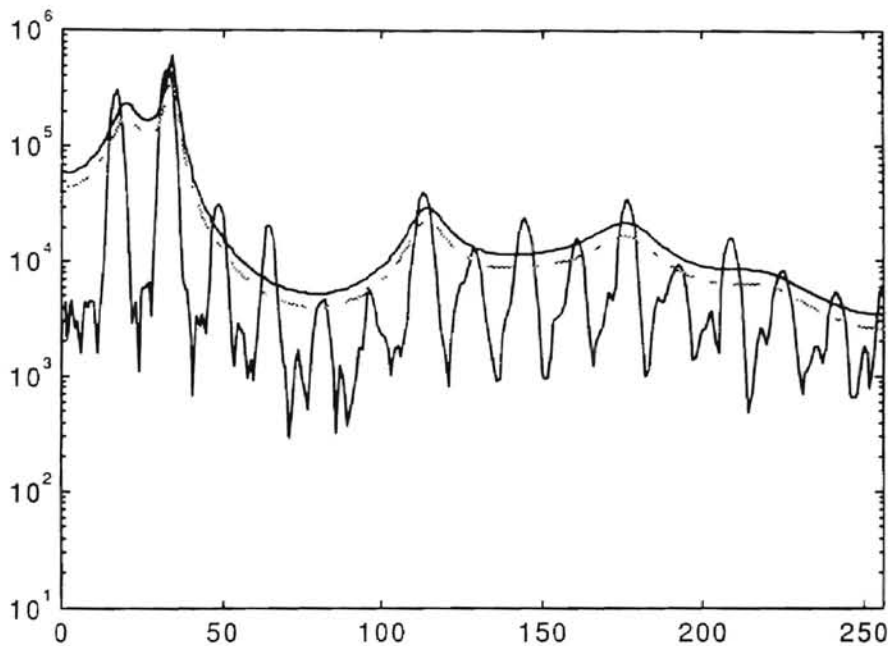


Figure 12. Gain based on sampled spectrum

Ideally for a harmonic coder, such as MBE, we would like to have an accurate fit at the harmonic amplitudes. In essence, the model would not need to represent the continuous spectrum, but a discrete spectrum containing only the harmonic amplitudes. However, fitting an LP model to a discrete spectrum is problematic [30].

A new spectral model, known as Discrete All Pole modeling (DAP) [31], attempts to overcome this problem. It is demonstrated in [31] that minimizing (3.16), for a discrete spectra, translates into trying to fit a continuous LP spectrum to a discrete spectrum. This results into equating the first P continuous autocorrelation coefficients with the first P discrete autocorrelation coefficients. The discrete autocorrelation, $R(i)$, is an aliased version of the continuous autocorrelation, $R_{cont}(i)$ as is shown in (3.23).

$$R(i) = \sum_{l=-\infty}^{\infty} R_{cont}(i - lN) \quad (3.23)$$

It is evident from (3.23) that as the number of discrete frequencies, N , decreases, the aliasing will get worse. This in turn implies that as the pitch increases, the aliasing increases.

The underlying reason for this can be traced back to the error criterion used in computing the coefficients of the all pole filter. The error criterion used in equation (3.16), possesses a “cancellation of errors” property [27]. That is, errors with $P(\omega) > P_{ip}(\omega)$ tend to cancel the errors where $P(\omega) < P_{ip}(\omega)$. This is somewhat easier to visualize in the following scenario. Taking the log of (3.16) and assuming that $P(\omega)$ is smooth in relation to $P_{ip}(\omega)$. This yields a new error criterion, E' , shown below in (3.24).

$$E' = \frac{1}{2\pi} \int_{-\pi}^{\pi} \log \frac{P(\omega)}{P_{ip}(\omega)} d\omega \quad (3.24)$$

From this equation we see that these types of spectral errors are capable of cancellation.

Numerous alternate error measures have been proposed, such as the Itakura - Saito error measure [32]. This is the error measure that DAP is based on. DAP uses an

iterative technique to obtain the correct spectral representation for discrete spectra. The final spectral envelope is arrived at using a gradient descent technique. Due to its computational complexity, this iterative technique is one of the major drawbacks of the DAP algorithm. In the following chapter an alternate method for improving the spectral fit of LP for harmonic coders is presented. This method is based on the use of spectral interpolation prior to calculation of the LP model.

CHAPTER IV

SPLINES AND SPECTRAL WARPING

Spectral Interpolation

In the previous chapter we saw some of the problems that are inherent in performing linear prediction on discrete spectra. These problems are due to the aliasing that occurs in the autocorrelation domain. This aliasing can be partially avoided by interpolating the discrete points to obtain a smoother, more correlated spectral envelope. This chapter will examine the effects of interpolation on the linear prediction model, with particular emphasis on the cubic spline interpolation function.

A number of different interpolation functions have been proposed for use in spectral modeling [30]. These include simple linear interpolation, parabolic interpolation functions, and cubic spline interpolation functions. The later approach has recently been taken by a number of speech researchers [33, 34 and 35]. The cubic spline is the mathematical equivalent of the mechanical splines that draftsmen use to smoothly connect points. Cubic splines, in this case, are used to smoothly connect the harmonic amplitudes, so as to improve the resulting fit obtained by an all pole model. Cubic spline interpolation can be viewed as a preprocessing stage before the application of linear prediction.

First, we need to more formally present the spline. The spline functions will not be derived here, only presented. The spline model used in this paper is obtain by a pragmatic approach that emphasizes performance more than mathematical vigor [36]. For a more mathematical treatment of splines, the reader is directed to [37].

The starting point for this approach is the simple cubic polynomial function, given in equation (4.1)

$$s_i(x) = a_i x^3 + b_i x^2 + c_i x + d_i \quad i = 1, \dots, L \quad (4.1)$$

where $s_i(x)$ represents the i^{th} spline connecting points (x_i, y_i) and (x_{i+1}, y_{i+1}) . Note that L in the above equation represents the total number of individual spline segments. These individual spline functions are specified by four coefficients (a_i , b_i , c_i , d_i) which are solved, based on a set of constraints. These constraints are used to tailor the spline to our particular use as an interpolation function between harmonic amplitudes. The first constraint is fairly obvious, the spline must pass through points (x_i, y_i) and (x_{i+1}, y_{i+1}) . The spline is also required to bend smoothly around these points. In other words, we require the first and second derivatives at (x_i, y_i) and (x_{i+1}, y_{i+1}) to match. The spline equations are given in (4.2-4.5) below. These equations represent the constraints on the general cubic polynomial given in (4.1). These constraints are chosen to enforce continuity and smoothness at the polynomial boundaries. Equation (4.6) results from expressing the general spline equation (4.1) in an alternate form and enforcing smoothness of its first derivatives [36]. The unknowns in the equations include the spline coefficients a_i , b_i , c_i , and d_i , as well as the second derivatives of the each spline segment, x_i . Two more conditions are needed to solve this system of equations, namely the conditions on the derivatives at the 1st and L^{th} spline segments. These are shown below in (4.7-4.8).

$$s_i(l_i) = a_i l_i^3 + b_i l_i^2 + c_i l_i + d_i = y(l_i) \quad (4.2)$$

$$s_i(l_{i+1}) = a_i l_{i+1}^3 + b_i l_{i+1}^2 + c_i l_{i+1} + d_{i+1} = y(l_{i+1}) \quad (4.3)$$

$$s_i''(l_i) = 6a_i l_i + 2b_i l_i^2 = p_i \quad (4.4)$$

$$s_i''(l_{i+1}) = 6a_i l_{i+1} + 2b_i l_{i+1}^2 = p_{i+1} \quad (4.5)$$

$$(l_i - l_{i-1})d_{i-1} + 2(l_{i+1} - l_{i-1})d_i + (l_{i+1} - l_i)d_{i+1} = \quad (4.6)$$

$$6 \left[\frac{y(l_{i+1}) - y(l_i)}{l_{i+1} - l_i} - \frac{y(l_i) - y(l_{i-1})}{l_i - l_{i-1}} \right]$$

$$p_1 = 0 \quad (4.7)$$

$$p_L = 0 \quad (4.8)$$

The actual solution to this system of equations is given in [36]. From (4.1) we see that the entire spline function is the superposition of these smaller splines connecting the harmonic amplitudes. An implementation of this procedure is also found in [36].

Now that the basis for cubic splines has been presented, we turn our attention to the specific task of fitting a cubic spline model to a speech spectrum. In the previous chapter it was mentioned that linear prediction performed acceptable for unvoiced frames, but was not as effective for voiced frames. Thus, the use of spline preprocessing will be restricted to voiced frames only.

The splines require a set of control points, or knots, to control the positioning of the spline itself. The knots are chosen as harmonic amplitudes, since these are the values we wish to model. Figure 13 illustrates the entire procedure of using cubic splines in conjunction with LP modeling to accurately fit the harmonics. It is assumed that the speech is already segmented into frames in which the speech waveform is quasi-stationary, denoted by $s(n)$. The output of this procedure is the enhanced model spectrum $S_{lp}(k)$.

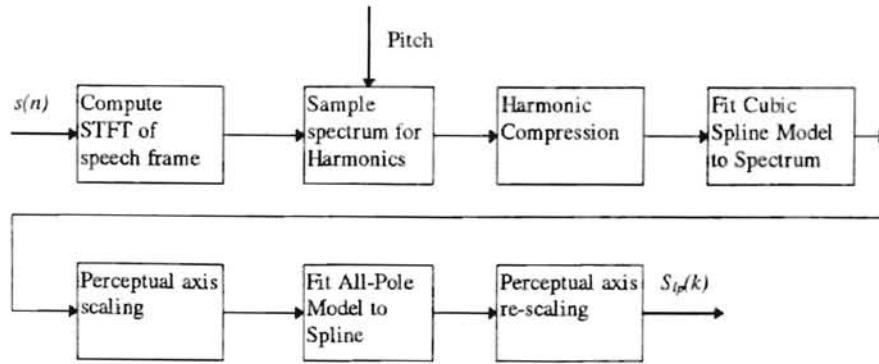


Figure 13. Interpolation procedure for voiced speech

The first block represents the traditional estimate of the short time spectrum of speech signal. This is typically accomplished using the Discrete Fourier Transform (DFT). Implicit in this block is the windowing operation that must precede the DFT. A Hamming window of 15-30 msec is typically used. Longer duration windows yield a higher frequency resolution at the expense of temporal resolution. The calculation of the DFT is shown below in (4.9) where, N is the size of the DFT, $s(n)$ corresponds to the speech signal for the current frame and $w(n)$ is the window used.

$$S(k) = \sum_{n=0}^{N-1} s(n)w(n)e^{\frac{-2\pi jkn}{N}} \quad 0 \leq k < N-1 \quad (4.9)$$

As previously mentioned, the goal of the spectral representation for a harmonic coder is the accurate representation of the harmonic amplitudes. The second block in Figure 13 represents the sampling of the spectrum for these amplitudes.

The locations of the harmonics are obtained based on the pitch value previously calculated for the frame. This calculation is not shown in Figure 13 and is assumed to have taken place in a previous stage. A high degree of accuracy is required for this pitch estimate. Inaccuracies in the pitch estimate will introduce errors in the sampling of the

spectrum that worsen as frequency increases. As an example, consider a speech signal sampled at 8 KHz, with a fundamental frequency of 200 Hz. This translates into a pitch period of 40 samples. If the pitch estimate is off by 1 sample, say an estimate of 41 samples, this will translate into an error of roughly 100 Hz at the upper end of the spectrum. These small errors will alter the sampling points of the spectrum. This will introduce inaccuracies in the spectral envelope generated by the spline interpolation.

The fourth block compresses the harmonic amplitudes. A logarithmic compression function is used to reduce the dynamic range of the amplitudes. The form of this compression function is given below in (4.10), where $A(l)$ is the DFT magnitude spectrum sampled at the harmonics, $A_c(l)$ is the compressed harmonic spectrum, and L is the number of harmonics in the frame.

$$A_c(l) = \ln[A(l)] \quad 1 \leq l \leq L \quad (4.10)$$

This is similar to the approach taken in scalar quantizers, such as μ law. It has been reported [30], that the use of harmonic compression improves the overall spectral match. In chapter 6 a quantitative evaluation of this procedure is presented.

The blocks labeled perceptual axis scaling and perceptual axis rescaling refer to the warping of the traditional frequency axis onto a perceptually more meaningful scale, such as the bark or mel scales. These scales are based roughly on the frequency response characteristics of the human ear. These concepts will be explored later in this chapter.

The two remaining blocks represent the bulk of the work: Fitting a spline to the compressed harmonic amplitudes and generating a linear predictive model based on the

spline envelope. It is these two blocks that form the foundation for much of the upcoming discussion.

The actual fitting of the cubic spline to the harmonics amplitudes is accomplished using (4.2-4.8), where the $y(l_i)$ corresponds to the harmonic amplitude located at DFT index point l_i , where l_i is a multiple of the pitch, and L is as previously defined.

Figures 14a and b show the result of fitting a cubic spline to a voiced frame for a male and a female speaker. As mentioned previously, all of the harmonic amplitudes serve as control points for the spline.

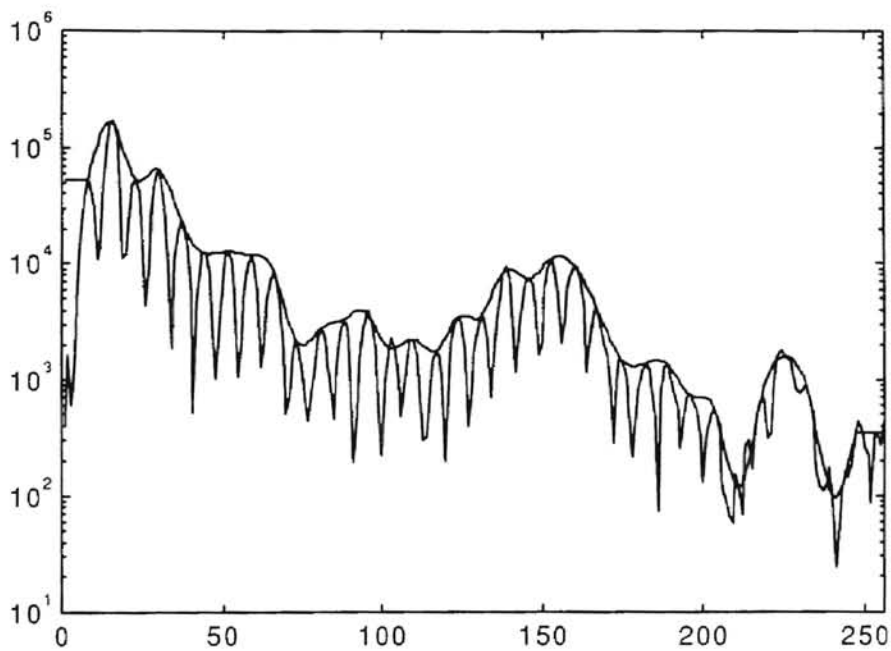


Figure 14a Spline envelope for voiced frame, male speaker

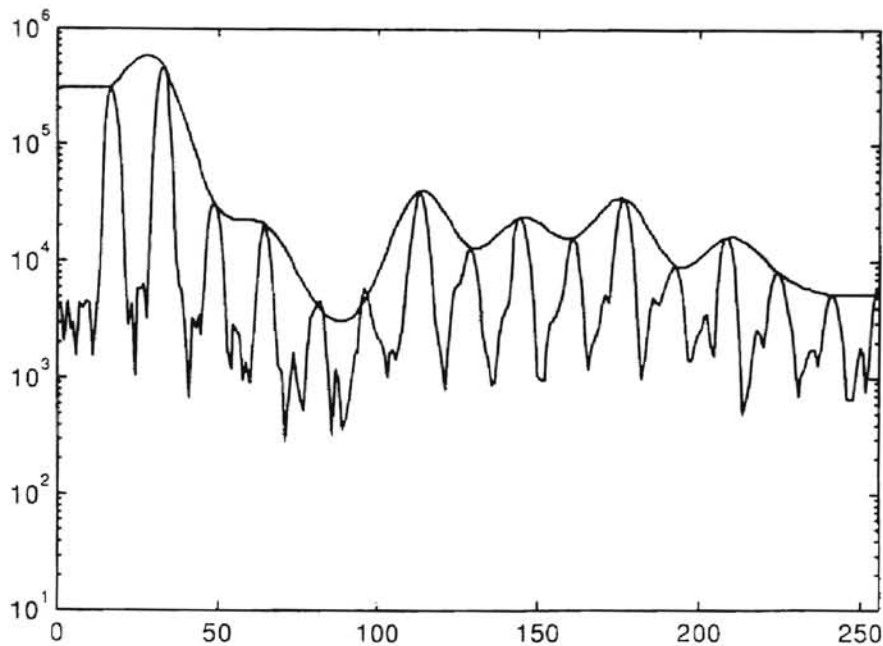


Figure 14b. Spline envelope for voiced frame, female speaker

Once the spline envelope is obtained, the envelopes are expanded using the inverse of the compressing function. A LP model is then fitted to the envelope using the techniques discussed in the previous chapter. No special care is needed when fitting a LP model to an unwarped spline envelope. If a warping function is first applied to the spline function to emphasize the perceptually important frequency ranges, then special care must be taken when fitting the LP model. This topic will be explored more thoroughly later in this chapter. Figures 15a and b show the effect of fitting a 14th order LP model to the spline envelope. The original LP model is also shown in each plot as the dashed curve.

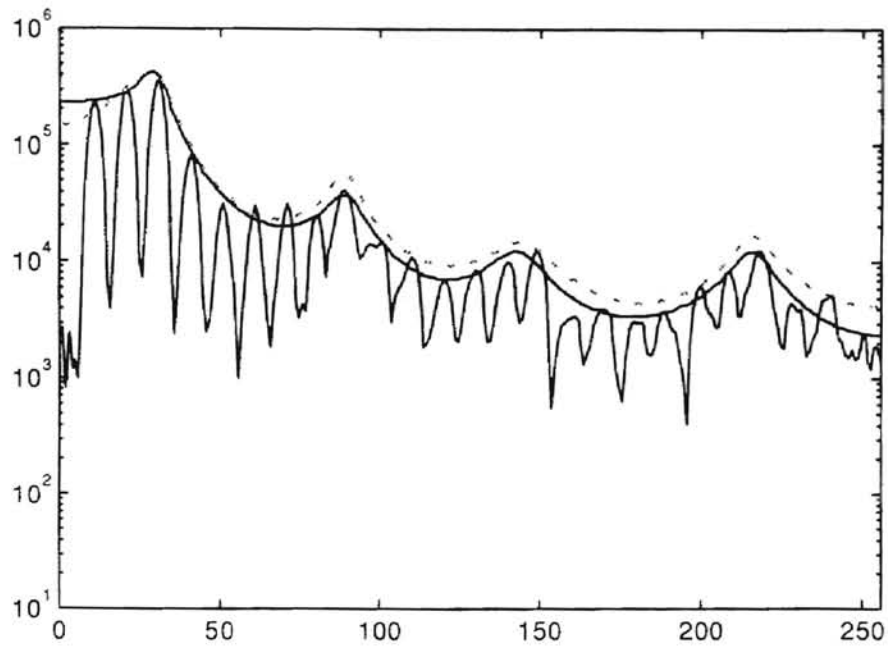


Figure 15a. Spline enhanced LP model, male speaker

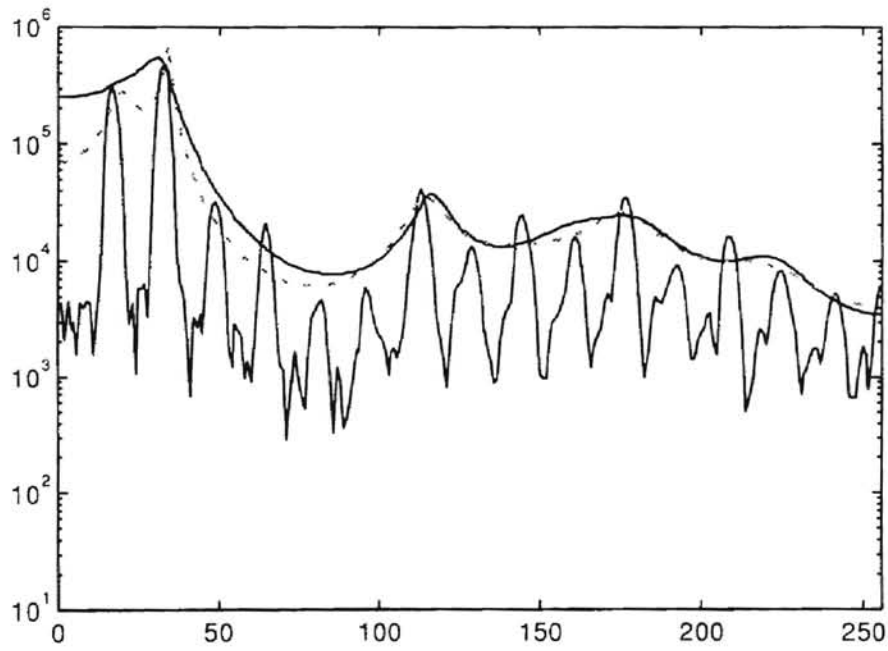


Figure 15b. Spline enhanced LP model, female speaker

As can be seen from the above figures, the spline preprocessing stage improves the spectral fit obtained through linear prediction. The most noticeable improvement appears in the lower half of the spectrum, around and below the first and second formant. This area is perceptually important, since errors at the low frequency end of the spectrum are more noticeable than errors in the upper frequencies.

Quantitative results of spline preprocessing are presented in chapter 6. At this point it is interesting to examine the underlying assumptions in generating the spline envelope. As presented, the cubic spline interpolation function is matched equally to all the harmonic amplitudes. In other words, the cubic spline envelope is not biased toward any particular spectral region. This approach allows an LP model to better represent the spectral amplitudes as a whole. However, it is well known that each harmonic is not perceptually equal. As just mentioned, errors in the low frequency, high energy harmonics are much more noticeable than errors in higher frequency, lower energy harmonics. Thus it may be advantageous to weight the spline model to track the lower frequency harmonics better. Alternatively, the LP model can be biased toward the lower frequency harmonics. The latter is the approach taken in this paper. The biasing that is used involves mapping the spectrum onto a more perceptually meaningful scale, in this case, the mel scale.

Spectral Warping

Spectral warping involves the transformation from one frequency axis to a different frequency axis. The warping function that will be examined in this paper is the mel warping function [25]. The mel scale is the result of a set of psychoacoustic

experiments into the way pitch is perceived. The mel itself is a unit of perceived pitch. In these experiments the frequency of 1000 Hz was arbitrarily assigned a value 1000 mels. Listeners were then asked to increase the frequency until the pitch that they perceived was twice the original. This frequency would be assigned the value of 2000 mels. The experiments continued in this manner. The results indicate that below roughly 1000 Hz, the frequency response of the ear is approximately linear, while above it the response is more logarithmic. A closed form expression for this mapping is given in (4.11), where, f_{hz} represents the frequency in hertz and f_{mels} corresponds to the warped frequency expressed in mels. This plot of this function is presented below in Figure 16.

$$f_{mels} = \begin{cases} f_{hz} & f_{hz} < 1000Hz \\ 1000 \log_2 \left(1 + \frac{f_{hz}}{1000} \right) & f_{hz} \geq 1000Hz \end{cases} \quad (4.11)$$

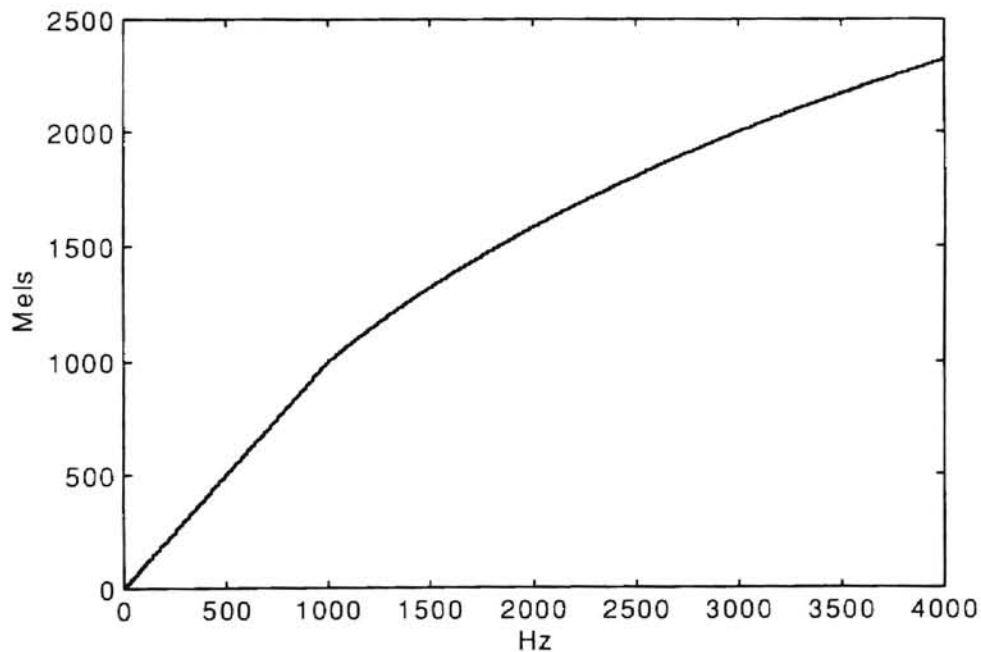


Figure 16. Mel warping function

As was mentioned previously, the goal of mel warping is to better model the more perceptually important parts of the spectrum, such as the lower frequency, higher amplitude regions. A number of recent papers on the use of splines in conjunction with an all pole spectrum indicate that the use of spectral warping improves the quality of the resulting speech [33 and 34]. This improvement is usually characterized as a decrease in the amount of reverberation in the output speech.

Referring to Figure 13, we can see that the use of spectral warping occurs after the spline fit has been generated for the spectrum. This is the opposite of the usual method employed. The usual course of action is to warp the magnitude spectrum and then compute the spline fit to the warped spectrum. However, this approach has some potential problems. Since the spectrum is discrete data, some detail will get lost due to the warping. Higher frequencies will get mapped onto the same index points, thus causing a loss in data. This is particularly worrisome when the resulting spectrum will be sampled for the harmonic amplitudes as previously discussed. Thus a preferable solution is to warp the spline envelope instead. This greatly reduces the chance that data will be lost at higher frequencies, due to the smoother and slower varying nature of the envelope, compared to the original DFT magnitude spectrum.

The actual mathematical process of warping the spline envelope, along with the discrete form of (4.11) is shown below in (4.12-4.13). The length of the warping, N' , is given by (4.14), F_s corresponds to the sampling frequency, and N is the length of the original DFT.

$$\hat{k} = \begin{cases} k & 0 \leq k < \frac{N}{8} \\ 1000 \log_2 \left(1 + \frac{8k}{N} \right) \frac{N}{F_s} & \frac{N}{8} + 1 \leq k < \frac{N}{2} - 1 \end{cases} \quad (4.12)$$

$$S(\hat{k}) = S_{sp}(k) \Big|_{k=\hat{k}} \quad 0 \leq \hat{k} < \frac{N'}{2} - 1 \quad (4.13)$$

$$N' = 2 \left\lceil 1000 \log_2 \left(1 + \frac{8 \left(\frac{N}{2} - 1 \right)}{N} \right) \frac{N}{F_s} \right\rceil + 1 \quad (4.14)$$

Figure 17 shows the result of warping the spectrum for a male speaker. Notice that there are now only 149 unique DFT indexes, compared to 256 for a 512 point DFT. For illustration purposes only, the actual DFT has also been spectrally warped, along with the spline envelope.

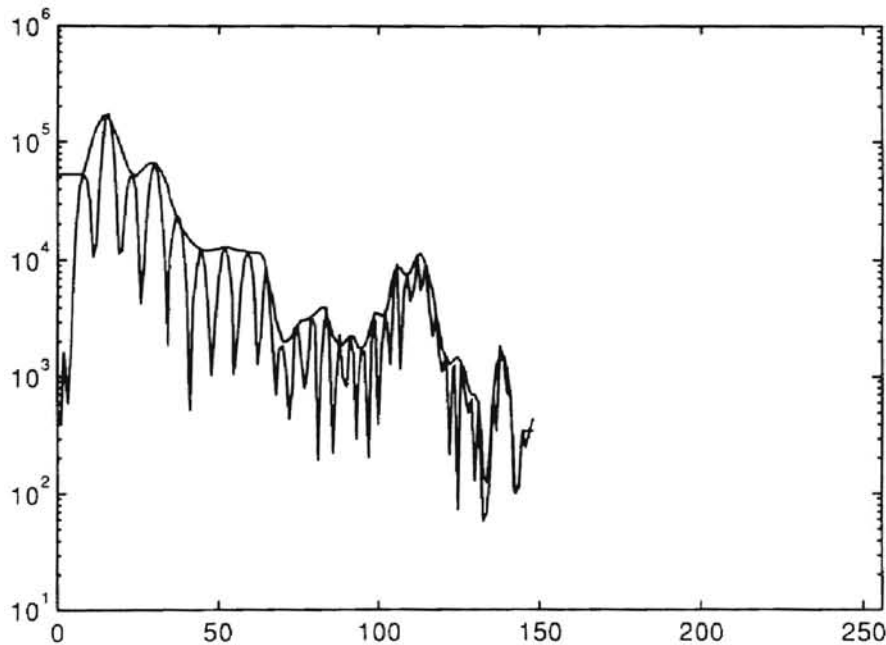


Figure 17. Mel warped spline and spectrum

Special care must be taken when fitting a linear predictive model to a warped spectrum. Referring back to (3.19), $P(\omega)$ is assumed to be spaced equally around the upper half of the unit circle from 0 to π . Before fitting an LP model to this data, the spectrum must be replicated so that it is an even function of frequency. Thus $P(\omega)$ will now range from 0 to 2π on the unit circle, with 0 to π being unique. Warping a spectrum effectively reduces the spacing between DFT coefficients and decreases the length of the replicated spectrum. If an LP model is blindly fitted to the warped spectrum the resulting fit will be incorrect. Instead the LP model must be adjusted to follow the new frequency scale. As an example, for the unwarped spectrum $N=512$ (assuming a 512 point DFT), while the warped spectrum has $N=298$. Properly replicating the spectrum and adjusting the sampling, N' , will allow the spectrum to be modeled by LP. These operations are shown below in (4.15-4.17), where N' is obtained from (4.14).

$$S_{sp}(\hat{k}) = S_{sp}(N' - \hat{k}) \quad \frac{N'}{2} \leq \hat{k} < N' \quad (4.15)$$

$$R_i = \frac{1}{N'} \sum_{k=0}^{\frac{N'}{2}-1} S_{sp}^2(\hat{k}) \cos\left(\frac{2\pi k i}{N'}\right) \quad 0 \leq i \leq P \quad (4.16)$$

$$\sum_{k=1}^P \alpha_k R_{|n-k|} = -R_n \quad 1 \leq n \leq P \quad (4.17)$$

Figure 18 below illustrates the effect of fitting an LP model to a warped spectrum. The effect of unwarping the LP model back to the original frequency axis is shown in Figure 19 (dashed curve), along with the corresponding spectral fit obtained through LP without the use of spectral warping (solid curve).

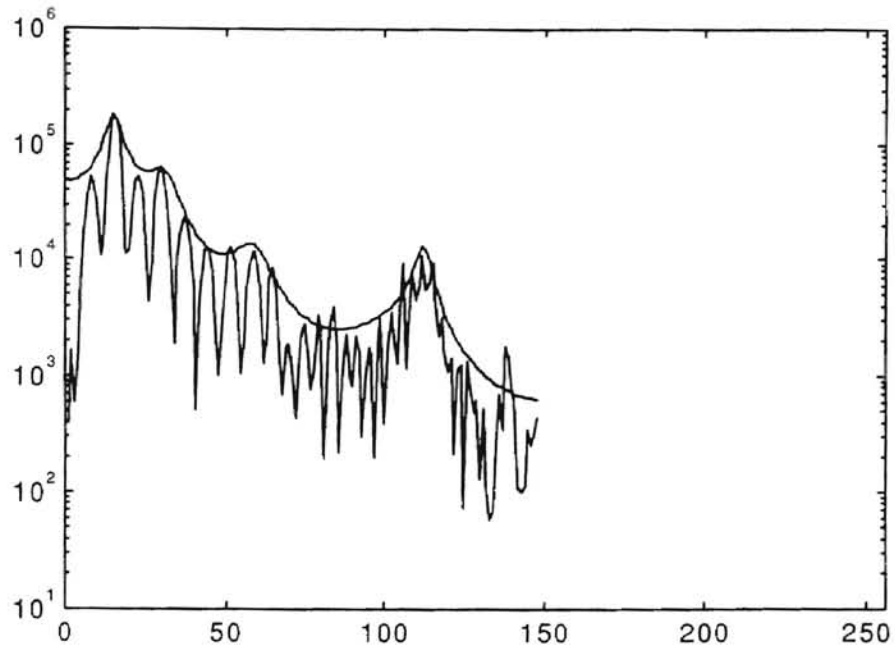


Figure 18. LPC fit to warped spectrum

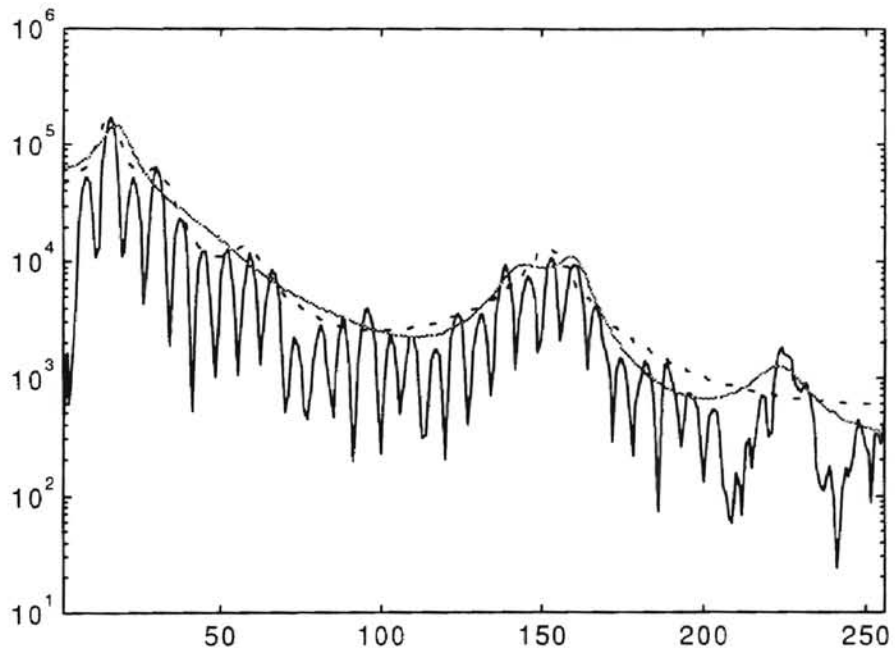


Figure 19. Warped LPC vs. Non-Warped LPC

As can be seen from Figure 19, warping the spectrum prior to fitting a LP model improves the match at lower frequencies, while sacrificing it at higher frequencies. The warping operation, in effect, increases the pole density at the low end of the spectrum while decreasing it at the high end. Perceptually, an accurate match at the lower end of the spectrum, where the energy is greatest, is more desirable than an even match across the entire spectrum. Chapter 6 provides a detailed discussion of the results of spline preprocessing as well as spectral warping.

In this chapter, two techniques for improving the spectral representation of the harmonic amplitudes through linear predictive analysis have been presented. The stated goal of a spectral model is to obtain an accurate representation of the harmonic amplitudes for voiced speech and an average fit for unvoiced speech. An interesting question arises as to whether an accurate fit in a signal-to-noise ratio sense results in the best perceptual quality of the resulting speech. This is the question is examined in the following chapter.

CHAPTER V

POSTFILTERING

It is known that while a processed speech signal may be close to the original in a signal-to-noise ratio sense, it may not sound like the original. Postfiltering attempts to alter the signal in such a way that while the signal-to-noise ratio may decrease, the perceived quality will increase. Numerous speech coding algorithms in use today employ some form of postfiltering to improve the synthetic speech quality.

The reasons for this apparent contradiction between signal-to-noise ratio and perceived quality is based upon the theory of auditory masking. Auditory masking suggests that a signal at one frequency may be obscured by a larger signal at a nearby frequency, i.e., it would be masked by it. In the case of speech coders, noise in the spectrum resulting from quantization will be masked in the formant regions, where the energy is relatively high, but will not be masked in the valleys between formants, where the energy is relatively low. The noise in the formant valleys decreases the perceptual quality of the speech. Noise in one region of the spectrum can only be reduced by shifting it into another region of the spectrum. Thus the usual procedure is to shift the noise from the formant valleys to the formants themselves, where they are effectively masked by the larger amplitudes.

A typical filter used to accomplish this is based on the LP synthesis filter, with the poles moved radially toward the origin. This is accomplished by multiplying each predictor coefficient by a fraction that is exponentially weighted. This process is referred to as

bandwidth expansion, since it has the effect of broadening the formants. The form of this filter is given in (5.1) below. As before, N represents the size of the DFT. In the equations that follow it is assumed that k lies in the range $0 \leq k < \frac{N}{2} - 1$.

$$H(k) = \frac{1}{1 - \sum_{n=1}^P \alpha^n a_n e^{\frac{-2\pi nk}{N}}} \quad (5.1)$$

A typical value of 0.8 is used for α . This corresponds to roughly 570 Hz of bandwidth expansion. Figure 20 below illustrates the effects of this type of filtering on a typical spectral envelope for voiced speech. The solid line represents the original spline enhanced LP spectrum and the dashed line represents the frequency response of the combined LP filter and the all pole postfilter, with $\alpha = 0.8$.

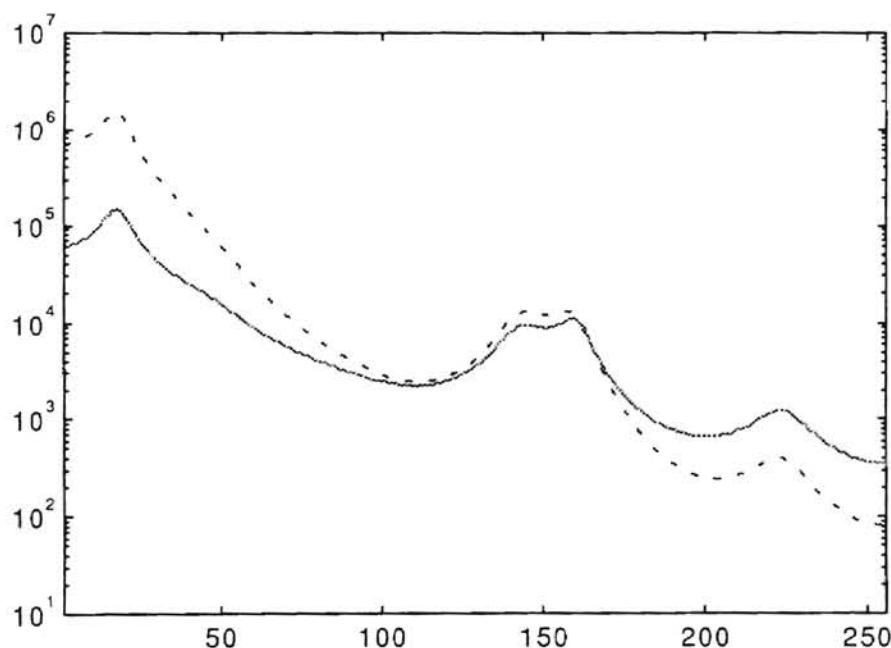


Figure 20. Effect of an all pole postfilter on a voiced spectrum

As can be seen from the above figure, the first formant is amplified, decreasing the amount of perceived noise by increasing the domination of the first formant in the resulting speech. While this does decrease the perceived noise level, the spectral tilt that is induced results in speech that is muffled and heavy.

One solution to this is to use a pole-zero postfilter, such as the one proposed by Chen and Gersho [38]. This filter has the following form

$$H(k) = \frac{1 - \sum_{n=1}^P \alpha^n a_n e^{\frac{-2\pi nk}{N}}}{1 - \sum_{k=1}^P \beta^n a_n e^{\frac{-2\pi nk}{N}}} \left[1 - \mu e^{\frac{-2\pi nk}{N}} \right] \quad (5.2)$$

where α and β determine the amount of filtering, and μ controls the relative brightness of the resulting speech. Typical values for α , β , and μ are 0.5, 0.8, and 0.5, respectively. The denominator represents the traditional all pole postfilter as given in (5.1). The numerator of (5.2) is designed to reduce the spectral tilt introduced by the all pole postfilter. The numerator introduces zeros into the spectrum that have the same phase angles as the poles, but with smaller radii. As can be seen from (5.2) an additional high pass filter is used to further eliminate the low pass characteristics of the postfilter. Figure 21 illustrates the final postfiltered spectrum, with the solid and dashed lines corresponding to the LP spectral envelope and the postfiltered envelope.

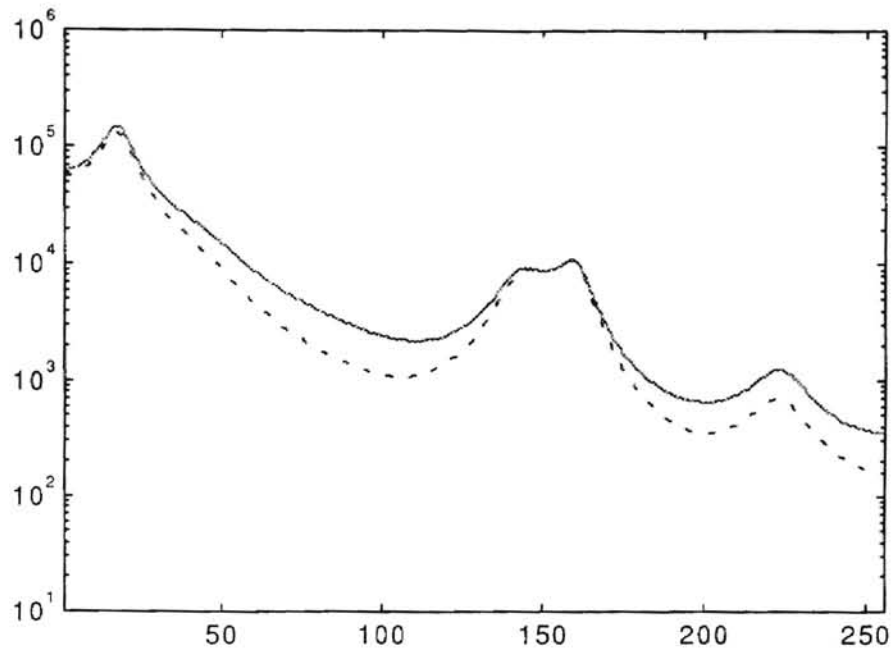


Figure 21. Effect of pole-zero postfilter on voiced spectrum

As can be seen from Figure 21, the postfilter attenuates the regions between the formants, but preserves the formant peaks fairly well. The last formant shows some attenuation, but this is not as perceptually important as the first two formants. The postfilter presented in (5.2) is one of the most popular forms of postfilters in use today. Numerous standards incorporate it or a slight variant of it. These standards include ITU-T G.723 [15], FS1016 [13] and ITU-T G.728 [14].

Previously it was stated that the reason for the use of a postfilter was to reduce the noise introduced through quantization. This is certainly the reason it is used in the three speech coding standards just mentioned. However, in a sinusoidal coder the effects of quantization are not as pertinent, indicating the noise in the spectrum probably results from another source. The apparent noise level may be a result of the initial analysis of the

speech signal. It is reasoned that the side lobe amplitudes introduced by the initial windowing of the speech signal is a primary cause [34]. Most speech coders use a Hamming window to segment the speech signal. The window side lobes are approximately 43 dB below the main lobe of the window. This, in effect, limits the available dynamic range of the input speech signal, resulting in formant valleys that have a higher amplitude than is present in the original speech signal. Therefore, a postfilter, similar to (5.2), could be used to attenuate these regions.

An alternative form of postfiltering is presented in [30] and [34]. These postfilters are both oriented toward sinusoidal coders instead of waveform coders. The postfilter techniques are based on the ideas and observations presented earlier. The general form of this postfilter is shown below in (5.3-5.5).

$$F(k) = \frac{|H(k)|}{|T(k)|} \quad (5.3)$$

$$F_{norm}(k) = \frac{F(k)}{\max[F(k)]} \quad (5.4)$$

$$P_f(k) = [F_{norm}(k)]^\gamma \quad (5.5)$$

In equations (5.3-5.5), $H(k)$ represents the spectral envelope, $T(k)$ a spectral tilt correction function, $F(k)$ the resulting flat spectrum, and $F_{norm}(k)$ the gain normalized flat spectrum. In (5.5), γ takes on values less than 1.0, corresponding to a compression function. The basic methodology in this type of postfiltering approach is to first compute a flattened spectrum shown in (5.3). This spectrum has the normal spectral tilt removed from it. The spectrum is then normalized to have a gain of unity, eq. (5.4), and the compression function is applied, eq. (5.5). Since the flat spectrum has been gain

normalized, the formants should have values close to 1.0, while the valleys between the formants are substantially less. Thus the compression function will reduce these valleys, while minimally altering the formants.

The spectral flattening function, $T(k)$, can be computed using numerous methods. The two primary methods will be discussed here. The method presented in [30] simply uses a bandwidth expanded function, such as the one presented in (5.1), with $\gamma=0.5$, to flatten the spectrum. An alternative method is used in [34] to obtain $T(k)$. In this method, a simple first order predictor is used. The form of this predictor is

$$T(k) = \frac{1}{1 - \rho e^{\frac{-2\pi k}{N}}} \quad (5.6)$$

with ρ representing the first normalized autocorrelation lag as shown below in (5.7).

$$\rho = \frac{\sum_{k=1}^L A(k)^2 \cos(k\omega_0)}{\sum_{k=1}^L A(k)^2} \quad (5.7)$$

In (5.7) $A(k)$ refers to the k^{th} harmonic amplitude, L corresponds to the number of harmonics, and ω_0 represents the fundamental frequency. The compression factor, γ , is chosen experimentally through listening tests to be 0.3. Figure 22 illustrates the effects of this postfilter on a frame of voiced speech. As in the past plots, the solid curve represents the original LP spectral envelope while the dashed curve represents the postfiltered envelope.

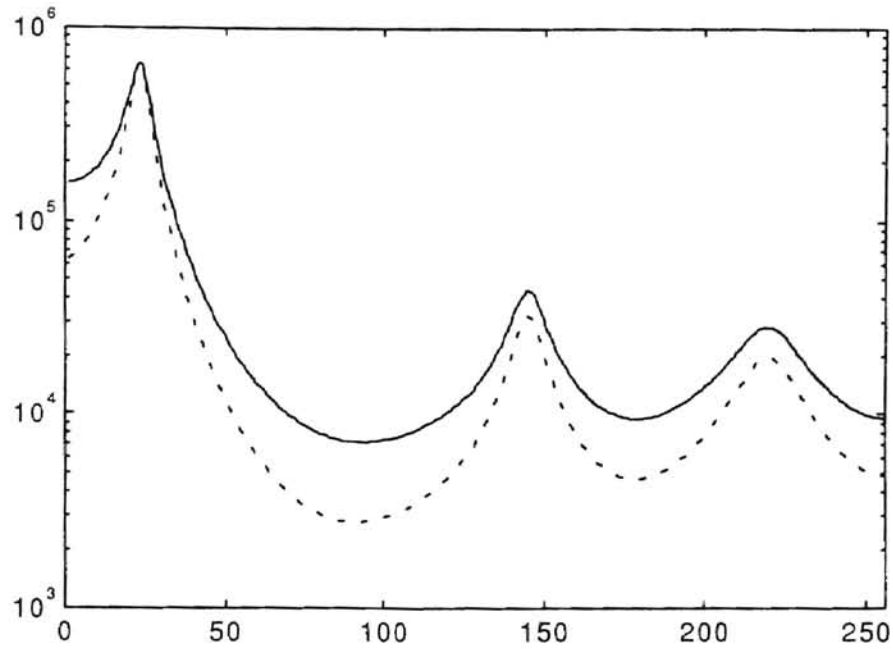


Figure 22. Postfilter for sinusoidal coders

As can be seen from the above figure, the valleys between the formants are significantly attenuated at the expense of the last formant. An adaptive highpass filter can be used to reduce this attenuation. The highpass filter should adapt based upon the spectral tilt present. For voiced speech, this is predominantly low pass in nature, while for unvoiced speech, the spectral tilt has a more highpass nature. Thus the adaption should reduce significantly the contribution of the highpass filter for unvoiced speech. The form of the adaptive highpass filter is shown below in (5.8-5.9), where N corresponds to the length of the DFT, μ is the adaption parameter, and K is a constant weighting factor.

$$H_{hp}(k) = 1 - K\mu e^{-\frac{2\pi k}{N}} \quad 0 \leq k < \frac{N}{2} - 1 \quad (5.8)$$

$$\mu = \frac{\sum_{k=1}^{\frac{N-1}{2}} P_f^2(k) \cos\left(\frac{k4\pi}{N}\right)}{\sum_{k=1}^{\frac{N-1}{2}} P_f^2(k)} \quad (5.9)$$

The adaption coefficient in (5.9) is the first normalized autocorrelation coefficient. For unvoiced speech the function will take on values close to zero, while for voiced speech the function takes on values close to one. The adaption coefficient in (5.8) is weighted by a constant factor, $K=0.2$, to reduce it's effect. Figure 23 illustrates the effects of the adaptive highpass filter, where the dashed curve represents the original postfiltered spectrum, and the solid curve shows the effect of incorporating the adaptive highpass filter. Notice that in this case the amplitude of the highest frequency formant is increased slightly at the expense of the lowest frequency formant.

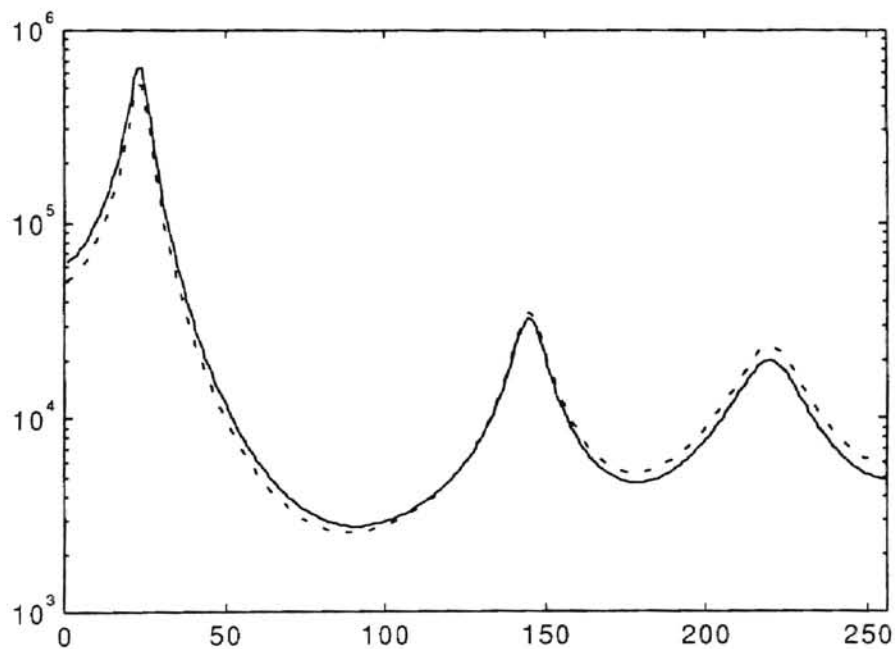


Figure 23. Postfiltering with adaptive highpass filter

The postfilter described by (5.3-5.9) was incorporated into the EMBE speech coder. A brief description of this coder and the results of the inclusion of postfiltering are presented in the next chapter.

CHAPTER VI

RESULTS

The previous chapters have presented several ways to improve the spectral match obtained through linear prediction. As was addressed earlier, the goal of a spectral model for a sinusoidal coder, such as MBE, is to represent accurately the harmonic amplitudes. The results of the various modifications to linear prediction are presented in this chapter. Quantitative results will be presented first followed by qualitative or perceptual results. Additionally, the computational considerations of linear prediction will also be discussed.

To quantify the results of the various modifications to the spectral fit, a suitable error criterion must be chosen. The most popular error measure is the standard signal-to-noise ratio (SNR). Other error measures, however, have been reported to give perceptually more significant results. One of these is the spectral distortion measure (SD) [39]. The form of this error measure is presented below in (6.1) where F_s is the sampling frequency and S_i and P_i represent the original and model spectra for the i^{th} frame.

$$D^2 = \frac{1}{F_s} \int_0^{F_s} \left[10 \log(S_i(f)) - 10 \log(P_i(f)) \right]^2 df \quad (6.1)$$

Since the goal for a harmonic coder is the accurate representation of the harmonic amplitudes, the error criterion presented above should be modified to reflect this requirement. This is accomplished by computing the spectral distortion over only the harmonics, instead of the entire spectrum. To further constrain the error measure, the voicing decisions for the various harmonics, or bands of harmonics in the case of an MBE coder, should be included into the error measure. The reason for this is the fact that

harmonics that are declared unvoiced can be more grossly matched than voiced harmonics. Thus they are not included into the error criterion.

This new spectral distortion measure is shown below in (6.2), where M corresponds to the number of harmonics declared voiced, ω_0 the fundamental frequency, and $S(\omega_0k)$ and $P(\omega_0k)$ are the harmonic magnitudes. It should be noted that $S(\omega_0k)$ and $P(\omega_0k)$ are restricted to the harmonics that are present in voiced bands only.

$$SD^2 = \frac{1}{M} \sum_{k=1}^M [10\log_{10}(S(\omega_0k)) - 10\log_{10}(P(\omega_0k))]^2 \quad (6.2)$$

Now that a suitable error measure has been obtained, the results for the various improvements, alterations, or modifications to the spectral modeling procedure are presented in Table I and Table II.

Table I. Spectral distortion measures for increased LP orders

Item	Spectral Modification	Spectral Distortion (SD) (dB)
1.	10 th order LP model	1.9618
2.	14 th order LP model	1.6607
3.	18 th order LP model	1.5364
4.	22 nd order LP model	1.4199

Table II. Spectral distortion measures for LP improvements

Item	Spectral Modification	Spectral Distortion (SD) (dB)
1.	14 th order LP + spline	1.3902
2.	14 th order LP + spline + compression	1.3514
3.	14 th order LP + spline + compression + warping	1.3910
4.	18 th order LP + spline + compression	1.0169

Before discussion the meaning and significance of the various table entries, the spectral distortion numbers must be put into perspective. A common threshold used in speech coding is that an average spectral distortion of 1 dB is inaudible [40]. This does not mean that differences in errors of less than 1 dB are inaudible, just that 1 dB represents the bottom threshold, below which there appears no difference in perceptual quality. While these numbers do not strictly satisfy the 1 dB requirement for transparent quantization, they do illustrate the effects of the various changes. In fact, for the case of the 18th order spline preprocessed LP model, the results are effectively at the 1 dB threshold.

Table I illustrates the effects of increasing the number of poles that are used to model the spectrum through linear prediction. This topic was discussed in Chapter 2, where it was shown that any spectrum can be arbitrarily matched by increasing the number of poles. This fact is supported by the table entries. An alternative viewpoint follows from the vocal tract model itself. A common assumption is that the human vocal tract is composed of resonances only. This is not entirely correct. Certain phonemes will introduce anti-resonances into the vocal tract, due to the coupling of the nasal cavity.

These anti-resonances will appear in the spectrum as zeros. Thus by increasing the number of poles, the zeros (anti-resonances) can be modeled.

Table II relates the effects of the various proposed enhancements to the LP model. The first entry illustrates the effect of fitting a LP model to a cubic spline envelope representing the harmonic amplitudes. As indicated by the table, the spline fit significantly improves the fit between the LP model and the original spectrum. This improvement allows a 14th order splined LP model to better represent the spectrum than an 22nd order non-splined LP model, in terms of spectral distortion.

The second item in Table II shows the effect of initially logarithmically compressing the harmonic amplitudes prior to the calculation of the spectral model. This concept was discussed in Chapter 3. By comparing items 1 and 2, we can see that initially compressing the harmonics results in a slightly better spectral match, in terms of spectral distortion. It is not known that if the use of other compression functions, such as a square or cube root compressor would provide lower spectral distortion than a logarithmic compressor.

The effect of spectral warping is illustrated by item 3 in Table II. As can be seen from the table, the use of a warping function actually increases the overall spectral distortion. This is to be expected. The warping function maps the spectrum onto a scale that is linear below 1,000 Hz and logarithmic above that, thus some detail at higher frequencies will be lost. This procedure was presented in Chapter 4. It is generally agreed, however, that all frequencies are not perceptually equal. Frequencies with higher energy will tend to mask some of the lower energy frequencies. This concept is the basis for an

entire area of study, known as auditory masking. In the case of our spectral model, these higher energy frequencies correspond to the lower end of the spectrum, especially, the area around the first formant, roughly below 1,300 Hz. To illustrate the performance of spectral warping, the distortion measure presented in (6.2) is recomputed based on two frequencies bands, one from 0 - 1,300 Hz, and the second from 1,300 Hz - 4,000 Hz. These results are presented below in Table III. It should be noted that the specific error measures are normalized based on the number of voiced harmonics in each band, with the combined error measure normalized to the total number of voiced harmonics.

Table III. Spectral distortion for warped frequency scale

Item	Specific Spectral Model	SD below 1,300 Hz (dB)	SD above 1,300 Hz (dB)	Total SD (dB)
1.	14 th order LP, spline, comp.	1.2722	1.4856	1.3514
2.	Item 1. + warping	0.95822	1.7454	1.3910

As can be seen from Table III, the warped spectral model has a lower spectral distortion for the first third of the spectrum, compared to the nonwarped model, and a higher distortion for the rest of the spectrum. It is believed that, perceptually, the lower frequency match is more important than an accurate match at the higher frequency harmonics. It is interesting to note that even the non-warped spectral model performed slightly better in the lower frequency region than in the higher frequency region.

Returning to Table II, the last item corresponds to the use of an 18th order spline preprocessed LP model with initial harmonic compression. As can be seen the resulting spectral distortion for this case is approximately at the 1 dB threshold. However, for low

bit rate coders, such as the 2,400 bps EMBE coder, this higher order model poses certain problems. These problems involve spectral quantization and computation complexities due to the model order.

The problem of spectral quantization is a result of the coding of the LP model for transmission. As mentioned in Chapter 3, LP coefficients are first converted to an alternate representation, Line Spectral Pairs (LSP's), prior to transmission. These are then quantized to a small number of bits for transmission. As an example, the EMBE speech coder uses a vector quantizer to code the 18 LSP's using only 39 bits. The resulting spectral match does however exhibit some degradation due to this coding. This is caused by the limited number of bits used to represent the LSP's. By decreasing the original LP model order to 14th, a better quantization of the spectrum can be obtained with the bits available. Recent experiments have indicated that a 14th order LSP model, vector quantized to 37 bits exhibits significantly less degradation than the 18th order model at 39 bits. An 18th order model, using more bits, would be better suited to a slightly higher bit rate, such as 4,800 bps.

The use of a higher order model, such as 18th order, also poses somewhat of a problem for real time implementations. In a real time implementation of an MBE based speech coder using linear prediction, such as EMBE [4], the computation of the LP parameters represents a significant amount of computation time. By decreasing the model order by 4 coefficients, in turn reduces the time required to calculate the model. This decrease in execution time can be critical in a real time implementation. Based on these

two factors, quantization and computation complexity, the 14th order model was chosen over the 18th order for implementation.

Computational Considerations

In the previous sections we have examined the effects of the various improvements and alterations to the spectral fit obtained by linear prediction. We have not yet addressed any computational considerations of the calculation of the LP model. The model is calculated based on the discussion and equations presented in Chapter 3. It is known that the resulting spectral model may be ill-conditioned, thus leading to model instability. The reason for this ill-conditioning is due to the sometimes large spectral dynamic range that may be present in the speech signal [41].

The use of a white noise correction factor has been proposed in [41] to correct this. By multiplying the 0th autocorrelation coefficient in the model R_0 , by a small delta, in this case 1/256, the ill-conditioning can be avoided. This has the effect of adding white noise to the spectrum 24 dB below the average value of the spectrum. Figure 24 below illustrates this effect.

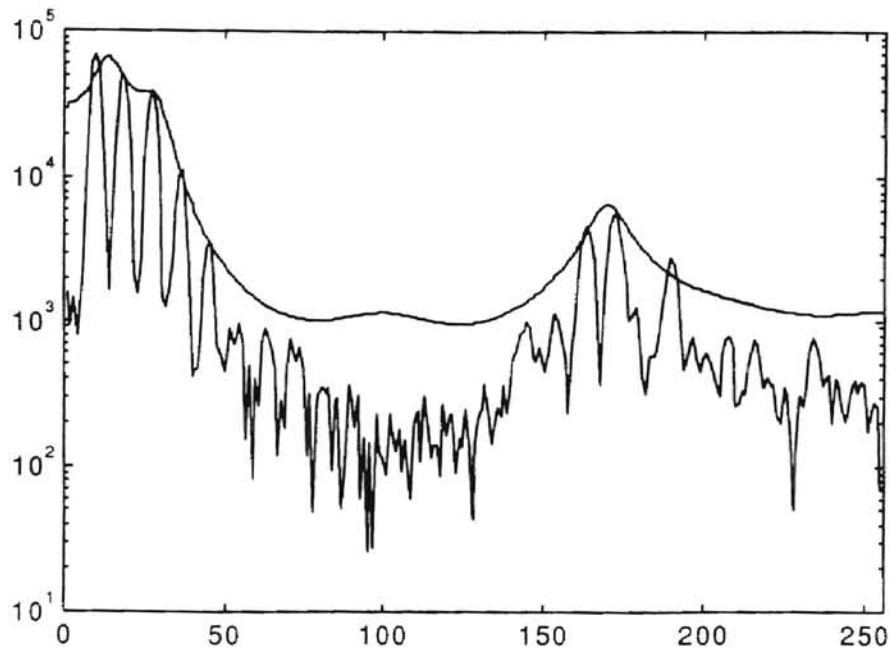


Figure 24. Whitened LP spectral fit (1/256)

As can be seen in Figure 24, the use of a white noise correction factor limits the dynamic range of the speech spectrum. In the case of a correction factor of 1/256, the spectrum is limited to 24 dB below the average value. The use of white noise correction was originally proposed in conjunction with the ITU-T G.728, LD-CELP coder [14]. In this coder, the reduced spectral dynamic range is largely compensated by the excitation sequence. In sinusoidal coders, such as EMBE, the excitation does not compensate for this amount of white noise correction. Thus the resulting speech has an artificial noise signal added to it, introducing a roughness to the synthetic speech signal.

Experiments have shown that this roughness can be eliminated, while still maintaining the stability of the LP model. Reducing the amount of white noise correction to 1/24576, white noise is added to the spectrum approximately 44 dB below the average

value. This appears sufficient to remove the ill-conditioning. Figure 25 below illustrates the same spectrum as Figure 24, but with a white noise correction factor of $1/24576$.

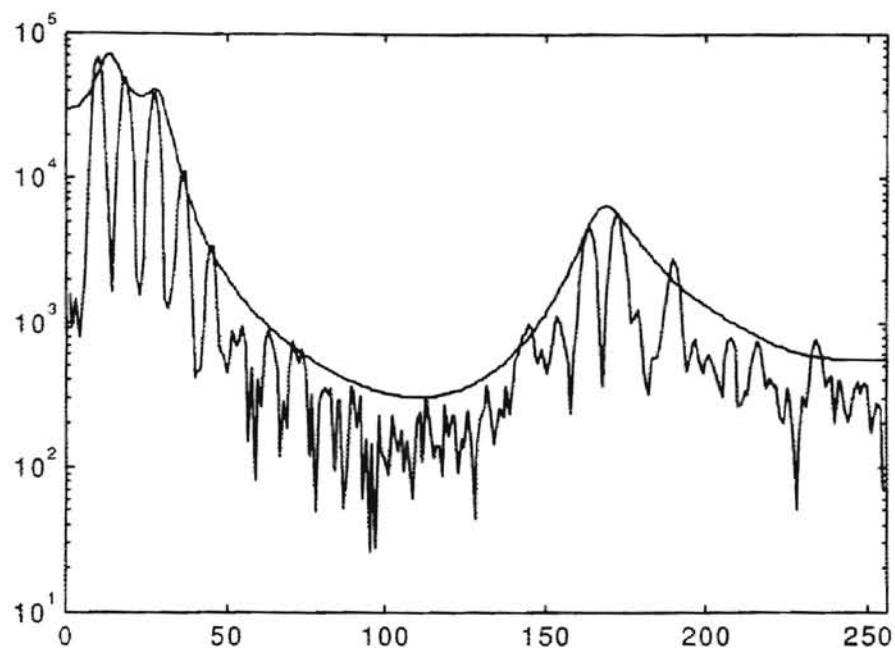


Figure 25. Whitened LP spectral fit ($1/24576$)

Table IV, below illustrates the spectral distortion resulting from the use of a white noise correction factor. All numbers are based on a 14th order LP model, using cubic spline and harmonic compression as a preprocessing stage. As can be seen the correction factor of $(1/256)$ increases the spectral distortion dramatically. The factor of $(1/24576)$ increases the distortion only slightly, while maintaining the model stability.

Table IV. Spectral distortion introduced by white noise correction

Item	Condition	Spectral Distortion (dB)
1.	14th order LP + spline + compression	1.3514
2.	Item 1. + white (1/256)	2.3175
3.	Item 1. + white (1/24576)	1.3637

Perceptual Results

In the previous sections we have examined the effects of the various modifications to the LP model in terms of spectral distortion. In this section the perceptual results will be discussed. To evaluate the perceptual effects, the above modifications were incorporated into the EMBE vocoder. This coder represents an enhanced version of the MBE model, operating at 2,400 bps. An LP model is used to represent the harmonic amplitudes which are vector quantized for transmission. The results obtained are through informal listening tests only.

The first area that was examined involved simply increasing the model order. By increasing the model order, the speech takes on a much brighter more pleasing quality. The most prominent difference appears to be increasing the model order from 10th order to 14th order. An increase from 14th to 18th order produced a more incremental improvement.

The use of spline preprocessing and harmonic compression yields output speech that is smoother and has a 'better' spectral balance than the standard LP model. The resulting speech signal obtained through the preprocessed 14th order LP model is equal to if not slightly higher in quality than the 18th non-preprocessed order LP model. An 18th

order spline preprocessed LP model produced slightly better results than the 14th order preprocessed model. However, due to the implementation considerations discussed above, the 14th order model was finally chosen for incorporation into the EMBE coder.

The addition of spectral warping increased the contribution of the low frequency region of the spectrum, however at the expense of the higher frequency region. The resulting speech signal is slightly heavier in quality, but otherwise indistinguishable from the non-warped spectral model. The use of a white noise correction factor of 1/256, makes the resulting speech noticeably rougher and unpleasant. Reducing the white noise correction to 1/24576, makes the noise relatively inaudible.

Finally, the postfilter described in the previous chapter was incorporated into the EMBE speech coder. The postfilter significantly reduced the overall noise level, resulting in a clearer, more natural sounding speech signal. The muffling effect present in some postfilters was not apparent in this implementation. The inclusion of the adaptive highpass filter slightly improved the overall spectral balance by increasing the amplitude of the higher frequency formants, and slightly attenuating the lowest frequency formant.

CHAPTER VII

CONCLUSION

In this paper we have examined a number of possible improvements to the spectral fit obtained through linear prediction for the case of sinusoidal speech coders. These improvements include: Increased model order, spectral interpolation prior to model calculation, the use of spectral warping to perceptually improve the fit, and certain computational considerations. Additionally, the use of adaptive postfiltering to improve the perceptual quality was also discussed. This chapter serves to briefly summarize and wrap up the work on improving the spectral match.

It was shown that increasing the model order used for linear prediction resulted in an improved spectral match. This improvement was quantified by the use of the spectral distortion error criteria. Perceptually, the effect of increasing the model order resulted in a decrease in the mechanical, reverberant quality of the synthetic speech signal. The most substantial improvement came from increasing the model order from 10th to 14th, with a more incremental improvement resulting from increasing the order from 14th to 18th. A discussion of the gain calculation showed that the traditional gain equation is not entirely accurate for harmonic coders. A gain measure based on the ratio of the original and synthetic spectra, sampled at the harmonics, appears more appropriate.

The use of spectral interpolation as a preprocessing stage for linear prediction was also examined in depth. A cubic spline was used to interpolate between the individual harmonic amplitudes. These amplitudes are initially compressed using a logarithmic

compression rule, so as to reduce their dynamic range. An LP model was then fitted to the spline envelope using traditional techniques. The improvements to the spectral match indicate that a spline preprocessed 14th order LP model can fit the spectrum better than an unprocessed 22nd order LP model. The use of spectral warping to bias the LP model toward perceptually more important regions was also discussed. It was shown that while the overall spectral distortion increased, the distortion over the first third of the spectrum decreased significantly. This resulted in a synthetic speech signal that was slightly heavier, containing more low frequency contributions.

Additionally, computational considerations of the LP model were briefly discussed. The focus centered on the use of prewhitening to properly condition the LP model. Incorporating the traditional level of prewhitening ($1/256$), while ensuring the model did not become ill-conditioned, resulted in a synthetic speech signal that had a noticeable increase in noise level and added harshness. A factor of $1/24576$ appeared to also ensure that the LP model does not become ill-conditioned, while introducing no perceivable distortion into the output speech.

Finally, the use of adaptive postfiltering to improve the perceptual quality of the output speech signal was also presented. It was stated that while postfiltering decreases the synthetic speech's signal-to-noise ratio, the synthetic speech was of higher perceptual quality. This higher quality can be characterized by an overall decrease in coder noise.

Future Research

The results of the work presented in this paper leaves the door open for additional research into a number of related topics. These include the use of preprocessing to enhance the performance of other spectral models, the use of alternative warping functions, and better methods of postfiltering.

The addition of a preprocessing stage for other spectral representations may yield improved performance over that obtained with linear prediction. Cepstral modeling, in particular, is attractive as a method of spectral modeling due to the incorporation of a phase function. This phase function could improve the quality and naturalness of the synthetic speech signal. As was shown in Chapter 3, cepstral modeling alone does not yield a low enough model order for low bit rate coders. Future research could investigate the possibility that a preprocessing stage, such as that presented in Chapter 4, would allow a reduction in model order while maintaining sufficient accuracy.

A second area that would benefit from additional research is the area of spectral warping. The warping function used in Chapter 4, involves warping the frequency axis onto the mel scale. Other perceptual scales exist, such as the bark scale, that may yield a better perceptual match to the spectrum. Additional research should address this issue.

Finally, the area of postfiltering should also be investigated further. In chapter 5, it was indicated that the need for postfiltering was a result of the windowing operation inherent in the analysis phase of the speech coder. Additional work has not been able to confirm this idea. It is possible that a more detailed investigation into why postfiltering is

beneficial for sinusoidal coders could result in an improved method of postfiltering, or even alterations to the underlying model.

REFERENCES

- [1] M. Kohler, L. Supplee, T. Tremain, "Progress towards a new government standard 2400 bps voice coder," *Proc. ICASSP 95*, pp. I488-I491, 1996.
- [2] APCO, "NASTD Federal Project 25 Vocoder: Version 1.0," Dec. 1992.
- [3] Inmarsat Satellite Communications Services, "Inmarsat-M System Definition, Issue 3.0-Module 1: System Description," Nov.1991.
- [4] K. Teague, W. Andrews and B. Walls, "Harmonic Speech Coding at 2,400 bps", *10th Annual Mid-America Symposium on Emerging Computer Technologies*, 1996.
- [5] H. Andrews, "Speech Processing," *Computer*, pp. 315-324, Oct. 1984.
- [6] L. Rabiner, "Applications of Voice Processing to Telecommunications," *Proc. IEEE*, vol. 82, no. 2, pp. 315-324, Feb. 1994.
- [7] B. Atal and L. Rabiner, "Speech Research Directions," *AT&T Technical Journal*, vol 65, issue 5, pp. 75-88, Sep./Oct. 1986.
- [8] CCITT Red Book. *Recommendation G. 711*, volume 3.
- [9] J. Flanagan, M. Schroeder, B. Atal, R. Crochiere, N. Jayant, J. Tribolet, "Speech Coding", *IEEE Trans. on Communications*, vol. com-27, no. 4, pp. 710-736, 1979.
- [10] CCITT Yellow Book. *Recommendation G.721*, volume 3.
- [11] B. Atal and J. Remde, "A new model of LPC excitation for producing natural-sounding speech at low bit rates," *Proc. ICASSP 82*, pp. 614-617, 1982.
- [12] M. Schroeder and B. Atal, "Code-Excited Linear Prediction (CELP): High-quality sounding speech at low bit rates," *Proc. ICASSP 85*, pp. 25.1.1-25.1.4, 1985.
- [13] R. Fenichel, "Federal Standard 1016, Telecommunications: Analog to Digital Conversion of Radio Voice by 4,800 bit/second Code Excited Linear Prediction (CELP)," *National Communications System, Office of Technology and Standards*, Washington, DC, 14. Feb. 1991.
- [14] "Coding of speech at 16 kbit/s using low-delay code excited linear prediction," *ITU-T Telecommunication standard, G.728*, Sep. 1992.
- [15] "Dual rate speech coder for multimedia communications transmitting at 5.3 & 6.3 kbit/s," *ITU-T Telecommunication standard, Draft G.723*, Oct. 17, 1995.

- [16] R. Salami, et. al., "Description of the Proposed ITU-T 8 KB/S Speech Coding Standard," *IEEE Speech Coding Workshop*, Annapolis, 1995.
- [17] A. Gersho, "Advances in Speech and Audio Compression," *Proc. IEEE*, vol. 82, no. 6, pp. 900-918, Jun. 1994.
- [18] L. Rabiner and R. Schafer, *Digital Processing of Speech Signals*, Prentice Hall, New Jersey, 1978.
- [19] J. Flanagan and R. Golden, "Phase Vocoder," *Bell System Technical Journal*, vol. 45, pp. 1493-1509, 1966.
- [20] T. Tremain, "The Government Standard Linear Predictive Coding Algorithm: LPC-10," *Speech Technology Magazine*, Apr. 1982.
- [21] R. McAulay and T. Quatieri, "Speech Analysis/Synthesis Based on a Sinusoidal Representation," *IEEE ASSP*, vol. ASSP-34, no. 4, pp.744-754, 1986.
- [22] D. Griffin and J. Lim, "Multiband Excitation Vocoder," *IEEE ASSP*, vol. 36, no. 8, pp. 1223-1235, 1988.
- [23] J. Hardwick and J. Lim, "A 4.8 KBPS Multi-Band Excitation Speech Coder," *Proc. ICASSP 88*, pp. 374-377, 1988.
- [24] A. Oppenheim and R. Schafer, *Discrete-Time Signal Processing*, pp. 768-825, Prentice Hall, New Jersey, 1989.
- [25] J. Deller, J.R, J. Proakis, J. Hansen, *Discrete-Time Processing of Speech Signals*, pp. 380-385, Prentice Hall, New Jersey, 1987.
- [26] R. McAuley and T. Quartieri, "Sine-Wave Phase Coding at Low Data Rates," *Proc. ICASSP 91*, pp. 577-580, 1991.
- [27] J. Makhoul, "Linear Prediction: A Tutorial Review," *IEEE Proc.*, vol. 63, no. 4, pp. 561-580, Apr. 1975.
- [28] J. Makhoul, "Spectral Linear Prediction: Properties and Applications", *IEEE ASSP*, vol. ASSP-23, no. 3, pp. 283-296, Jun. 1975.
- [29] G. Kang and L. Fransen, "Application of Line-Spectrum Pairs to Low-Bit-Rate Speech Encoders", *Proc. ICASSP 85*, pp. 244-247, 1985.
- [30] A. Kondo, *Digital Speech, Coding for Low Bit Rate Communications Systems*, pp. 239-272, John Wiley & Sons Ltd, West Sussex, England, 1994.

- [31] A. El-Jaroudi, "Discrete All-Pole Modeling," *IEEE Transactions on Signal Processing*, vol. 39, no. 2, pp. 411-423, Feb. 1991.
- [32] F. Itakura and S. Saito, "A Statistical Method for Estimation of Speech Spectral Density and Formant Frequencies," *Electronics and Communications in Japan*, vol. 53-a, no. 1, pp. 36-43, 1970.
- [33] E. Singer, R. McAulay, R. Dunn and T. Quartieri, "Low Rate Coding of the Spectral Envelope Using Channel Gains," *Proc. ICASSP 96*, pp. II-769-II-772, 1996.
- [34] R. McAulay and T. Quartieri, "Sinusoidal Coding," in *Speech Coding and Synthesis* (W.B. Kleijn and K. Paliwal, eds.), pp. 148-150, Amsterdam, The Netherlands, Elsevier Science B.V., 1995.
- [35] H. Hermansky, H. Fujisaki and Y. Sato, "Spectral Envelope Sampling and Interpolation in Linear Predictive Analysis of Speech," *Proc. ICASSP 84*, pp. 2.2.1-2.2.4, 1984.
- [36] R. Sedgewick, *Algorithms in C*, Addison-Wesley Publishing Company, Reading, Massachusetts, 1990.
- [37] M. Unser, A. Aldroubi and M. Eden, "B-Spline Signal Processing: Part I - Theory," *IEEE Transactions on Signal Processing*, vol. 41, no. 2, pp. 821-832, Feb. 1993.
- [38] J. Chen and A. Gersho, "Adaptive Postfiltering for Quality Enhancement of Coded Speech," *IEEE Transactions on Speech and Audio Processing*, vol. 3, no. 1, pp. 59-71, Jan. 1995.
- [39] A. Gray, Jr. and J. Markel, "Distance Measures For Speech Processing," *IEEE ASSP*, vol. ASSP-24, no. 5, pp. 380-391, Oct. 1976.
- [40] K. Paliwal and B. Atal, "Efficient Vector Quantization of LPC Parameters at 24 Bits/Frame", *IEEE Transactions on Speech and Audio Processing*, vol. 1, no. 1, pp. 3-14, Jan. 1993.
- [41] J. Chen and R. Cox, "Convergence and Numerical Sensitivity of Backward-Adaptive LPC Predictor," *Personal Communication via Tom Tremain*, 1995.



VITA

Buddy Walls

Candidate for the Degree of

Master of Science

Thesis: ENHANCED SPECTRAL MODELING FOR SINUSOIDAL SPEECH CODERS

Major Field: Electrical Engineering

Biographical:

Personal Data: Born in Enid, Oklahoma, July 2, 1972, the son of Ray and Polly Walls.

Education: Graduated from Garber High School, Garber, Oklahoma in May, 1990; received Bachelor of Science degree in Electrical and Computer Engineering from Oklahoma State University, Stillwater, Oklahoma in May, 1995. Completed the requirements for the Master of Science degree with a major in Electrical Engineering at Oklahoma State University, Stillwater, Oklahoma, in December, 1996.

Experience: Assistant Radiological Safety Officer, Oklahoma State University, May 1993 to December 1994. Research Assistant, Department of electrical and Computer Engineering, Oklahoma State University, January 1995 to present.

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