Representing Voiced Speech Using Prototype Waveform Interpolation for Low-rate Speech Coding

Michael Leong

A thesis submitted to the Faculty of Graduate Studies and
Research in partial fulfillment of the requirements
for the degree of Master of Engineering

Department of Electrical Engineering
McGill University
Montreal, Canada
November 1992

Abstract

In recent years, research in narrow-band digital speech coding has achieved good quality speech coders at low rates of 4.8 to 8.0 kb/s. This thesis examines the method proposed by W.B. Kleijn called prototype waveform interpolation (PWI) for coding the voiced sections of speech efficiently to achieve a coder below 4.8 kb/s while maintaining, even improving, the perceptual quality of current coders.

In examining the PWI method, it was found that although the method generally works very well there are occasional sections of the reconstructed voiced speech where audible distortion can be heard, even when the prototypes are not quantized. The research undertaken in this thesis focuses on the fundamental principles behind modelling voiced speech using PWI instead of focusing on bit allocation for encoding the prototypes. Problems in the PWI method are found that may be have been overlooked as encoding error if full encoding were implemented.

Kleijn uses PWI to represent voiced sections of the excitation signal which is the residual obtained after the removal of short-term redundancies by a linear predictive filter. The problem with this method is that when the PWI reconstructed excitation is passed through the inverse filter to synthesize the speech undesired effects occur due to the time-varying nature of the filter. The reconstructed speech may have undesired envelope variations which result in audible warble.

This thesis proposes an energy fixup to smoothen the synthesized speech envelope when the interpolation procedure fails to provide the smooth linear result that is desired. Further investigation, however, leads to the final proposal in this thesis that PWI should be performed on the clean speech signal instead of the excitation to achieve consistently reliable results for all voiced frames.

Sommaire

La recherce en codage numérique de la parole bande étroite durant les dernières années a produit des codeurs de bonne qualité de parole à tes taux aussi bas que 4.8 à 8.0 kb/s. La présente thèse examine la méthode proposée par W. B. Kleijn appelée interpolation d'un signal prototype (abrégé PWI pour *prototype waveform interpolation*) pour coder les sections voisées de la parole d'une manière efficace, afin de réaliser un codeur à un taux plus bas que 4.8 kb/s tout en maintenant et même améliorant la qualité perceptuelle des codeurs existants.

Après l'examen de la méthode PWI, il fut conclu que malgré le bon rendement de la méthode, il existe occasionnellement des sections de parole voisée reconstruite où des audibles distortions étaient perçues, même sans quantification des prototypes. La recherche accomplie dans cette thèse se concentre sur les principes fondamentaux de la modélisation de la parole voisée au lieu de s'acharner sur l'allocation des bits pour le codage des prototypes. Des problèmes inhérents à la méthode PWI sont repérés qui aurait pu passer pour des erreurs de codages.

Kleijn utilise PWI pour représenter les sections voisées du signal d'excitation, qui en fait est le résiduel obtenu après l'extraction des redondances à courte-terme dans la parole, à l'aide d'un filtre de prédiction linéaire. Le problème détecté est que quand le signal d'excitation reconstruit par PWI est passé par le filtre inverse afin de synthétiser la parole des indésirables effets apparaissent dus au filtrage adaptatif. La parole reconstruite peut exhiber une enveloppe indésirable dont le résultat est un gazouillement audible.

Cette thèse propose une compensation d'énergie pour lisser l'enveloppe de la parole synthétisée quand la procédure d'interpolation ne produit pas le résultat linéaire et lisse qui est désiré. De plus profondes investigations sur le sujet conduisent à la dernière proposition de cette thèse qui stipule que la méthode PWI devrait être appliquée sur le signal de parole original plutot que sur l'excitation, afin de réaliser des résultats consistents et bien fondés pour toutes les fenêtres voisées.

Acknowledgements

I would like to thank my supervisor Dr. Peter Kabal for suggesting the research subject and for his guidance throughout the course of the research. The major portion of the research was conducted at the Institut National de la Recherche Scientifique (INRS) - Télécommunications whose laboratory facilities provided great assistance to my research. The research was funded by a scholarship from the Natural Sciences and Engineering Research Council of Canada (NSERC).

I would like to thank my family and friends for their support. I am also thankful to my friends at McGill and INRS for their companionship and assistance.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In modern communications systems signals are sampled and encoded digitally into binary information streams for transmission through the channel and these information streams are decoded by the receiver to reconstruct the analog signal. Digital signals can be transmitted through a channel with minimal loss in quality by encoding the signal with error protection which aids the receiver in making the decisions.

An analog speech signal for telephone communication is bandlimited below 4 kHz by passing it through a low-pass filter with a 3.4 kHz cutoff frequency and then sampling at an 8 kHz sampling frequency to represent it as an integer stream. For terrestrial telephone networks, simple pulse-coded modulation (PCM) is used to convert the informatic into a 64 kb/s (kilo-bits/sec) binary stream. Despite the heavy user demand, the terrestrial network is able to accommodate the many 64 kb/s signals by installing a sufficient number of transmission lines.

In mobile communications, on the other hand, very low bit rate coders are desired because of the bandwidth constraints. In 1989 an 8 kb/s vector sum excited linear prediction (VSELP) [1] speech coder was established in the IS-54 standard for the North American digital cellular system. A half-rate 4 kb/s coder is also desired to allow the system to accommodate the growing user demand [2]. Also in 1989, a 4.8 kb/s code-excited linear prediction (CELP) speech coder was established in the U.S. federal 1016 standard for the United States Department of Defense [3]. An even

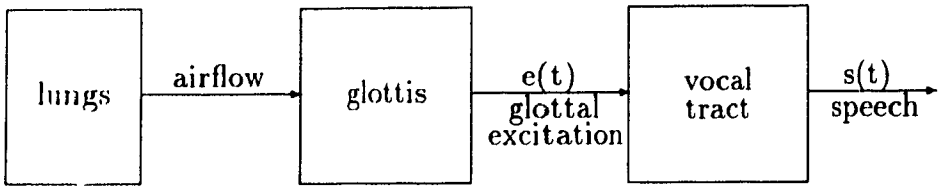


Figure 1.1: Speech production model

lower bit rate is desirable and while the 1016 coder offers highly intelligible speech reproduction it is often unnatural sounding and distorted. The research for a high quality 4 kb/s (or lower) speech coder therefore continues. This thesis investigates a promising method for achieving such a coder.

A coder compresses the source data by removing redundant information. The receiver must know the compression mechanism to decompress the data. Coders intended specifically for speech signals employ schemes which model the characteristics of human speech to achieve an economical representation of the speech.

A simple block diagram of a speech production system is shown in Fig. 1.1. Airflow from the lungs passes through the opening between the vocal folds called the glottis. The size of the opening is varied to constrict the airflow and create turbulence. This wave is called the glottal excitation. It excites the vocal tract which shapes the wave with a spectral envelope known as the formant structure (the formants are the energy peaks in the spectral structure, corresponding to resonances) [4]. Speech coders can take advantage of the formant structure by applying modelling techniques.

Many low rate coders, for example, model the effects of the vocal tract by a linear predictive (LP) filter. The speech signal can be passed through an LP filter to remove the formant structure. This procedure leaves a lower energy residual to be encoded along with the filter parameters. This residual is called the excitation signal for the purposes of this thesis.

The model can be further refined by classifying speech into voiced and unvoiced segments, with the voiced segments having a highly periodic nature such as for vowel

sounds. Voiced speech has a very high level of redundancy which can be exploited using, for example, another LP filter, this time to model the periodic and impulsive waveform produced by the glottis. This LP filter would employ longer delays than the previously discussed filter which employs short delays. It is called the *long-term* LP filter or the *pitch* filter while the other is called the *short-term* LP filter or *formant* filter.

Speech coding methods which employ LP filters are known as linear predictive coding (LPC) methods. CELP has proven to be a particularly effective implementation of LPC coding. Some other schemes which have attempted to effectively encode voiced speech at low bit rates are single-pulse excitation (SPE) [5] and sinusoidal coding [6].

Traditionally, error criteria called distortion measures which compare the reconstructed signal sample-by-sample to the original signal have been used to evaluate coder quality and to optimize the quantization performed in an encoding frame. CELP, single-pulse excitation, and sinusoidal coders all employ such distortion measures. At low bit rates, however, such sample-by-sample signal-to-noise ratio (SNR) measures may not reflect well the perceptual quality of the reconstructed speech, even when so-called perceptual weighting filters are used to shape the error.

A method for coding voiced speech called prototype waveform interpolation (PWI) has been recently proposed in [7, 8, 9, 10] which exploits the periodic nature of a voiced speech frame by extracting and encoding only a prototype representative of one pitch cycle of the waveform for each update frame, instead of encoding every pitch pulse in the frame. A frame of speech is reconstructed by interpolating from one prototype waveform to the next. The interpolation is allowed to proceed naturally to produce a smoothly evolving waveform. The synthesized signal is not forced to maintain sample-by-sample phase synchrony with the original signal. Therefore, a sample-by-sample SNR criterion is not imposed on the synthesized speech frame. Such distortion measures are used for encoding the prototypes only.

The objective of PWI is not only to achieve a lower bit rate over other coders,

but also to achieve higher quality speech reconstruction by preserving fundamental characteristics of voiced speech - its high level of periodicity and continuously time-varying pitch period. Traditional coders are driven to maximize the weighted SNR instead of preserving the characteristics of the speech.

It has been suggested that PWI coding can produce high quality voiced speech at rates below 4 kb/s [8], even as low as 2.6 kb/s [9]. The coder must be integrated with another scheme for encoding the unvoiced segments of the speech. An obvious candidate is the linear predictive scheme CELP used by the 1016 standard.

1.1 Overview of Thesis

This thesis is a study of the newly proposed prototype waveform interpolation scheme proposed for coding speech at low rates below 4 kb/s. Since the current literature on PWI comes only from its originator Kleijn, the goal of this research is to determine some of the details required to effectively implement a PWI coder (details that are not presented in Kleijn's papers), to investigate the theoretical soundness of using PWI, and to suggest solutions or improvements to problems found.

Chapter 2 provides a background on some of the current low-rate speech coding techniques. The emphasis is on linear predictive coding using CELP because it is a current standard and because it is the method chosen for integration with PWI for coding the unvoiced frames of speech. Furthermore, the coding scheme proposed in [7, 8, 9, 10] in fact applies PWI within an LPC framework. The speech is first filtered by the short-term LP filter (the formant filter) and then the PWI method is applied on the excitation signal instead of the input speech. A brief description of the single-pulse excitation and sinusoidal coding methods is also provided in Chapter 2.

In Chapter 3, the PWI method is explained in detail. The representation and interpolation of prototype waveforms using the discrete Fourier transform (DFT) is formalized. The integration of PWI with CELP is also discussed, followed by an

outline of the bit encoding procedure.

In the course of the research it was found that performing PWI on the excitation signal does not always produce the smooth linear interpolation desired in the reconstructed speech. Undesired envelope variations may appear in the reconstructed speech which produce audible warble. Chapter 4 analyzes this problem and it is shown to be caused by the time-varying nature of the short-term LP filter. A post-processor is suggested for reducing the undesired envelope by smoothing out the reconstructed speech amplitudes. However, the final proposal in the thesis is that the prototype extraction and interpolation procedure should be performed on the unfiltered speech in order to achieve a smooth linear reconstruction of the speech.

The conclusion of this thesis is presented in Chapter 5. It summarizes the objectives of PWI for achieving a high quality low-rate speech coder and the improvements suggested in this thesis. Finally, areas for further research are proposed.

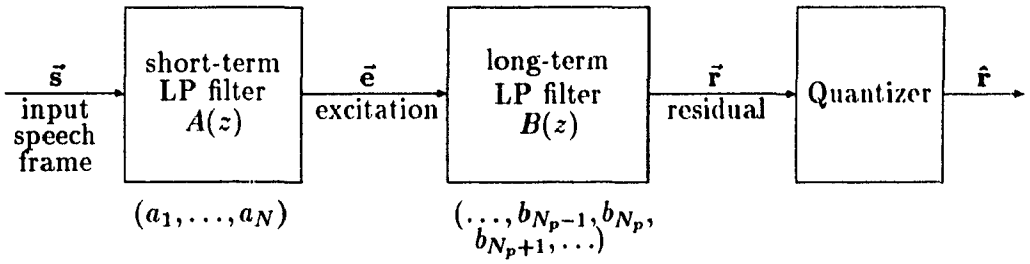
Chapter 2

A Background on Some Low-Rate Speech Coders

Speech coders take advantage of general characteristics that are common to all human speech to achieve efficient encoding. Coding strategies can be viewed from the perspective of removing redundant information or from the perspective of preserving the key information in the signal. Speech coders operating at very low rates must apply these strategies to extremes to achieve such low bit rates. The goal is to achieve reconstructed speech that has little perceptual distortion to the original despite the substantial sample-to-sample error which must exist when using very low coding rates.

The three low-rate speech coders described in this chapter have received the most interest for speech coding in the neighbourhood of 4 kb/s. They are CELP, single-pulse excitation, and sinusoidal coding.

a) encoding



b) resynthesis

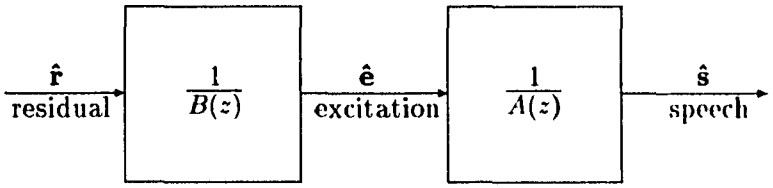


Figure 2.1: LPC coding

2.1 CELP Coding

2.1.1 Linear Predictive Coding of Speech

As was mentioned in the Introduction, speech signals exhibit a structure which can be modelled and parametrized for efficient encoding. The vocal tract produces a spectral envelope in the speech, called the formant structure, which has short-term redundancy. Long-term redundancy exists in the quasi-periodic voiced segments of speech. In linear predictive coding (LPC) these redundancies are removed by employing a short-term and a long-term LP filter, called the formant filter and pitch filter, respectively. Efficient coding is achievable since only a much smaller residual signal needs to be encoded along with the filter parameters. A block diagram of LPC coding is shown in Fig. 2.1a. The speech is reconstructed by passing the quantized residual through the inverse filters corresponding to the long-term and short-term LP filters, as shown in Fig. 2.1b.

The formant filter employs short-term delays of $n = 1, \ldots, N_f$ samples. For low-

rate speech coders a filter order of $N_f = 10$ is commonly used. The speech sample $s(k)$ is predicted by a linear combination of the N_f previous samples as follows,

$$\hat{s}(k) = \sum_{n=1}^{N_f} a_n s(k-n). \tag{2.1}$$

The resulting error signal at this filter output represents the glottal excitation signal and is given by,

$$e(k) = s(k) - \sum_{n=1}^{N_f} a_n s(k-n). \tag{2.2}$$

In the z-domain the formant filter is represented by the transfer function $A(z)$,

$$A(z) = 1 - \sum_{n=1}^{N_f} a_n z^{-n}. \tag{2.3}$$

The pitch filter employs long-term delays of $n = N_p - D, \ldots, N_p, \ldots, N_p + D$ samples. N_p represents the pitch period of the current speech frame to the nearest sample. Pitch filters achieve high coding gains in the voiced segments of speech since the current sample is well predicted by the sample a pitch period before. Normally, one or three filter taps are used ($D = 0$ or 1, respectively). Three filter taps provide for an interpolation between samples to correspond to a non-integer pitch period. When a one-tap filter is employed, acceptable performance is achieved by interpolating the signal to a higher sampling resolution, such as from an 8 kHz to 16 kHz sampling resolution (an interpolation procedure is explained in Section 3.1.1). Such one-tap filters are called high resolution pitch filters. A high resolution pitch period may be needed for some coding schemes such as PWI which are integrated with LPC.

The sample $e(k)$ is predicted by the sample(s) closest to a pitch period previous as follows,

$$\hat{e}(k) = \sum_{n=N_p-D}^{N_p+D} b_n e(k-n) \tag{2.4}$$

and the resultant error signal is the residual:

$$r(k) = e(k) - \sum_{n=N_p-D}^{N_p+D} b_n e(k-n) \tag{2.5}$$

The z-domain transfer function is:

$$B(z) = 1 - \sum_{n=N_p-D}^{N_p+D} b_n z^{-n}. \tag{2.6}$$

The predictor coefficients $\mathbf{a} = (a_1, \ldots, a_{N_f})$ and $\mathbf{b} = (b_{N_p-D}, \ldots, b_{N_p}, \ldots, b_{N_p+D})$ are chosen to minimize the energy of the respective error signals $e(k)$ and $r(k)$ and thus maximize the prediction gain, defined as:

$$G = \sigma_s^2/\sigma_e^2 \tag{2.7}$$

which is the ratio of the variance of the original signal over the variance of the error signal. In the case of the pitch filter the parameter N_p must also be determined.

There are two common approaches for minimizing the error energy, the autocorrelation method and the covariance method (lattice methods [11] are not discussed here). The system for solving the predictor coefficients using the autocorrelation and covariance methods is formulated below first for the formant filter.

Denoting the length of a data frame by N samples, the approach of the autocorrelation method is to limit the speech signal to a finite interval $0 \leq k \leq N-1$. This is accomplished by multiplying the signal by a window $w_s(k)$ which is non-zero only for $0 \leq k \leq N-1$. Typically, the window is chosen to be rectangular or Hamming. The data frame extracted is thus,

$$x(k) = w_s(k)s(k) \tag{2.8}$$

and the energy to minimize is:

$$\varepsilon = \sum_{k=-\infty}^{+\infty} e^2(k) = \sum_{k=-\infty}^{+\infty} \left[x(k) - \sum_{n=1}^{N_f} a_n x(k-n) \right]^2 \tag{2.9}$$

The energy is minimized by taking partial derivatives of the equation above with respect to the parameters a_n, $n = 1, \ldots, N_f$, and setting each of the resulting N_f equations to zero. Noting that $x(k) = 0$ for $k < 0$, the system of equations to solve is

$$\sum_{i=1}^{N_f} a_i \sum_{k=n}^{+\infty} x(k-n)x(k-i) = \sum_{k=n}^{+\infty} x(k)x(k-n), \qquad n = 1, \ldots, N_f. \tag{2.10}$$

Defining the autocorrelation function of $x(k)$ as

$$R(n) = \sum_{k=n}^{N-1} x(k)x(k-n) \tag{2.11}$$

and noting that $R(n) = R(-n)$ then the system of equations can be expressed in matrix form as

$$\begin{bmatrix} R(0) & R(1) & \dots & R(N_f-1) \\ R(1) & R(0) & \dots & R(N_f-2) \\ \vdots & \vdots & & \vdots \\ R(N_f-1) & R(N_f-2) & & R(0) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_{N_f} \end{bmatrix} = \begin{bmatrix} R(1) \\ R(2) \\ \vdots \\ R(N_f) \end{bmatrix} \tag{2.12}$$

The autocorrelation method does not minimize the error energy inside only the desired frame ($k = 0, \dots, N-1$). However, it has the very important property that the resulting formant filter $A(z)$ is minimum phase and consequently the synthesis filter $1/A(z)$ is stable [12].

The approach of the covariance method is to window the error signal and thus minimize the error energy for each frame. The energy to minimize is thus

$$\varepsilon = \sum_{k=-\infty}^{+\infty} w_e(k)e^2(k) \tag{2.13}$$

where $w_e(k)$ is non-zero for only $0 < k < N-1$. For a rectangular window the energy to mimimize is

$$\varepsilon = \sum_{k=0}^{N-1} \left[s(k) - \sum_{n=1}^{N_f} a_n s(k-n) \right]^2 . \tag{2.14}$$

Differentiating with respect to a_n, $n = 1, \dots, N_f$, gives the equations

$$\sum_{i=1}^{N_f} a_i \sum_{k=0}^{N-1} s(k-n)s(k-i) = \sum_{k=0}^{N-1} s(k)s(k-n), \qquad n = 1, \dots, N_f. \tag{2.15}$$

Defining the covariance function of $s(k)$ as

$$\phi(i,j) = \sum_{k=0}^{N-1} s(k-i)s(k-j) \tag{2.16}$$

then the system of equations can be expressed in matrix form as

$$\begin{bmatrix} \phi(1,1) & \phi(1,2) & \dots & \phi(1,N_f) \\ \phi(2,1) & \phi(2,2) & \dots & \phi(2,N_f) \\ \vdots & \vdots & & \vdots \\ \phi(N_f,1) & \phi(N_f,2) & \dots & \phi(N_f,N_f) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_{N_f} \end{bmatrix} = \begin{bmatrix} \phi(0,1) \\ \phi(0,2) \\ \vdots \\ \phi(0,N_f) \end{bmatrix} \tag{2.17}$$

The covariance method minimizes the error energy for each frame, but does not guarantee that the filter $A(z)$ is minimum phase and therefore the synthesis filter $1/A(z)$ may be unstable.

When the autocorrelation and covariance methods are applied to determine the parameters of a pitch filter a system of equations similar to (2.12) or (2.17) is obtained with, of course, the appropriate filter delay values. The parameter N_p, however, must also be determined. The conventional approach is to first determine N_p disjointly from the predictor coefficients. It is assumed that the delay N_p should correspond to the pitch period of the speech and this can be determined by searching for the delay which provides the maximum normalized correlation in the current frame of the input speech signal

$$N_p = \underset{l_{min}<l<l_{max}}{\text{argmax}} \frac{\sum_{k=0}^{N-1} s(k)s(k-\iota)}{\left[\sum_{k=0}^{N-1} s^2(k) \sum_{k=0}^{N-1} s^2(k-l)\right]^{1/2}} \tag{2.18}$$

in which the search is restricted to a range of pitch periods encountered in human speech. A slight variation on (2.18) is to normalize the correlation using the term $\sum_{k=0}^{N-1} s^2(k-l)$ instead of the denominator in (2.18) since this optimization provides maximum prediction for a one-tap pitch filter [13]. It is important to note that for pitch filters the autocorrelation method does not guarantee that $B(z)$ is minimum phase.

Line Spectral Frequencies

When a minimum phase formant filter $A(z)$ is obtained its predictor coefficients must be quantized for data transmission. It is desired that the resulting filter with quan-

tized coefficients maintain the minimum phase property. The method discussed here is to transform the predictor coefficients into line spectral frequencies (LSF's) and perform the quantization in the LSF domain.

Given a minimum phase filter A(z), its corresponding LSF's are defined to be the zeros of the polynomials

$$P(z) = A(z) + z^{-(N+1)}A(z^{-1})$$
$$Q(z) = A(z) - z^{-(N+1)}A(z^{-1}) \tag{2.19}$$

The LSF domain is useful for the quantization of the LPC coefficients because of the following properties [14]:

1. all zeros of $P(z)$ and $Q(z)$ lie on the unit circle

2. the zeros of $P(z)$ and $Q(z)$ are simple and interlaced with each other

If the LSF's are quantized such that the above two properties are maintained, then the quantized LSF's can be used to construct a minimum phase filter (necessary and sufficient conditions for stability are provided in [15]). Methods for solving the roots of $P(z)$ and $Q(z)$ are presented in [16, 17, 18].

In many LPC speech coders the LPC filtering is carried out by interpolating the predictor coefficients between two successive analysis frames into a subframe level such that a smoother transition is achieved. The interpolation is performed in the LSF domain to guarantee the minimum phase property of the resulting filters. The first subframe begins in the middle of the first analysis frame and the last subframe ends in the middle of the second analysis frame. For N_{sub} number of subframes, the subframe LSF's are determined by interpolating the LSF's $\vec{l}_0$ and $\vec{l}_1$ from the analysis frames as follows,

$$\mathbf{l}_{sub}^{(i)} = w_i \mathbf{l}_0 + (1 - w_i)\mathbf{l}_1, \quad i = 0, \ldots, N_{sub} - 1. \tag{2.20}$$

in which w_i are the weighting factors for each subframe. In 1016 CELP subframes of one quarter the size of the update length are interpolated as shown in Fig. 2.2.

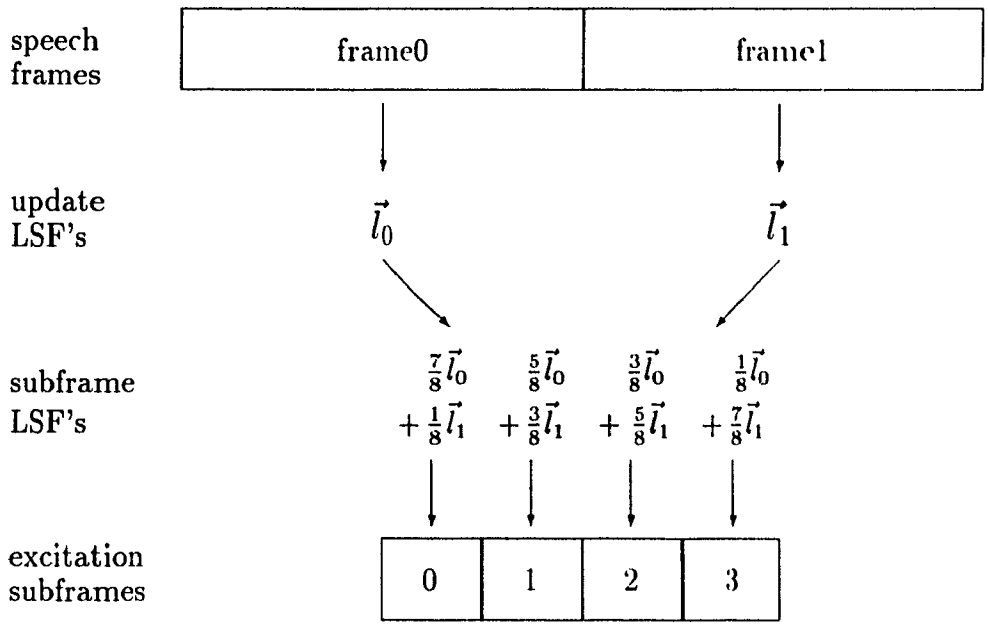


Figure 2.2: LSF subframe structure

2.1.2 Code-Excited Linear Prediction

In Section 2.1.1 the open-loop structure depicted in Fig. 2.1 was presented for LPC coding. The optimal filter coefficients are first determined and then quantized. The filtering operations leave a residual signal to quantize. The problem with an open-loop structure is that since the sensitivity of the system model to the parameters is not known, the coder is unable to determine which of the available quantized values provides the best result. This is especially a problem for low-rate speech coding since coarse quantization can result in significant error.

To overcome the problem with open-loop quantization, the approach of CELP is to work backwards by starting with a candidate quantized residual vector and then pass it through a candidate quantized filter(s) to compare the synthesized result to the original signal based on a distortion measure. The coder therefore performs a closed-loop search over all the available quantized residual vectors, called the codebook, all the available gain factors for the residual vector, and all the available filter parameters to select the best set. This closed-loop approach is called *analysis-by-synthesis*.

The disadvantage of an analysis-by-synthesis approach is, of course, the computational effort required by the exhaustive search. Even for low-rate speech coding in which a very limited set of quantized parameters is available, the computational effort is too great to allow compact real-time implementation.

In CELP coders some of the following simplifications are often made to reduce the search complexity. The predictor coefficients for the formant filter are determined and quantized by open-loop techniques. Another simplification is to perform the search for the optimal pitch filter parameters and residual vector disjointly. The sequential optimization is first performed for the pitch filter by assuming a zero residual vector, after which the residual vector is selected. In an analysis-by-synthesis structure the pitch filter is often referred to as the "adaptive" codebook since the coder selects from a set of candidate vectors which are constructed from previous samples. If desired, the codebook of residual vectors can be orthogonalized to the winning contribution from the adaptive codebook to allow a sequential search which is optimal. A diagram of a CELP coder is shown in Fig. 2.3.

The filter $W(z)$ in Fig. 2.3 is a perceptual error weighting filter which deemphasizes the error near the the formant frequencies since this error is less perceptible to the human ear than the equivalent amount of error away from the formants. The formants are thus said to have a "spectral masking" effect and the error weighting filter allows for the error to be more concentrated near the formants. The error weighting filter is defined as [3],

$$W(z) = \frac{A(z)}{A(z/\gamma)} \tag{2.21}$$

which in effect undoes the operation of the resynthesis filter $1/A(z)$ and then resynthesizes using the filter $1/A(z/\gamma)$. The factor $1/\gamma$ moves the poles of the synthesis filter closer to the origin to deemphasize the formants. Typically, $\gamma = 0.75$ or 0.8 is used. An adaptive postfilter can also be applied on the output speech for further spectral shaping [19].

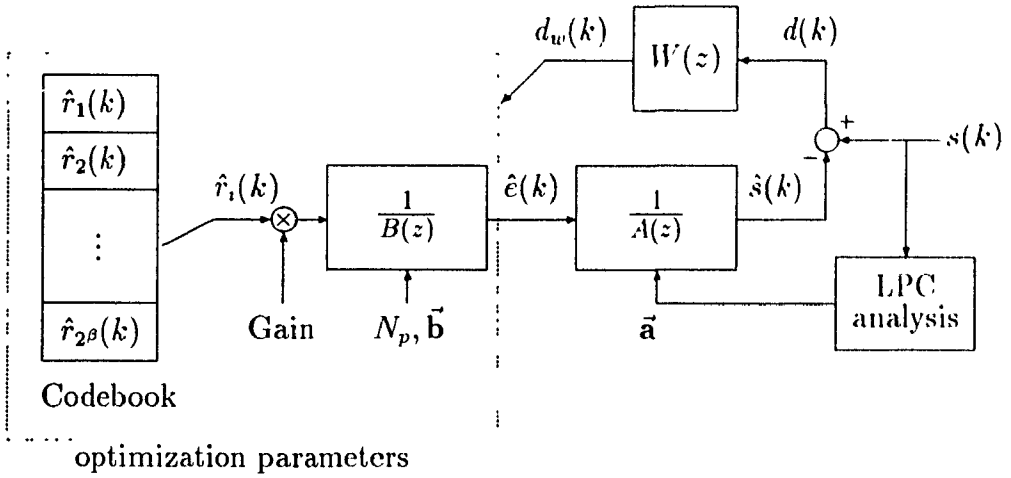


Figure 2.3: CELP coder

2.2 Single-Pulse Excitation

Recently, attempts have been made to encode speech at 4kb/s or lower using the single-pulse excitation (SPE) method [5]. SPE is applied on the voiced segments of speech and CELP is applied on the unvoiced segments. The assumption behind SPE is that for voiced speech the excitation signal is a quasi-periodic impulse train and therefore only one impulse per pitch period needs to be transmitted. For each update frame, the pulse locations and the pulse amplitudes must be encoded along with the predictor coefficients of the LP formant filter. Since SPE is implemented inside an LPC framework and is integrated with CELP, it can be considered to be an LPC coder with a special choice of excitation vectors which can achieve good speech quality at very low bit rates. The CELP coder used in the unvoiced frames does not employ an adaptive codebook, but uses only a fixed stochastic codebook since the pitch filter does not contribute much to the quality in the unvoiced frames. A block diagram of an SPE-CELP coder is shown in Fig. 2.4. An example of a voiced/unvoiced classification scheme is described is Section 3.2.1. A three-bit voiced/unvoiced classification is used in [5] to allow for a transition to occur at a subframe inside the update frame.

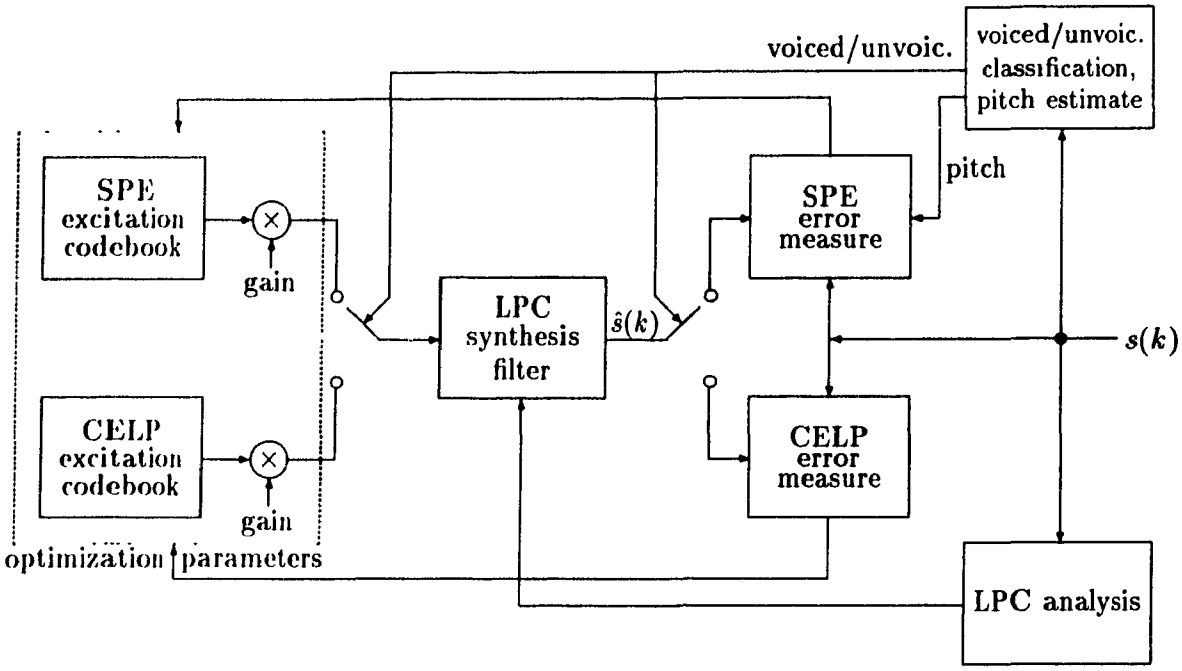


Figure 2.4: Single-pulse excitation coder

The selection of the pulse locations and amplitudes is performed by an analysis-by-synthesis SNR based approach. Candidate excitation signals are passed through the formant synthesis filter and the synthesized speech is compared to the original by computing an error distance measure. The measure in [5] is a cost function which penalizes a low spectrally weighted SNR and also inconsistencies between the current candidate pulses and the previous pulses. The inconsistencies are checked using an estimate of the average pitch period and the pulse amplitudes.

The SPE approach of modelling a voiced excitation signal by a quasi-periodic impulse train is similar to the approach in artificial speech synthesizers of passing an impulse train through a vocal tract model. It is therefore not surprising that SPE can suffer from an artificial buzzy character that is common to similar speech synthesizers. SPE does, nevertheless, provide very intelligible speech reconstruction at very low bit rates.

2.3 Sinusoidal Coding

Another speech coding strategy which has received much investigation for coding at low rates is sinusoidal coding. The motivation behind sinusoidal coding is that because of the periodic nature of voiced speech, the speech can be well represented by a sum of sinusoids. A frame of speech is represented by a limited number of sinusoidal components whose frequencies, amplitudes, and phases must be encoded. A sinusoidal coder attempts to extract the key frequency components in a frame of speech. In harmonic coding approaches, the sinusoids are chosen to be multiples of the pitch period and therefore the frequencies can be encoded efficiently. However, generalized frequency approaches offer the flexibility to allow for the sinusoidal representation of unvoiced speech. For the generalized frequency model the n^{th} speech frame is modelled by [6],

$$s^{(n)}(k) = \sum_{l=1}^{L^{(n)}} A_l^{(n)} e^{j\theta_l^{(n)}} e^{jk\omega_l^{(n)}}. \tag{2.22}$$

The generalized frequency approach is to first compute the short-time Fourier transform (STFT) of a speech frame using a discrete Fourier transform (DFT) or a fast Fourier transform (FFT) as follows,

$$S(m) = \sum_{k=0}^{N-1} w(k)s(k)(e^{j2\pi/N})^{-km} \tag{2.23}$$

in which $w(k)$ is a window. From the STFT an analysis-by-synthesis search is performed to extract a limited number of sinusoids which provide the best speech reconstruction based on an SNR criterion. To reconstruct the speech frame the sinusoid parameters are interpolated from one frame to the next to avoid discontinuities at the frame boundaries. The sinusoids must be matched up using a frame-to-frame peak matching algorithm. Such algorithms must allow for the "birth" and "death" of sinusoids which cannot be matched. The amplitudes of the matched peaks can be linearly interpolated but more sophisticated procedures must be used to interpolate the phase and frequency since the frequency is the derivative of the phase and the phase is represented cyclically, modulo-2π. A block diagram of a sinusoidal coding

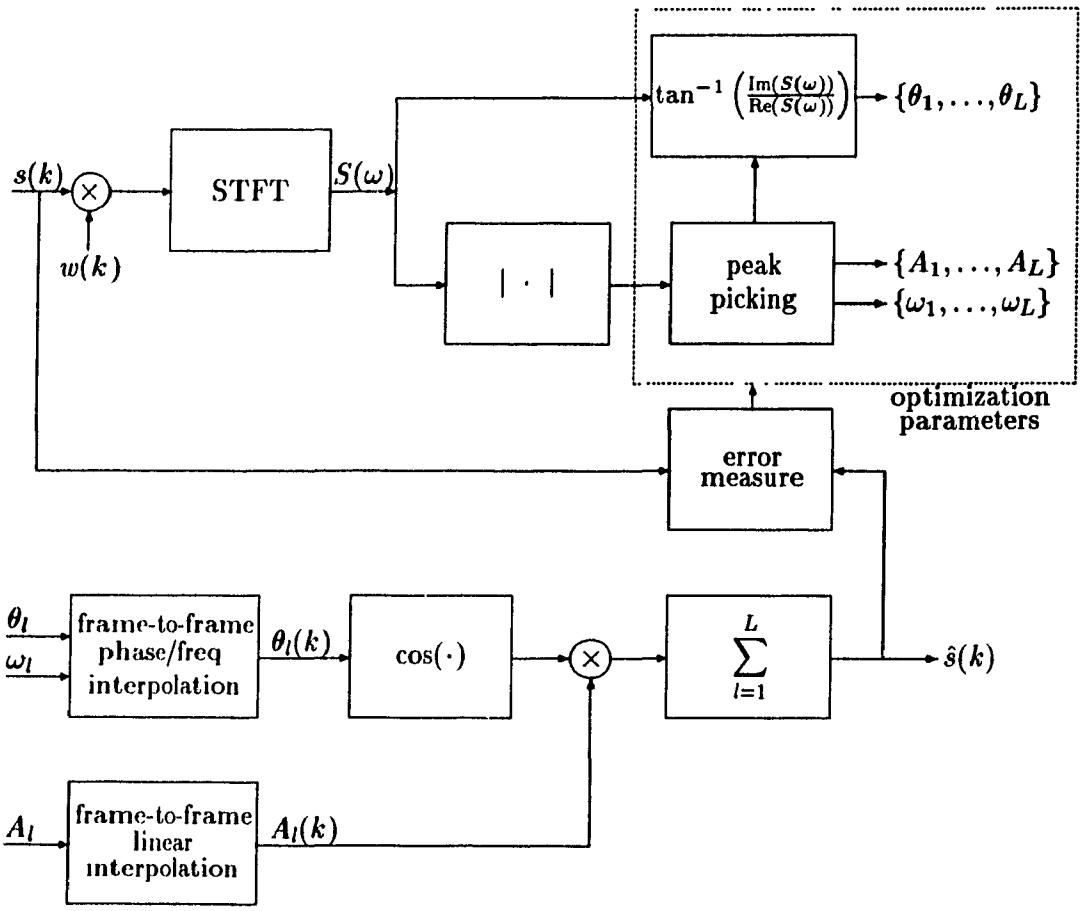


Figure 2.5: Sinusoidal coder

scheme is shown in Fig. 2.5.

Chapter 3

Prototype Waveform Interpolation

Some speech coders which exploit the periodic nature of voiced speech were discussed in the previous chapter. In CELP coding a long-term LP filter is used to reconstruct a sample from the previous pitch pulse. In SPE the excitation signal for voiced speech is represented by only its most important feature - a single impulse per pitch period. Sinusoidal coding encodes voiced speech effectively because of its harmonic nature.

While these three methods certainly exploit the periodic nature of voiced speech, perhaps they do not exploit it well enough. They effectively compress and quantize an N sample update frame but the periodicity of voiced speech suggests that not all N samples need to be encoded. Some of the samples can be generated by interpolation. Furthermore, by coarsely quantizing all N samples of a speech frame according to a sample-by-sample SNR criterion, the level of periodicity of the voiced speech may not be preserved and therefore the reconstructed speech may not have the "voicedness" of the original. One pitch cycle of the reconstructed speech may not be sufficiently consistent in shape with the next and the difference may be audible. For example, Fig. 3.1a shows a segment of voiced speech reconstructed by 4.8 kb/s CELP along with the original in Fig. 3.1b. Some of the pitch cycles in the reconstructed speech show substantial change from the shape of the previous pitch cycle when viewed in

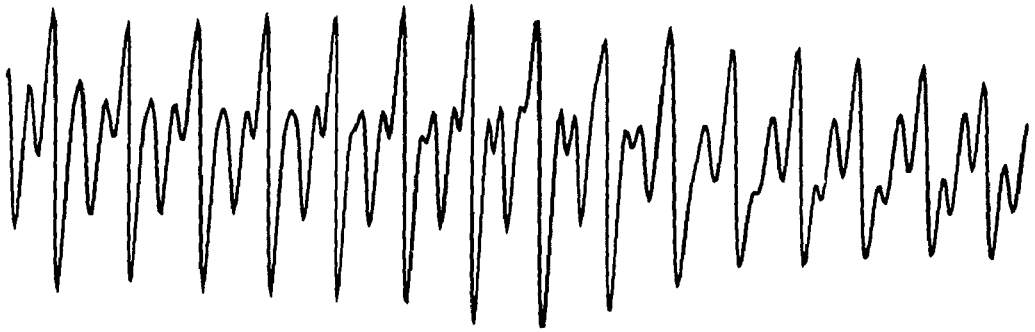
comparison to the smoothly evolving original waveform. Furthermore, the amplitudes of the peak of each pitch cycle are erratic and do not follow the envelope of the original speech.

The recently introduced PWI method for encoding voiced speech offers high quality speech reconstruction at low bit rates. Instead of encoding an entire N sample data frame, the PWI method encodes only a one pitch cycle waveform, called the prototype waveform, per update frame. The signal is reconstructed by interpolating from one prototype to the next. Because the phase of the interpolated signal cannot be maintained sample-by-sample synchronously to the original, an SNR criterion cannot be used. However, by encoding the prototypes accurately, the shape of the speech waveform can be well reproduced and the interpolation procedure allows the waveform to evolve smoothly. This is more important than trying to force the reconstructed speech to maintain sample-by-sample phase synchrony to the original, which is particularly a problem in applying sinusoidal coding on an arbitrary frame length of data. Even in the case when a very low bit rate is used such that the quantized prototypes do not very accurately represent the original prototype shape, PWI can achieve high quality speech since the method allows the level of periodicity in the reconstructed speech to be controlled and therefore the extent of voicing in the original speech can be preserved.

3.1 Representing Voiced Speech Using PWI

The basic steps of PWI are to extract a pitch cycle prototype of the original signal, represent and quantize the prototype using the discrete Fourier transform (DFT), and synthesize the signal using the inverse DFT by interpolating from one prototype to the next. The DFT domain is used since it is a convenient domain for interpolating both the pitch period and the amplitude of the waveforms. The details of these steps is presented in the following subsections. The PWI method introduced in [7, 8, 9, 10] encoded prototypes of the excitation signal and passed the reconstructed excitation

a) original



b) CELP

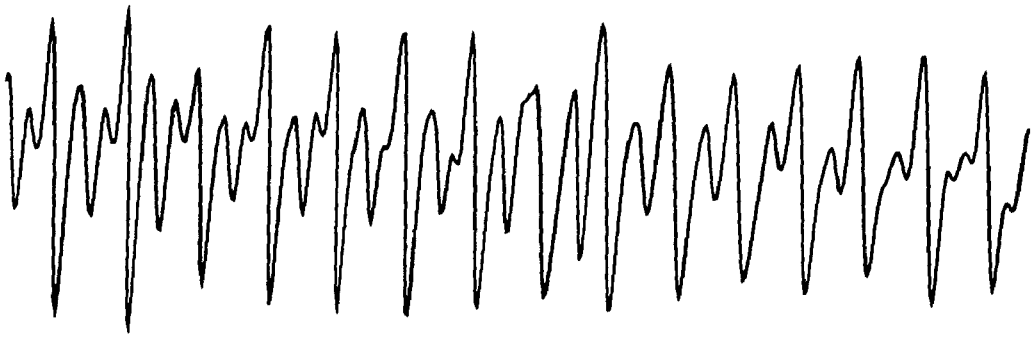


Figure 3.1: Speech reconstructed by 4.8kb/s CELP

signal through a formant resynthesis filter to reconstruct the speech. This approach is presented in this chapter (a discussion on applying PWI in the unfiltered speech domain is presented in Chapter 4).

3.1.1 Prototype Extraction

The PWI method requires that prototype waveforms are extracted which accurately represent one pitch cycle of the signal at each update interval. The accuracy that is needed is to have a good pitch estimate and to have a sufficiently fine sampling resolution to allow for the adjacent prototypes to be well aligned to each other. It is therefore required to have a reliable pitch estimation algorithm and also to double the sampling resolution of the signal from the standard 8 kHz rate to 16 kHz by interpolating in between the samples. A pitch estimation and an upsampling procedure are presented below, beginning first with the upsampling procedure.

Increasing the Sampling Resolution

The temporal resolution of a sampled signal can be increased by interpolating in between the samples. From sampling theory, it is well known that filtering a signal which was sampled according to the Nyquist criterion at a rate R by an ideal lowpass filter with a cutoff frequency $R/2$ recovers the continuous-time signal. In practice, the choice of the interpolation filter depends on the filter delay and the computational cost. The choice here, as in [13, 20], is to construct a less than ideal lowpass filter by windowing the coefficients of an ideal lowpass filter.

A procedure to increase the sampling resolution of a signal $s(k)$ by a factor M is [13]:

1. Insert $(M-1)$ zeros equally spaced in between each adjacent pair of samples from $s(k)$ to produce the signal $s_z(k)$,

$$s_z(k) = \begin{cases} s(k/m) & , \ k/m = \text{integer} \\ 0 & , \ k/m \neq \text{integer} \end{cases} \tag{3.1}$$

2. Filter the signal $s_z(k)$ with a lowpass filter. The ideal filter would be an ideal lowpass filter with a cutoff frequency $R/2$. Now working in a sampled domain of rate $M * R$, the choice here is a filter with coefficients,

$$h(k) = \frac{\sin(\pi k/M)}{\pi k/M} \cdot w_H(k) \tag{3.2}$$

in which $\sin(\pi k/M)/(\pi k/M)$ are the coefficients of an ideal lowpass filter and $w_H(k)$ are the coefficients of a Hamming window, defined by:

$$w_H(k) = \begin{cases} (1-\alpha) - \alpha\cos(2\pi k/L) & , \ |k| \leq L \\ 0 & , \ \text{otherwise} \end{cases} \tag{3.3}$$

in which $\alpha = 0.46$.

The upsampled signal $s_M(k)$ is defined by,

$$s_M(k) = \sum_{l=-L}^{+L} h(l) s_z(k-l) \tag{3.4}$$

The choice of the window length L is a compromise between the "idealness" of the filter response and the filter delay.

Pitch Estimation

A variety pitch estimation procedures are available, some of which are based on locating "pitch markers" (the dominant spike in each pitch cycle of the excitation signal) and some of which are based finding the delay which achieves the maximum auto-correlation in a data frame [21]. A correlation method is described below.

Since the delay which achieves the maximum correlation is the delay which best aligns a signal to its previous samples, then this delay should correspond to the best estimate of the pitch period at the given sampling resolution. For LPC coders, the correlation is computed on an arbitrary frame length since the objective is to maximize the prediction gain on this frame. For PWI, however, the objective is to determine the pitch period at an arbitrary update time instant. If, for example, the pitch period at the beginning of a frame is 70 samples and at the end it is 72 samples, then for

PWI it is desired to know that the pitch period at the beginning of the frame is 70 samples and not to know whether a filter delay of 70, 71, or 72 samples provides the maximum prediction gain over the frame. As in [7] the approach taken here is thus to compute the correlation on a variable window size of double the candidate pitch period, as shown in Fig. 3.2. The pitch estimate at the sampling instant $k = 0$ is therefore,

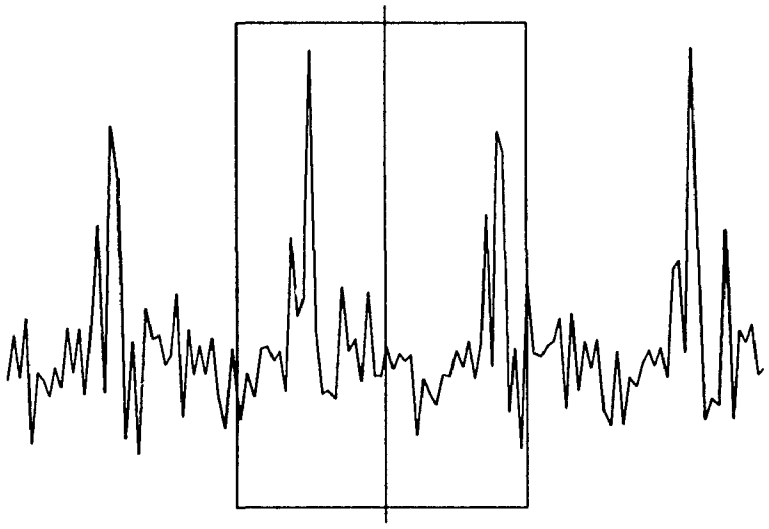
$$p = \underset{l_{min} \le l \le l_{max}}{\text{argmax}} \frac{\sum_{k=0}^{l-1} s(k)s(k-l)}{\left[\sum_{k=0}^{l-1} s^2(k) \sum_{k=0}^{l-1} s^2(k-l)\right]^{1/2}} \tag{3.5}$$

in which the denominator is a normalization factor. Note that in evaluating the correlation coefficient for lag l only samples inside a window of length $2l$ are used.

Letting N_p be the number of samples closest to the true pitch period, the search in (3.5) should produce local maxima in the neighbourhoods of $N_p, 2N_p, 3N_p$ (assuming $N_p, 2N_p, 3N_p \le l_{max}$) and occasionally the global max may occur at $2N_p$ or $3N_p$ instead of N_p. This "pitch doubling" or "pitch tripling" can sometimes occur because of the limited resolution of the sampled signal. For example, a signal with a true pitch period of 70.5 samples may have its global maximum in (3.5) at 141 samples instead of 70 or 71. This phenomenon may also occur because the signal can occasionally exhibit a greater difference between two adjacent pitch cycles than it does when the differences are averaged over several pitch cycles. For a global maximum of p samples the pitch estimation algorithm should also check for local maxima in the neighbourhood of $p/2$ and $p/3$ samples to avoid pitch doubling or tripling. The local maximum should be accepted if the correlation exceeds some threshold factor of the global maximum, such as 90%.

A correlation based pitch estimation algorithm can be applied on the unfiltered speech or on the excitation signal. When the correlation is also used to make a voiced/unvoiced decision it is preferable to determine the correlation sequence on the excitation signal. This is further discussed in Section 3.2.1.

a) candidate window



b) winning window

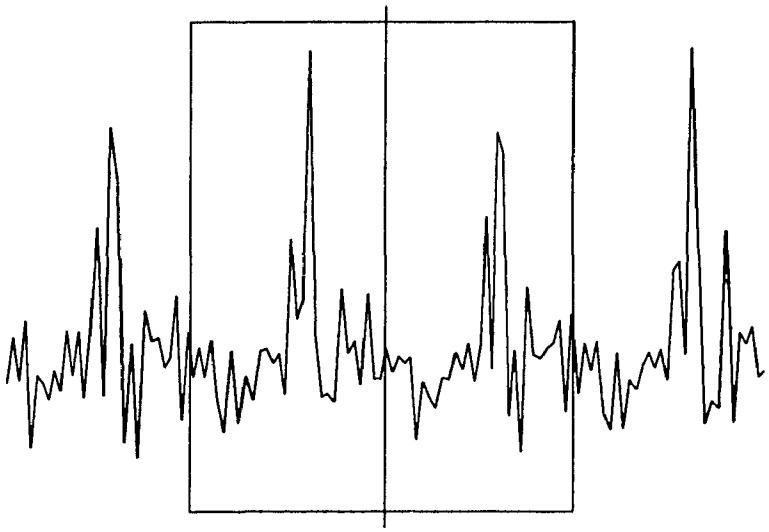


Figure 3.2: Variable pitch analysis window

Remarks on Pitch Extraction for PWI

Experimental results show that using the pitch estimation procedure described above on a speech signal whose temporal resolution is doubled to a 16 kHz rate provides for reliable pitch estimation. Without the algorithm checking for pitch doubling or tripling, the 16 kHz resolution generally reduces the cases of pitch doubling/tripling by 40–50% compared to performing the pitch estimation on an 8kHz sampled signal. When pitch doubling/tripling is checked using a threshold factor of 90% the instances of pitch doubling/tripling are almost eliminated, appearing only a couple of times in each of the female speech sentences tested. Doubling the temporal resolution is therefore a good compromise for PWI between extracting accurate prototype waveforms and the cost of computation and bit encoding.

3.1.2 DFT Representation and Synthesis

The principle of PWI is to synthesize a signal by interpolating from a prototype representative of one pitch cycle of the waveform to the next prototype an arbitrary frame length away. The waveform shape, amplitude, and period must therefore be interpolated. A convenient domain for performing the interpolation is the DFT domain.

In [7, 8, 9, 10] the PWI method is formulated for a continuous-time signal using the Fourier Series (FS) for ease of explanation. In practice, an approximation to the ideal continuous case must be formulated using the DFT. In the following sections the PWI method is first formulated in continuous-time using the FS, followed by a DFT formulation.

Continuous-time Formulation

Letting the continuous-time prototype waveform of the glottal excitation be represented at an update instant by $g(t), 0 \leq t \leq T_g$ where T_g is the pitch period, then

this waveform can be represented by a FS as follows,

$$g(t) = \sum_{m=0}^{M_g} G_c(m)\cos(2\pi mt/T_g) + G_s(m)\sin(2\pi mt/T_g), \quad 0 \le t \le T_g. \tag{3.6}$$

The coefficients $G_c(m)$ and $G_s(m)$ are determined by,

$$\begin{aligned} G_c(m) &= \tfrac{1}{T_g}\int_0^{T_g} g(t)\cos(2\pi mt/T_g)dt \\ G_s(m) &= \tfrac{1}{T_g}\int_0^{T_g} g(t)\sin(2\pi mt/T_g)dt \end{aligned} \tag{3.7}$$

The finite limit M_g is determined by the pitch period and the cutoff frequency. For example, if the signal is bandlimited to 4 kHz and the pitch period is 10 ms (corresponding to a fundamental frequency of 100 Hz) then $M_g = 40$ is needed.

In order to interpolate between the current prototype represented by $G_c(m)$ and $G_s(m)$ and the previous prototype represented by $F_c(m)$ and $F_s(m)$, a phase alignment between the two must be established. The approach is determine the time-shift τ on $g(t)$ which maximizes the cross-correlation between $g(t)$ and $f(t)$. This criterion equivalently finds the shift which minimizes the mean-square error between the two prototypes. It is convenient to perform the alignment in the FS domain since the (possibly) different pitch periods T_f and T_g would require a time-scaling operation in the temporal domain. The FS representation with the greater number of terms should be truncated to the same number of terms as the series with the smaller number of terms to ignore the higher frequency components in the alignment. Therefore, using a series limit of

$$M_{\min} = \min[M_f, M_g] \tag{3.8}$$

the time-shift is determined by the following optimization,

$$\begin{aligned} \tau = \underset{\tau'}{\operatorname{argmax}} \sum_{m=0}^{M_{\min}} & [F_c(m)G_c(m) + F_s(m)G_s(m)]\cos(2\pi m\tau') \\ & +[F_s(m)G_c(m) - F_c(m)G_s(m)]\sin(2\pi m\tau'). \end{aligned} \tag{3.9}$$

It is shown in Appendix A that (3.9) is equivalent to maximizing the cross-correlation between $f(t)$ and $g(t+\tau)$,

$$\tau = \underset{\tau'}{\operatorname{argmax}} \int_{t=0}^{T_p} f(t)g(t+\tau')dt \tag{3.10}$$

in which the waveforms are time-scaled to have the same period T_p (this equivalence is used by Kleijn, but not proved in his work).

A problem with (3.9) is that the optimization may not produce a reliable time-shift τ because of the noisy character of the excitation prototypes. It is therefore useful to put back the formant structure in the waveforms with, if desired, some deemphasis at the formant frequencies. The deemphasis is similar to that applied in the error weighting filter of CELP which deemphasizes the error near the formants to take advantage of the spectral masking effect. In the FS domain, the formant structure can be put back into the prototypes by the following procedure using the LPC coefficients $(\alpha_0, \alpha_1, \alpha_2, \ldots, \alpha_{N_f}) = (1, -a_1, -a_2, \ldots, -a_{N_f})$,

$$V_c(m) = \frac{G_c(m)\sum_{n=0}^{N_f}\gamma^n\alpha_n\cos(\frac{2\pi mnT}{T_g}) + G_s(m)\sum_{n=0}^{N_f}\gamma^n\alpha_n\sin(\frac{2\pi mnT}{T_g})}{\left[\sum_{n=0}^{N_f}\gamma^n\alpha_n\cos(\frac{2\pi mnT}{T_g})\right]^2 + \left[\sum_{n=0}^{N_f}\gamma^n\alpha_n\sin(\frac{2\pi mnT}{T_g})\right]^2}$$

$$V_s(m) = \frac{-G_c(m)\sum_{n=0}^{N_f}\gamma^n\alpha_n\sin(\frac{2\pi mnT}{T_g}) + G_s(m)\sum_{n=0}^{N_f}\gamma^n\alpha_n\cos(\frac{2\pi mnT}{T_g})}{\left[\sum_{n=0}^{N_f}\gamma^n\alpha_n\cos(\frac{2\pi mnT}{T_g})\right]^2 + \left[\sum_{n=0}^{N_f}\gamma^n\alpha_n\sin(\frac{2\pi mnT}{T_g})\right]^2} \tag{3.11}$$

in which T is the sampling interval used in the LPC analysis. If $\gamma = 1$ then $V_c(m)$ and $V_s(m)$ would represent the FS of the prototype in the unfiltered speech domain. With $\gamma < 1$ (a factor of 0.75 or 0.8 is typically used) the formants are deemphasized. $V_c(m)$ and $V_s(m)$ can therefore be considered to be a spectrally weighted FS of the excitation. The spectral weighting of (3.11) is explained in more detail in Appendix B. Note that (3.11) is exact for recovering the speech domain prototype only for a periodic continuation of the prototype. In practice, the filter memory values are previous values of the actual speech which are not a periodic continuation of the speech domain prototype. The weighting gives a relatively short effective filter memory so that with a quasi-stationary assumption, this procedure in the frequency domain gives results equivalent to a time-domain filtering operation.

voiced	time-shift (samples)		
frame	measured	no sp.wt.	sp. weight
1	42	34	42
2	11	52	11
3	56	0	57
4	29	52	29
5	2	75	1

Table 3.1: Prototype Alignment – no spectral weighting vs. spectral weighting compared to the time-shifts measured by visually examining the waveforms

Denoting $V_c(m)$ and $V_s(m)$ as the spectrally weighted FS for $g(t)$ and $U_c(m)$ and $U_s(m)$ for $f(t)$, then these coefficients should be used in (3.9) instead of $G_c(m)$ and $G_s(m)$ and $F_c(m)$ and $F_s(m)$ to determine the time-shift τ, i.e.

$$\tau = \underset{\tau'}{\text{argmax}} \sum_{m=0}^{M_{min}} [U_c(m)V_c(m) + U_s(m)V_s(m)]\cos(2\pi m\tau') + [U_s(m)V_c(m) - U_c(m)V_s(m)]\sin(2\pi m\tau'). \tag{3.12}$$

Experimentation has confirmed the need to perform the cross-correlation analysis on the spectrally weighted FS and not directly on the FS representation of the excitation waveforms. At an upsampled resolution of 16 kHz the results in Table 3.1 were obtained on a sample voiced excitation waveform. Performing the cross-correlation analysis of (3.9) on the unweighted FS (actually a DFT in practice – see the following section for more details) clearly gave inconsistent results while performing the analysis of (3.12) on the spectrally weighted FS produced time-shifts that are very close to the measured ones.

Once τ is determined, the time-shift $g'(t) = g(t+\tau)$ is performed in the FS domain as follows,

$$\begin{aligned} G'_c(m) &= G_c(m)cos(2\pi m\tau) - G_s(m)sin(2\pi m\tau) \\ G'_s(m) &= G_c(m)sin(2\pi m\tau) + G_s(m)cos(2\pi m\tau). \end{aligned} \tag{3.13}$$

With the current prototype aligned to the previous one, prototype waveform interpolation can be performed to synthesize an update frame. The inconsistency between the number of FS terms M_f and M_g is handled in this case by padding the FS representation having the lower number of terms with zero amplitude coefficients (this differs from the previous procedure of removing the higher harmonics for alignment purposes). By padding with zeros the waveform can thus be warped to lengthen its period while preserving its harmonic structure. The number of FS coefficients to use is therefore,

$$M_{\max} = \max[M_f, M_g]. \tag{3.14}$$

The prototype waveforms can be interpolated using a function $\beta(t)$ which is monotonically increasing from 0 to 1 over the update interval. For example, if the update interval is T_{update} then $\beta(t)$ can be chosen to be simply the linear slope

$$\beta(t) = \frac{t}{T_{update}}, \quad 0 \le t \le T_{update}. \tag{3.15}$$

The interpolated pitch contour over the update interval is

$$T_p(t) = (1 - \beta(t))T_f + \beta(t)T_g \tag{3.16}$$

and the phase of the interpolated waveform should follow the contour

$$\phi(t) = \phi_0 + \int_0^t \frac{2\pi}{T_p(t')} dt' \tag{3.17}$$

where ϕ_0 is the phase at the end of the previous PWI synthesis frame which is now the starting phase of the current synthesis frame. In the FS representation, each prototype has a normalized period of 2π. A phase $\phi(t)$ corresponds to the phase at $\phi(t)$ for the initial prototype in which ϕ_0 is set to zero. The update frame is thus synthesized by taking an inverse FS transform with continuous interpolation of the pitch period and FS coefficients as follows,

$$\begin{aligned} e(t) = \sum_{m=0}^{M_{\max}} & [(1 - \beta(t))F_c(m) + \beta(t)G'_c(m)] \cos\left(\phi_0 + \int_0^t \frac{2\pi m dt'}{(1-\beta(t'))T_f + \beta(t')T_g}\right) \\ & + [(1 - \beta(t))F_s(m) + \beta(t)G'_s(m)] \sin\left(\phi_0 + \int_0^t \frac{2\pi m dt'}{(1-\beta(t'))T_f + \beta(t')T_g}\right) \end{aligned} \tag{3.18}$$

Occurrences of pitch doubling or tripling are handled by repeating the cycle of the prototype of one true pitch period to match the number of true cycles in the prototype which has a double or triple pitch period. The cycle can be repeated in the FS domain by inserting zeros in between the FS coefficients.

Discrete-time Formulation

In the previous section the PWI method was outlined in continuous-time using the Fourier series. In this section a practical implementation of the PWI method is formulated in discrete-time using the DFT. The discrete-time formulation is an approximation to the continuous-time case.

It is important to note that the DFT is a transform of an integer number L temporal samples to L frequency components for which the fundamental period is LT, an integer times the sampling period. The inverse DFT is meant to transform the DFT coefficients back to the L equally spaced temporal samples. Care must be taken to ensure that the interpolated parameters are properly rounded to correspond to a feasible set of values that can be used for the phase of the inverse DFT since the DFT should not be used to interpolate for samples in between these phases (an interpolation filter, such as the one described in Section 3.1.1, should be employed for such a task).

In the discrete-time domain the prototype waveform is represented by the L_g-point sequence $g(k), 0 \leq k \leq L_g - 1$, where L_g is the integer pitch period. For the PWI method the prototype is represented in the DFT domain by the following DFT coefficients,

$$\begin{aligned} \tilde{G}_c(m) &= \sum_{k=0}^{L_g-1} g(k)\cos(2\pi mk/L_g), \quad 0 \leq m \leq L_g - 1 \\ \tilde{G}_s(m) &= \sum_{k=0}^{L_g-1} g(k)\sin(2\pi mk/L_g), \quad 0 \leq m \leq L_g - 1 \end{aligned} \tag{3.19}$$

Because of the symmetry properties which result when taking a DFT of a real sequence, there are only L_g free coefficients of the $2L_g$ coefficients in (3.19). The

discrete-time domain sequence $g(k)$ can be recovered by taking the inverse DFT,

$$g(k) = \frac{1}{L_g} \sum_{m=0}^{L_g-1} \tilde{G}_c(m)\cos(2\pi mk/L_g) + \tilde{G}_s(m)\sin(2\pi mk/L_g), \quad 0 \le k \le L_g - 1. \quad (3.20)$$

The DFT coefficients $\tilde{G}_c(m), \tilde{G}_s(m)$ must be modified to correspond to a time-alignment of $g(k)$ to the previous prototype $f(k)$. Performing the alignment in the DFT domain for the discrete-time case requires more careful consideration than performing the alignment in the FS domain for the continuous-time case. The inverse DFT of (3.20) is meant to transform the DFT coefficients back to the L_g temporal samples of $g(k)$ at the phases $(2\pi k/L_g)$, $0 \le k \le L_g - 1$. It does not provide for an interpolation at arbitrary phases in between these L_g phases. Therefore, the alignment can be performed in the DFT domain on a restricted set of L_g phases which correspond to time-shifts of an integer number of sampling intervals in the discrete-time domain. This restriction is one of the principal reasons why the temporal signal is upsampled to a 16 kHz resolution to provide a sufficiently fine resolution.

When performing the alignment of $\tilde{G}_c(m), \tilde{G}_s(m)$ to $\tilde{F}_c(m), \tilde{F}_s(m)$ in the DFT domain, if $L_f \le L_g$ it is *not* desirable to truncate the $\tilde{G}_c(m), \tilde{G}_s(m)$ to correspond to a period of L_f samples and perform the alignment using a phase shift of $(2\pi K_\tau/L_f)$, K_τ = integer. Such a phase-shift is valid for a DFT with a period of L_f but not L_g so that the DFT $\tilde{G}_c(m), \tilde{G}_s(m)$ with a period $L_g \ne L_f$ cannot be phase-shifted this amount (unless it is left permanently truncated, i.e. the higher order terms cannot be aligned so they are permanently removed, but this is undesirable). It is therefore preferable to pad $\tilde{F}_c(m), \tilde{F}_s(m)$ with zero coefficients to stretch its period to L_g samples and perform a phase-shift on $\tilde{G}_c(m), \tilde{G}_s(m)$ of $(2\pi K_\tau/L_g)$, $0 \le K_\tau \le L_g - 1$. Note that for the DFT the zero coefficients are inserted in the middle instead of padded on the end because of its symmetrical structure (the high frequency components are located in the middle of the DFT such that the coefficients $\tilde{F}_c(m), \tilde{F}_s(m)$, $m = 1, \ldots L_f - 1$ exhibit even and odd symmetry, respectively, about the middle $L_f/2$). Using a period of L_g samples the optimal phase-shift is thus determined by maximizing the cross-correlation between $\tilde{V}_c(m), \tilde{V}_s(m)$

and $\tilde{U}_c(m), \tilde{U}_s(m)$, which are the spectrally weighted versions of $\tilde{G}_c(m), \tilde{G}_s(m)$ and $\tilde{F}_c(m), \tilde{F}_s(m)$, as follows:

$$K_\tau = \underset{0\le k\le L_g-1}{\operatorname{argmax}} \sum_{m=0}^{L_g-1} [\tilde{U}_c(m)\tilde{V}_c(m) + \tilde{U}_s(m)\tilde{V}_s(m)]\cos(2\pi mk/L_g) + [\tilde{U}_s(m)\tilde{V}_c(m) - \tilde{U}_c(m)\tilde{V}_s(m)]\sin(2\pi mk/L_g). \tag{3.21}$$

Note that if $L_f \le L_g$, the higher order terms in $\tilde{V}_c(m), \tilde{V}_s(m)$ do not affect the computation of the cross-correlation anyway since the corresponding coefficients $\tilde{U}_c(m), \tilde{U}_s(m)$ are zero. The spectral weighting procedure is similar to that in (3.11) with the factor T/T_g replaced by $1/L_g$ for the discrete-time formulation, i.e.

$$\tilde{V}_c(m) = \frac{\tilde{G}_c(m)\sum_{n=0}^{N_f}\gamma^n\alpha_n\cos(2\pi mn/L_g) + \tilde{G}_s(m)\sum_{n=0}^{N_f}\gamma^n\alpha_n\sin(2\pi mn/L_g)}{\left[\sum_{n=0}^{N_f}\gamma^n\alpha_n\cos(2\pi mn/L_g)\right]^2 + \left[\sum_{n=0}^{N_f}\gamma^n\alpha_n\sin(2\pi mn/L_g)\right]^2}$$

$$\tilde{V}_s(m) = \frac{-\tilde{G}_c(m)\sum_{n=0}^{N_f}\gamma^n\alpha_n\sin(2\pi mn/L_g) + \tilde{G}_s(m)\sum_{n=0}^{N_f}\gamma^n\alpha_n\cos(2\pi mn/L_g)}{\left[\sum_{n=0}^{N_f}\gamma^n\alpha_n\cos(2\pi mn/L_g)\right]^2 + \left[\sum_{n=0}^{N_f}\gamma^n\alpha_n\sin(2\pi mn/L_g)\right]^2} \tag{3.22}$$

The phase-shifting of $2\pi K_\tau/L_g$ in the DFT domain, which corresponds to a time-shift of K_τ samples in the discrete-time domain, is performed by modifying the DFT coefficients as follows:

$$\begin{aligned}\tilde{G}'_c(m) &= \tilde{G}_c(m)\cos(2\pi mK_\tau/L_g) - \tilde{G}_s(m)\sin(2\pi mK_\tau/L_g)\\ \tilde{G}'_s(m) &= \tilde{G}_c(m)\sin(2\pi mK_\tau/L_g) + \tilde{G}_s(m)\cos(2\pi mK_\tau/L_g).\end{aligned} \tag{3.23}$$

The DFT coefficients of the aligned prototypes and the pitch period lengths are interpolated to synthesize the update speech frame using the inverse DFT. The interpolation of the parameters is performed at every sampling interval. For an update frame length of N_{update} samples, the interpolating function can be chosen to be,

$$\beta(k) = \frac{k}{N_{update}}, \quad 0 \le k \le N_{update} - 1. \tag{3.24}$$

The interpolated pitch contour is thus,

$$p(k) = (1 - \beta(k))L_f + \beta(k)L_g. \tag{3.25}$$

At each sampling instant the phase contour is updated by,

$$\phi(k) = \phi(k-1) + 2\pi/p(k). \tag{3.26}$$

This is an approximation to the continuous-time domain update at a time instant $t + dt$ by,

$$\phi(t + dt) = \phi(t) + \frac{2\pi}{T_p(t)}dt \tag{3.27}$$

as given by (3.17). Note that $p(k)$ has not yet been rounded to an integer pitch period in (3.26) to avoid additional error in approximating the continuous-time case. The phase in (3.26) cannot, however, be directly used in the inverse DFT since a proper phase should satisfy the condition,

$$\phi'(k) = 2\pi C(k)/P(k) \tag{3.28}$$

where $P(k)$ is an integer pitch period and $C(k)$ is an integer phase index. The pitch period $p(k)$ must therefore be rounded to an integer pitch period $P(k)$,

$$P(k) = \text{int}[p(k) + 0.5] \tag{3.29}$$

and the integer phase index is determined by,

$$C(k) = \text{int}\left[\frac{\phi(k)P(k)}{2\pi} + 0.5\right]. \tag{3.30}$$

Having determined $P(k)$ and $C(k)$ a suitable phase is determined by computing (3.28). The update frame is synthesized using the inverse DFT to generate the samples,

$$\begin{aligned} c(k) = \frac{1}{P(k)} \sum_{m=0}^{P(k)-1} & [(1 - \beta(k)\tilde{F}_c(m) + \beta(k)\tilde{G}'_c(m)] \cos\phi'(k) \\ & + [(1 - \beta(k)\tilde{F}_s(m) + \beta(k)\tilde{G}'_s(m)] \sin\phi'(k), \\ & 0 \le k \le N_{update} - 1 \end{aligned} \tag{3.31}$$

It is important to note that the phase of the synthesized waveform corresponds to a linearly evolving pitch contour (neglecting the rounding error) and no effort is made to force the synthesized waveform maintain sample-by-sample synchrony to the original. An analysis-by-synthesis technique which uses a sample-by-sample SNR criterion is not used and instead the synthesized waveform is simply allowed to evolve smoothly. At the end of the synthesis frame the synthesized signal may slightly lead or lag the original.

An example of an excitation frame reconstructed by PWI is shown in Fig. 3.3b. The prototypes extracted from the original excitation are shown in Fig. 3.3a. The speech frame reconstructed by passing the PWI produced excitation through the formant synthesis filter is shown in Fig. 3.4b along with the original speech in Fig. 3.4a. The speech has been excellently reconstructed.

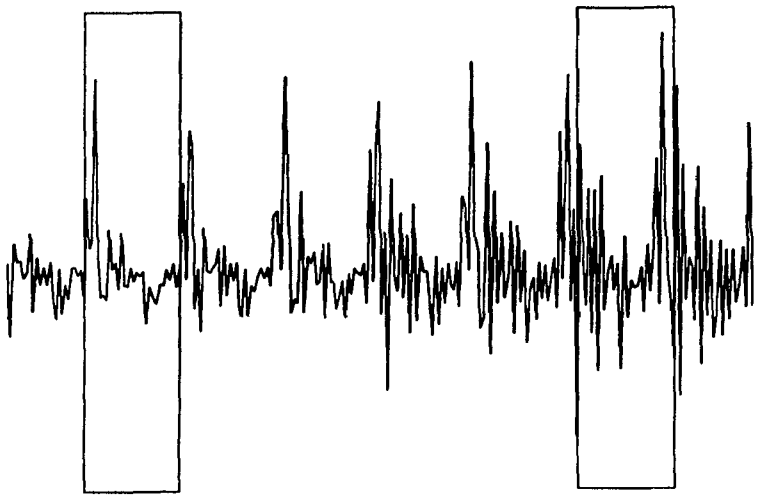
3.2 Integrating PWI with CELP

The PWI method of Section 3.1 is useful for representing only the voiced segments of speech so that another coding method must be used for representing the unvoiced segments. In this section it is shown how PWI can be integrated with CELP. A criterion for deciding between the two methods is needed and the coder must also handle the transition from one method to the other.

3.2.1 PWI/CELP Decision

The PWI method is suitable for encoding only the frames of speech which exhibit sufficient periodicity. The CELP method, on the other hand, is suitable for encoding both unvoiced and voiced segments of speech but it may not encode the voiced segments as well as PWI. The coder must therefore check for sufficient periodicity in the signal to use PWI and default to CELP when PWI is inappropriate. The periodicity of the signal can be measured by computing a suitable correlation value, after which this correlation must be compared to an ad hoc threshold level which determines if

a) prototypes extracted from original excitation



b) reconstructed excitation frame

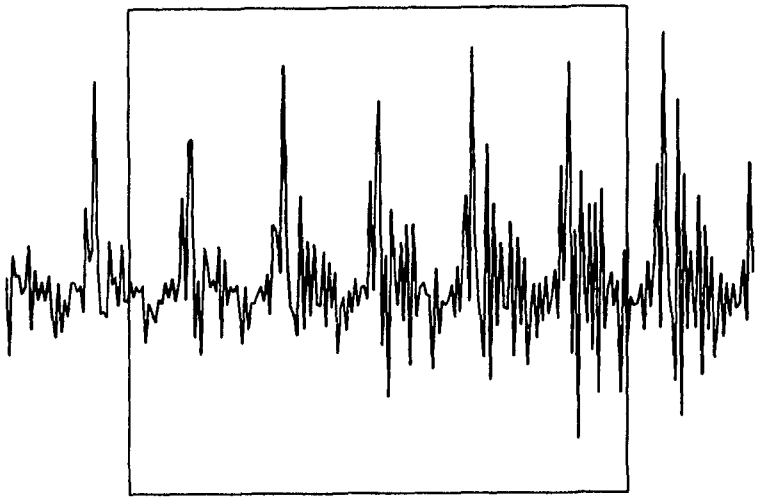
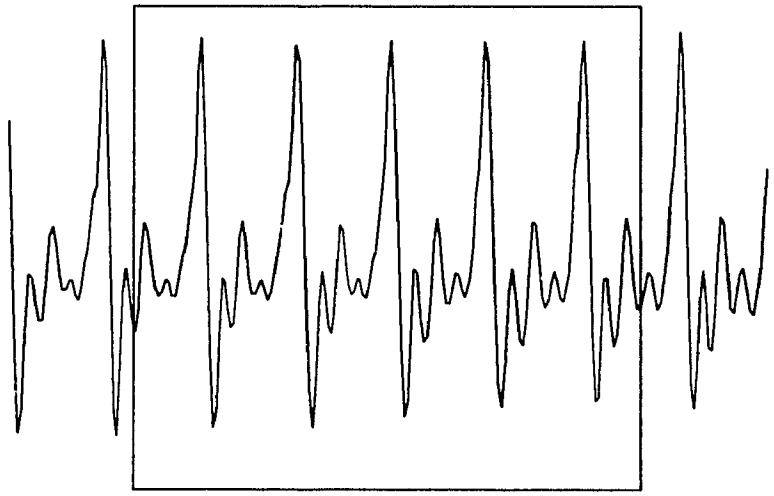


Figure 3.3: PWI reconstructed excitation with unquantized prototypes

a) original frame



b) reconstructed frame

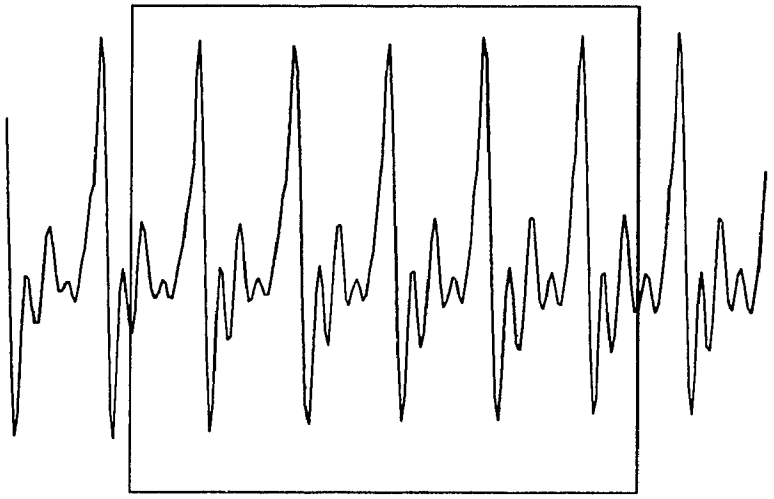


Figure 3.4: PWI reconstructed speech with unquantized excitation prototypes

the level of periodicity is sufficient.

An initial voiced/unvoiced frame classification can be made by computing the normalized correlation in the signal over the update frame of N samples as follows,

$$R_N = \max_{l_{min}<l<l_{max}} \frac{\sum_{k=0}^{N-1} s(k)s(k-l)}{\left[\sum_{k=0}^{N-1} s^2(k) \sum_{k=0}^{N-1} s^2(k-l)\right]^{1/2}} \tag{3.32}$$

which is simply the correlation at the optimal LPC pitch filter delay $l = N_p$ from (2.18). Note, however, that if the frame is classified as a voiced frame from the correlation R_N, then another correlation, this time over a variable length of two pitch periods, must be computed anyway in (3.5) to determine the pitch period. Therefore, the computation in (3.32) can be omitted if desired and the voiced/unvoiced decision can be made using the following correlation,

$$R_p = \max_{l_{min}\le l\le l_{max}} \frac{\sum_{k=0}^{l-1} s(k)s(k-l)}{\left[\sum_{k=0}^{l-1} s^2(k) \sum_{k=0}^{l-1} s^2(k-l)\right]^{1/2}}. \tag{3.33}$$

It has been found from experimentation that it is more reliable to make the voiced/ unvoiced decision on the excitation signal instead of the unfiltered speech since the speech signal can occasionally exhibit very high correlation in an unvoiced frame. It has been found that the value 0.7 serves as a good threshold for determining if there is sufficient periodicity in the excitation signal to classify the frame as voiced.

In [7, 8] it was chosen to use the unfiltered speech for making the voiced/ unvoiced classification instead of the excitation signal as suggested here, however, this difference is not of great importance since in the end the PWI/CELP decision must be made by determining the cross-correlation between the two extracted prototypes. The voiced/unvoiced frame classification serves as preliminary decision to use CELP when the frame is classified as unvoiced. When the frame is classified as voiced the cross-correlation between the two prototypes remains to be computed to determine if there is sufficient periodicity to use PWI. A more effective preliminary decision simply

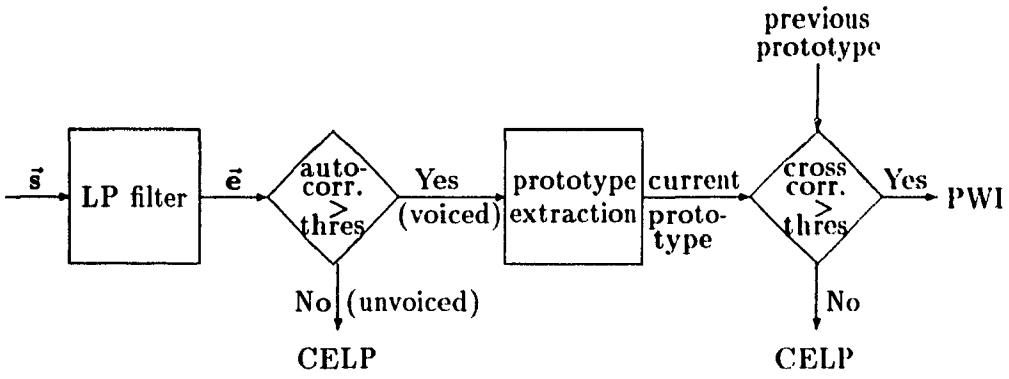


Figure 3.5: PWI/CELP decision

reduces some needless computation in extracting and aligning prototypes when the frames may actually be unvoiced.

As shown in Appendix A, the normalized cross-correlation between the two spectrally weighted excitation prototypes can be computed in the DFT domain by,

$$R_X = \frac{\sum_{m=0}^{L_g-1} \tilde{U}_c(m)\tilde{V}'_c(m) + \tilde{U}_s(m)\tilde{V}'_s(m)}{\left[\sum_{m=0}^{L_g-1} (\tilde{U}_c^2(m) + \tilde{U}_s^2(m)) \sum_{m=0}^{L_g-1} (\tilde{V'}_c^2(m) + \tilde{V'}_s^2(m))\right]^{1/2}} \tag{3.34}$$

in which the numerator has already been computed in (3.21) for performing the prototype alignment.

Choosing the threshold to use on the cross-correlation R_X is a compromise between encoding with the low-bit rate of PWI and avoiding an excessive number of voiced/unvoiced transitions versus the potential distortion that may occur when PWI is applied on a signal that is not highly periodic. It was found that good results are obtained when the cross-correlation of the spectrally weighted excitation prototypes exceeds a threshold value of, again, 0.7. A block diagram of the PWI/CELP decision process is illustrated in Fig. 3.5.

To avoid some unnecessary transitions from CELP to PWI inside an unvoiced segment of speech or from PWI to CELP inside a voiced segment of speech, a delayed-decision algorithm of the kind suggested in [22] can be applied to the preliminary

voiced/unvoiced frame classification. If a frame initially classified as voiced is in between two unvoiced frames, then it should be re-classified as unvoiced. If, on the other hand, a frame initially classified as unvoiced is in between two frames classified as voiced, then the threshold can be lowered by 10%, for example, and the voiced/unvoiced decision can be re-evaluated. If the classification is changed to voiced then a PWI/CELP decision must be made.

3.2.2 PWI to CELP Transition

The transition from a CELP to PWI coding mode does not impose a problem, however, the transition from PWI to CELP does pose a problem since the PWI method synthesizes a signal which is does not maintain sample-by-sample synchrony to the original signal while the linear predictive methods and the SNR error criterion of CELP requires sample-by-sample synchrony. A point of synchrony must therefore be re-established before the coder can proceed with CELP.

When the PWI decoder synthesizes N_{update} samples the synthesized signal is likely to lead or lag the original by a slight amount. The coder must send some additional information to the decoder to allow the decoder to re-establish a point of synchrony. One possibility is for the coder to transmit to the decoder the number of samples $N_{update} + \Delta$ that it should synthesize in order to end the synthesis at the point that is in synchrony to the end of the original fram Another possibility is for the coder to send information to the decoder that allows the decoder to determine how much it is out of synchrony to the original signal and consequently allow the decoder to re-establish synchrony. This former method is discussed here.

When the decoder synthesizes the speech frame it knows the phase $\phi(t)$ at which the synthesis has ended. However, $\phi(t)$ can be referenced only to the phase of the initial prototype for the current voiced speech section. $\tilde{G}'_c(m), \tilde{G}'_s(m)$ are shifted versions of the originally extracted prototype $\tilde{G}_c(m)$, $\tilde{G}_s(m)$ so the decoder does not know where the synthesis has ended in relation to the original signal. This problem is overcome by transmitting to the decoder at the PWI-to-CELP transition the shift

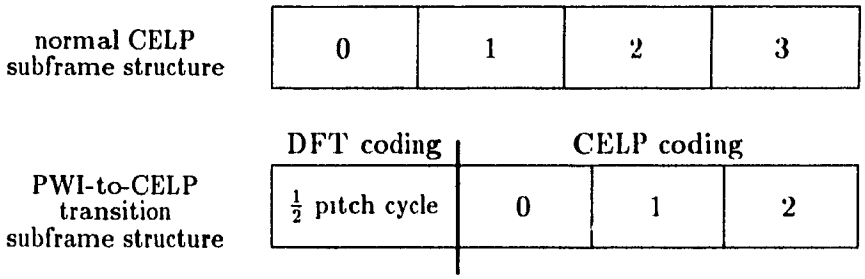


Figure 3.6: PWI-to-CELP transition subframe structure

of K_τ samples performed on the last prototype. The cost of transmitting K_τ is eight bits which corresponds to the allowable pitch period range at a 16 kHz sampling resolution.

The cost of transmitting the eight bits for K_τ is, however, not necessarily a penalty. If the centre of a prototype is taken to be the beginning of the update frame for prototype extraction, then the prototype contains a half pitch cycle of samples from the next update frame. The number of samples at an 8 kHz resolution is $L_g/4$ (with allowance for rounding error) where L_g is the pitch period at an upsampled 16 kHz resolution. Therefore, there are $L_g/4$ samples which the CELP coder does not have to encode in the PWI to CELP transition frame since they can be synthesized by an inverse DFT and this leaves $N_{update} - L_g/4$ samples which to be encoded by CELP. One approach is to divide the remaining $N_{update} - L_g/4$ samples into one less than the regular number of subframes to allow for a savings in bits (a 1016 CELP subframe employs 26 bits for encoding the residual and adaptive codebook parameters [3]). Another advantage of synthesizing the half pitch cycle and re-establishing synchrony at the "right hand" end of the prototype is that this produces $L_g/4$ samples to be used in the pitch filter memory that are synchronous to the original signal and not phase warped by the interpolation process of PWI. This subframe coding structure is illustrated in Fig. 3.6.

3.3 Quantization and Bit Allocation for PWI

The PWI method allows for an efficient encoding of voiced frames of speech using a small number of bits since only a prototype representative of one pitch period needs to be encoded instead of the entire update frame. Although quantization has not been implemented in the course of the research for this thesis, since this research has been focused on the fundamental principles of the interpolation process of PWI for speech representation, a scheme is presented here that is based on the scheme proposed in [8, 9] for quantizing the excitation prototypes.

3.3.1 Differential Encoding of the Prototypes

Since the current prototype is highly correlated to the previous one and has also been aligned to the previous, then it is efficient to employ differential quantization to represent the current prototype as a factor of the previous prototype plus additional codebook contributions to encode the difference. In [8, 9] it is suggested to use two codebooks to encode the difference. The DFT of the current aligned prototype is thus encoded as,

$$\hat{\mathbf{G}}' = \lambda_0 \hat{\mathbf{F}} + \lambda_1 \mathbf{c}_i^{(1)} + \lambda_2 \mathbf{c}_j^{(2)} \tag{3.35}$$

where $\mathbf{G}'$ and $\mathbf{F}$ are vectors whose components are the DFT coefficients $\tilde{G}'_c(m), \tilde{G}'_s(m)$ and $\tilde{F}_c(m), \tilde{F}_s(m)$ ($\hat{\mathbf{G}}'$ and $\hat{\mathbf{F}}$ are the quantized versions), $\mathbf{c}_i^{(1)}$ and $\mathbf{c}_j^{(2)}$ are the winning codewords i and j from the codebooks $\mathbf{c}^{(1)}$ and $\mathbf{c}^{(2)}$, and $\lambda_0, \lambda_1, \lambda_2$ are gain factors. Note that the previous prototype $\mathbf{F}$ has been aligned to the prototype before it but the $'$ has been dropped in the notation. At the beginning of a voiced speech segment, a "previous" prototype can be extracted from the last pitch period number of samples synthesized in the previous CELP frame to use for differentially encoding the initial prototype. The DFT vectors should have their cosine and sine terms paired into real and imaginary components such that they have the structure,

$$\mathbf{G}' = [\tilde{G}'_c(0) + j\tilde{G}'_s(0), \tilde{G}'_c(1) + j\tilde{G}'_s(1), \ldots, \tilde{G}'_c(L_g) + j\tilde{G}'_s(L_g)]^H \tag{3.36}$$

If $L_f \neq L_g$ then $\mathbf{F}$ must be truncated or padded with zeros to have the same number of terms as $\mathbf{G}'$ in (3.35). Also, the codebooks must be designed to allow for variable length codewords.

A good choice of codewords for the first codebook is one in which the codewords represent a single pulse in the sampled-time domain to allow for an accurate modelling of the impulsive nature of the excitation waveform. The gain λ_0 is first optimized assuming a zero contribution from the two codebooks, followed by an optimization for $\mathbf{c}_i^{(1)}$ assuming a zero contribution from the second codebook. The second codebook consists of codewords which are stochastically generated. In [8, 9] it is suggested that a good choice of codewords are ones which represent sparse pulses in the sampled-time domain. If desired, the second codebook can be orthogonalized to $(\lambda_0\hat{\mathbf{F}} + \lambda_1\mathbf{c}_i^{(1)})$ allow the sequential search to be optimal for the chosen codebook.

At each optimization stage, an error criterion must be employed. A spectrally weighted mean square error distortion measure is the common choice in coding speech frames in traditional coders and for PWI it can be used for encoding the prototypes. In the case here it is necessary to impose the spectral weighting on prototypes of the excitation signal in the DFT domain. The spectral weighting used in (3.22) for aligning the prototypes can again be applied here. The spectral weighting operation can be represented in matrix form using the following $L_g \times L_g$ diagonal matrix:

$$W_{mm} = \frac{\sum_{n=0}^{N_f}\gamma^n\alpha_n\cos(\frac{2\pi mn}{L_g}) + j\sum_{n=0}^{N_f}\gamma^n\alpha_n\sin(\frac{2\pi mn}{L_g})}{\left[\sum_{n=0}^{N_f}\gamma^n\alpha_n\cos(\frac{2\pi mn}{L_g})\right]^2 + \left[\sum_{n=0}^{N_f}\gamma^n\alpha_n\sin(\frac{2\pi mn}{L_g})\right]^2}, \quad m = 0, \ldots, L_g - 1. \quad (3.37)$$

The spectral weighting in (3.22) can thus be performed by the operation $\mathbf{V} = \mathbf{WG}$. For two arbitrary DFT vectors $\mathbf{X}$ and $\mathbf{Y}$ their spectrally weighted distance can be evaluated by the following distortion measure,

$$D(\mathbf{X}, \mathbf{Y}) = \frac{(\mathbf{X} - \mathbf{Y})^H\mathbf{W}^H\mathbf{W}(\mathbf{X} - \mathbf{Y})}{\mathbf{X}^H\mathbf{W}^H\mathbf{W}\mathbf{X}} \quad (3.38)$$

It can be easily shown from Parseval's relation that (3.38) is equivalent to sampled-

time domain distortion measure

$$d(\mathbf{x},\mathbf{y}) = \frac{(\mathbf{x}-\mathbf{y})^T\mathbf{H}^T\mathbf{H}(\mathbf{x}-\mathbf{y})}{\mathbf{x}^T\mathbf{H}^T\mathbf{H}\mathbf{x}} \tag{3.39}$$

in which the matrix **H** represents the spectral weighting in the sampled-time domain.

The following bit allocation is suggested in [8]. Eight bits are required to encode the pitch period at the upsampled 16 kHz resolution. Similarly, the first codebook requires eight bits to encode the possible pitch pulses and the second codebook is also assigned eight bits. Each of the three gain factors is allocated five bits. The LSF's for the short-term LP filter are quantized with 24 bits using the split vector quantization method of [23]. Lastly, one bit is required for the PWI/CELP decision. The total number of bits per update frame is thus 64. This bit allocation scheme is summarized in Table 3.2. At a 20 ms update rate the rate of the PWI coder is 3.2 kb/s.

A slight variation on this encoding scheme is presented in [9]. Only seven bits instead of eight are used for the pitch period so that a DFT with a period of an odd number of samples (at 16kHz resolution) must to padded with a zero to have an even period. In other words, the prototype waveform shape is extracted at high resolution but its period is then rounded to the original resolution. Three bits are used in this scheme for the PWI/CELP decision instead of just one, likely because the subframe classification presented in [5] is being used to allow for a transition within an update frame. The total number of bits used in this encoding scheme is 65 bits. It was applied in [9] using a 25 ms update rate to achieve a 2.6 kb/s coding rate.

In comparison, 1016 CELP which operates at an update frame rate of 30 ms has a 4.8 kb/s rate. However, this is not an entirely fair comparison since there are six extra bits used by 1016 CELP which are not accounted for by the PWI coding scheme. These six bits are used for synchronization, error correction, and future expansion. Furthermore, ten bits can be removed in 1016 CELP if the 24 bit LSF quantization scheme of [23] is used, instead of its current 34 bit scalar quantization scheme. Subtracting these sixteen bits, a fair bit rate to use for 1016 CELP is 4.27 kb/s when comparing to the PWI encoding schemes above.

Parameter	Bits Allocated
LSF's	24
pitch period	8
pitch gain	5
codebooks	2×8
codeword gains	2×5
frame classification	1

Table 3.2: Bit Allocation for PWI

3.3.2 Controlling the Level of Periodicity

A problem sometimes associated with differential quantization techniques at low bit rates is that a quantized component from a previous update time remains too dominant in the quantization of successive updates. Using a very low bit rate for PWI would cause the quantized prototypes in a voiced segment to be very similar to the first prototype extracted and quantized. This characteristic is, however, not necessarily a disadvantage for encoding voiced speech since it allows the reproduced speech to maintain its quasi-periodic un-noisy character. This result was observed in [24] in which the perceptual performance of a low-rate CELP coder was improved by reducing the magnitude of the residual codebook contribution in the reconstructed speech to emphasize the adaptive codebook contribution. Although this scheme achieves a lower SNR measure than standard CELP, the perceptual quality is improved by allowing the synthesized signal to have more periodicity. A similar improvement was also obtained in [25] in which the feedback of the residual codebook contribution to updating the adaptive codebook was reduced.

The PWI encoding method lends itself to easily control the level of periodicity in the synthesized update frame so that the desired level of periodicity can be controlled. Too little periodicity results in a noisy character and too much periodicity can result

in a buzzy character [7]. A convenient measure of the periodicity to use for PWI is the normalized cross-correlation which can be computed in the DFT domain between the two excitation prototypes. The spectrally weighted cross-correlation between two DFT vectors $\mathbf{X}$ and $\mathbf{Y}$ is:

$$R_X(\mathbf{X},\mathbf{Y}) = \frac{\mathrm{Re}\{\mathbf{X}^H\mathbf{W}^H\mathbf{W}\mathbf{Y}\}}{[\mathbf{X}^H\mathbf{W}^H\mathbf{W}\mathbf{X}\mathbf{Y}^H\mathbf{W}^H\mathbf{W}\mathbf{Y}]^{1/2}} \tag{3.40}$$

which is the correlation in (3.34) in matrix form. Equivalently, the periodicity can be measured in terms of the *signal-to-change ratio* (SCR) defined in [7, 9],

$$\mathrm{SCR}(\mathbf{X},\mathbf{Y}) = \left[1 - \frac{(\mathrm{Re}\{\mathbf{X}^H\mathbf{W}^H\mathbf{W}\mathbf{Y}\})^2}{\mathbf{X}^H\mathbf{W}^H\mathbf{W}\mathbf{X}\mathbf{Y}^H\mathbf{W}^H\mathbf{W}\mathbf{Y}}\right]^{-1} \tag{3.41}$$

The encoding procedure is to first measure $\mathrm{SCR}(\mathbf{G}',\mathbf{F})$, the SCR between the unquantized prototypes, then to quantize $\mathbf{G}'$ such that the synthesized speech maintains an SCR as close as possible to the original, i.e. $\mathrm{SCR}(\hat{\mathbf{G}}',\hat{\mathbf{F}}) \approx SCR(\mathbf{G}',\mathbf{F})$. The SCR between the quantized prototypes is controlled by adjusting the gains $\lambda_0, \lambda_1, \lambda_2$ and no extra bits are required to be transmitted. The SCR between two update prototypes is called the *long-term* SCR.

It was found in [7, 9] that a buzziness can sometimes occur in the reconstructed voiced speech. At very low bit rates the buzziness was identified to be caused by poor quantization of the prototypes. When the prototypes are quantized according to the bit assignment proposed in Section 3.3.1 it was found that the small amount of buzziness that was still present could be almost completely removed by injecting a small amount of high frequency noise (above 2 kHz). The explanation in [7, 9] is that for high pitched speakers, such as female speakers, there may be several pitch cycles within an update frame and PWI may synthesize a signal that is too periodic even when the long-term SCR is maintained since the variation from cycle-to-cycle is not modelled. It is suggested that the *short-term* SCR between adjacent pitch cycles can be measured and controlled by injecting a small amount of noise into the reconstructed speech and that a practical solution is to inject a standard amount of noise above 2 kHz for all speakers.

In the simulations performed for this thesis with unquantized prototypes there was no buzziness detected in the reconstructed speech. Occasionally, some warble or noise could be heard and, as will be explained in Chapter 4, this is caused by the poor speech synthesis that can occur when the PWI reconstructed excitation is passed through the time-varying formant synthesis filter. When PWI is applied directly on the unfiltered speech instead of the excitation, then there is no audible distortion when the speech domain pitch prototypes are unquantized. It is therefore concluded that the theory of needing to maintain the short-term SCR is not a complete explanation for the perceptual improvement that occurs in the reconstructed speech when it is injected with noise and a further investigation into this phenomenon is needed.

Chapter 4

Improving PWI

The objective of PWI is to synthesize a smoothly evolving speech waveform by performing an interpolation between pitch cycle prototypes and thereby maintain a high level of periodicity in the voiced speech. When a linear interpolating function is used then it is desired that the reconstructed speech also corresponds to a linear interpolation. However, it was found through experimentation that the PWI approach of Kleijn [7, 8, 9, 10] does not always achieve its objective. PWI can impose undesired envelope variations in the synthesized speech which occasionally results in audible warble. This phenomenon is examined in Section 4.1 and it is found to be a result of inconsistencies which occur when passing the PWI synthesized excitation signal through the formant synthesis filter because of the time-varying nature of the filter.

Methods for improving PWI are discussed in Sections 4.2 and 4.3. The first method is a post-processor which performs an energy fixup to smoothen the envelope imposed on the synthesized speech. The true solution, it is proposed in Section 4.3, is to apply the PWI method on the unfiltered speech and avoid the time-varying effects of the LP formant filter.

4.1 Time-varying Effects in PWI

The LP formant filter has proven to be very effective for removing redundancy in a speech signal to leave a smaller signal, the excitation signal, to be quantized. In an LPC/PWI framework the formant filter removes some spectral redundancy and leaves prototypes of the excitation signal to be encoded by PWI. While the formant filter provides for efficient encoding it can produce undesired effects in the synthesized speech due to its time-varying nature.

Some examples of these effects which may occur in the synthesized speech are shown in Fig. 4.1 and 4.2, in which the DFT prototypes of the excitation were not quantized but quantized LSF's for the formant filter were used. A synthesized speech frame is shown Fig. 4.1b in which it can be seen that the amplitude of the synthesized signal decreases in the middle of the frame compared to the original speech shown in Fig. 4.1a. Fig. 4.2a shows a segment of original speech and Fig. 4.2b shows the synthesized speech segment in which the non-linear effects which occur in the successive update frames produces an envelope in the speech which results in a slightly audible warble. The perceptibility of the warble may be small when compared to other perceptual noises which may occur when full encoding is applied in the simulation (then again, the quantization is also likely to increase the amount of warble), but because this undesired phenomenon occurs when the DFTs are not even quantized then this a problem in the model which warrants investigation.

Applying PWI on the excitation signal produces, by definition of PWI, a synthesized excitation signal which is a linear interpolation between two prototype waveforms. However, the LPC parameters of the formant filter are changing every subframe such that the current and previous prototype are not filtered with the same LPC parameters. Passing an excitation signal through a formant synthesis filter whose LPC coefficients are different from the ones used to filter the original speech produces a sub-optimally reconstructed speech signal. Therefore, for an update frame synthesized by the PWI method, the middle of the frame is likely to suffer the poorest

speech reconstruction since the combination of the two prototypes and LPC coefficients is greatest here. The amplitude of the speech sometimes dips in the middle of the frame and if this phenomenon occurs for a few successive frames then a warble in the speech is likely to be audible. Increases in amplitude are also possible. In Section 4.2 a post-processor is examined to reduce the non-linear progression of the speech amplitude which can occur in a frame.

While the instances of poor speech reconstruction occur more likely in the middle of a frame, they may also occur near the beginning or end of the frame due to subframe LSF interpolation. A prototype waveform is an extraction of one pitch cycle of the excitation signal and it may have some of its samples which resulted from formant filtering the speech using one set of LPC coefficients while other samples in the same prototype resulted from filtering the speech with a different set of coefficients. For example, if the prototype is extracted such that the centre of the prototype is located at the beginning of an update frame (this is the case for the simulations performed for this thesis) then the left-hand side and the right-hand side of the prototype are produced by a formant filtering operation with different LPC coefficients since they fall in different subframes. If half the pitch period is longer than a subframe length then parts of the prototype extends into further subframes and this would result in even greater inconsistency in the resynthesis filtering.

The problem of having segments of a prototype excitation derived from different LPC coefficients can be partially avoided by structuring the prototype extraction such that right-hand end of the prototype corresponds to the end of an update frame, i.e. the prototype is completely to left of the frame boundary, but this approach does not avoid the problem when a pitch period is longer than one subframe. Furthermore, the previous prototype would be extracted completely from the previous update frame and may not serve as a good excitation signal for synthesizing speech using the LPC coefficients of the subframes in the current update frame. In others words, this structure could improve the speech reconstruction near the end of the frame but also worsen the reconstruction near the beginning of the frame in comparison to extracting

the prototypes with the centres of the prototypes located at the frame boundaries.

When pitch doubling occurs then the extracted prototype of two true pitch cycles carries over into even more subframes. The likelihood of poor speech reconstruction is therefore greater. Furthermore, the fact that pitch doubling occurs implies that two successive pitch cycles of the excitation may not be highly correlated to each other and therefore poor prediction gain may result when sample-by-sample synchrony is not maintained. In Fig. 4.3b four successive speech frames reconstructed by PWI are shown in which pitch doubling occurs at one of the updates and pitch tripling occurs at another. The original speech is shown in Fig. 4.3a. The reconstructed speech shows some inconsistency in the amplitudes, especially near the frame boundaries.

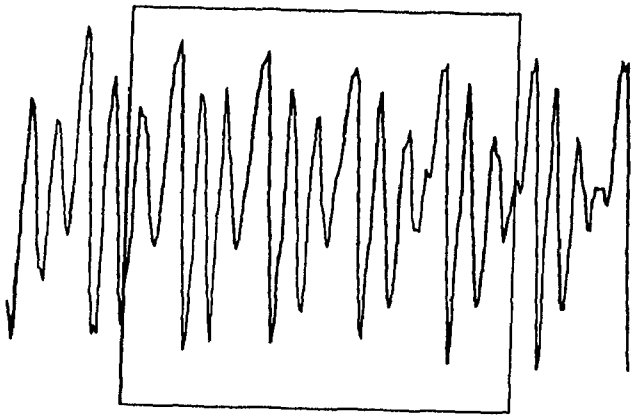
Pitch doubling is, in general, a problem for any coder since it inherently implies that there is a lack of correlation between two successive pitch cycles. It is most certainly a problem for PWI but fortunately it does not occur very often when a high sampling resolution and a good pitch estimation algorithm is used.

In summary, the time-varying coefficients of the LP formant filter introduce undesired effects in the speech synthesized by passing the PWI synthesized excitation signal through the formant synthesis filter. The LP filter of tenth order is sensitive to changes in its parameter values as well as its previous filter memory values and excitation. The procedure of filtering and re-synthesizing speech using such a filter in an open-loop manner may produce poor results when the sample-by-sample consistency is not maintained between the excitation samples and the time-varying filter coefficients. The PWI method proposed by Kleijn does not truly provide the control desired over the level of periodicity of the reconstructed speech and thus can fail to produce a smoothly evolving waveform.

4.2 Energy Fixup for PWI

The non-linear progression of the speech amplitude in a PWI reconstructed frame is often in the form of a dip or increase in the middle of a PWI synthesized frame

a) original



b) reconstructed

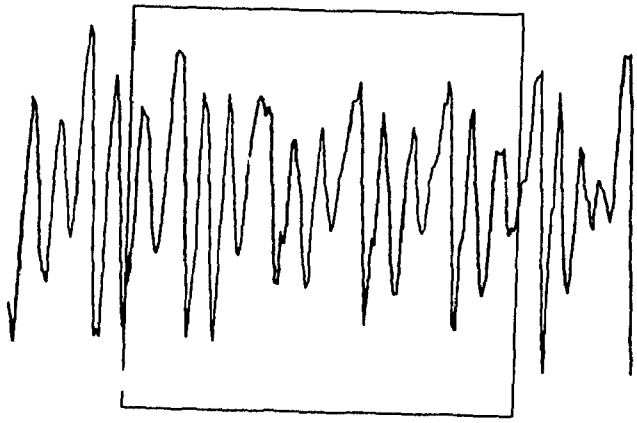
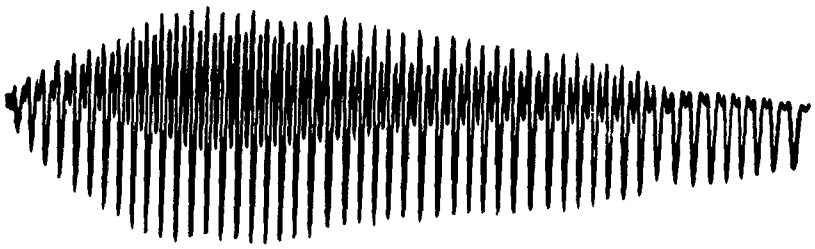


Figure 4.1: PWI reconstructed speech frame exhibiting a drop in amplitude

a) original



b) reconstructed

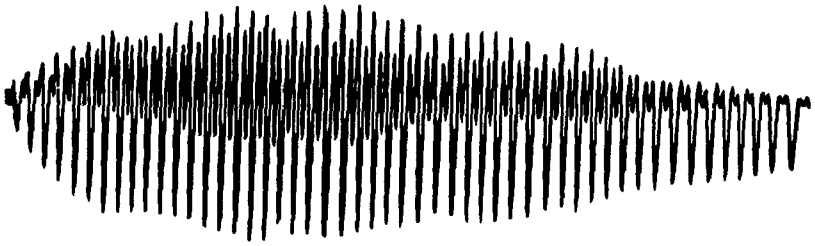
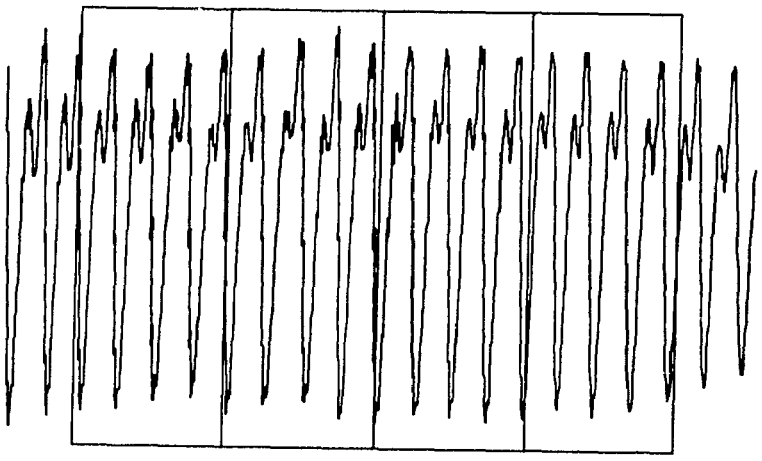


Figure 4.2: PWI reconstructed speech exhibiting undesired envelope variations

a) original



b) reconstructed

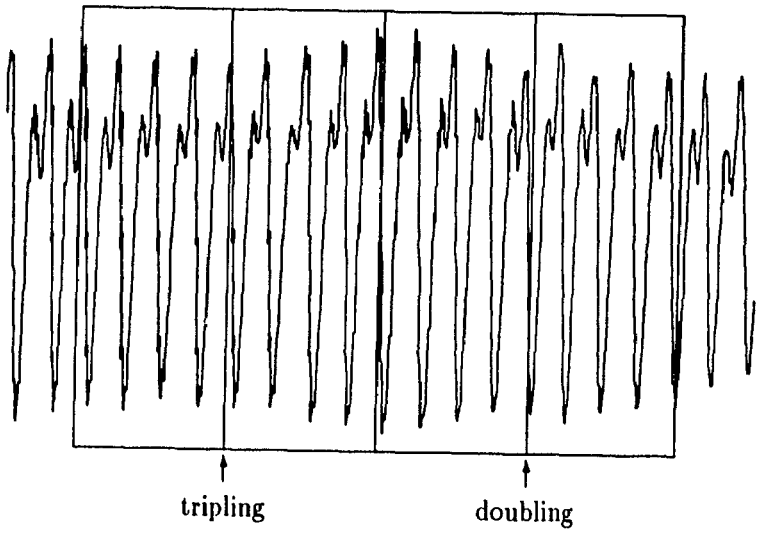


Figure 4.3: PWI reconstructed speech under pitch doubling

and this can cause an audible warble. In this section a post-processing algorithm is investigated which smoothens the undesired envelope. The post-processor does not require any extra information bits to be sent by the source.

The post-processor proposed is an energy fixup which measures the energy of a pitch cycle at the beginning of a synthesized speech frame, at the end of the frame, and in the middle of the frame, and then attempts to smoothen the evolution of the energy in the frame. The synthesized speech frame is modified by multiplying the samples according to an "energy" curve. The curve consists of two line segments defined such that the endpoints of the curve are constrained to a value of unity and the centre point of the curve has the value,

$$\Gamma = \left[\frac{E_B + E_E}{2E_M}\right]^{1/2}. \tag{4.1}$$

in which E_B, E_E, E_M are the energies in the beginning, end, and middle of the frame, respectively, and using E_B for example they can be computed by,

$$E_B = \frac{1}{L_f/2} \sum_{k=0}^{L_f/2-1} |\hat{s}(k)|^q \tag{4.2}$$

in which $\hat{s}(k)$ are the output synthesized speech samples and $L_f/2$ should be rounded to an integer to correspond to the pitch period at the normal 8 kHz resolution. Using an exponent $q = 2$ corresponds the energy of a pitch cycle, but it may be preferable to use a higher exponent, such as $q = 3$ to emphasize the peaks in the pitch cycle. The energy curve for a frame length of N samples is defined as,

$$c(k) = \begin{cases} 1 + \dfrac{\Gamma - 1}{N/2} \cdot k & , \quad k = 0, \ldots, N/2 - 1 \\ 1 + \dfrac{\Gamma - 1}{N/2} \cdot (N - k) & , \quad k = N/2, \ldots, N - 1 \end{cases} \tag{4.3}$$

A typical fixup curve is illustrated in Fig. 4.4.

A synthesized speech frame which suffers from a dipping in amplitude is shown in Fig. 4.5b with the original speech shown in Fig. 4.5a. In Fig. 4.5c the reconstructed speech frame has been post-processed by an energy fixup with $q = 3$. A longer segment of the same speech is shown in Fig. 4.6, illustrating the smoothening effect of the post-processor.

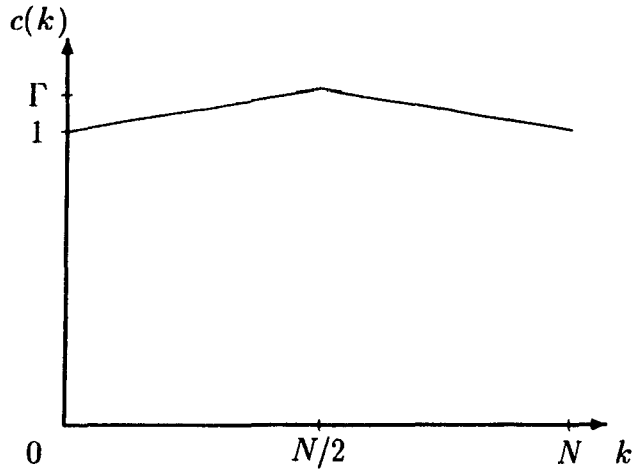


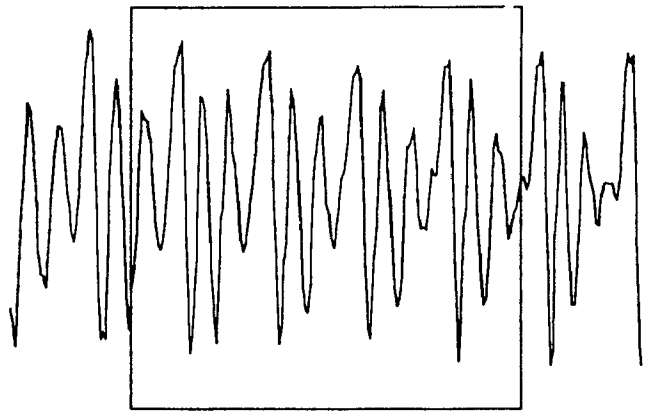
Figure 4.4: Energy Fixup Curve

The reconstructed speech segment from Fig. 4.2b which suffers from an undesired envelope is again shown in Fig. 4.7a and in Fig. 4.7b it has been post-processed by an energy fixup with $q = 3$. For this example, the amplitude of the undesired envelope is only slightly reduced and the shape of the envelope remains imposed on the speech. In general, the post-processor cannot remove warble in the synthesized speech but only reduce the level of the warble by a small amount. Even if a more sophisticated fixup algorithm is designed, it would be able to achieve only limited success. It can only be helpful when the speech reconstruction at the beginning and end of the frame is reliable since it attempts to smoothen the evolution of the waveform from the beginning of the frame to the end. When, occasionally, the beginning or the end of a frame is not well reproduced then unfortunately the post-processor cannot help improve the speech.

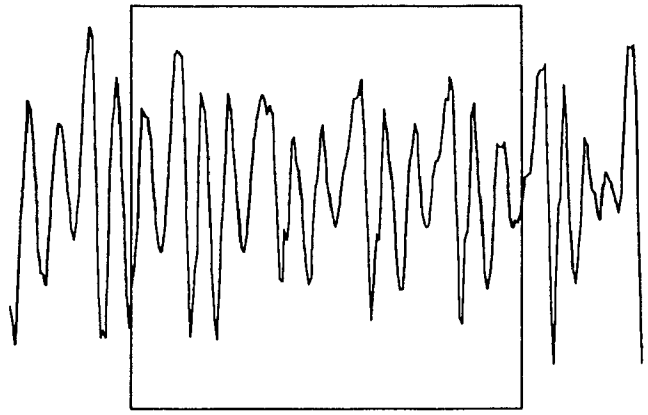
4.3 Applying PWI on the Unfiltered Speech

Careful listening tests and detailed visual examination of the waveforms reconstructed by Kleijn's PWI method revealed undesired amplitude variations. Distortion in the

a) original



b) reconstructed without fixup



c) reconstructed after fixup

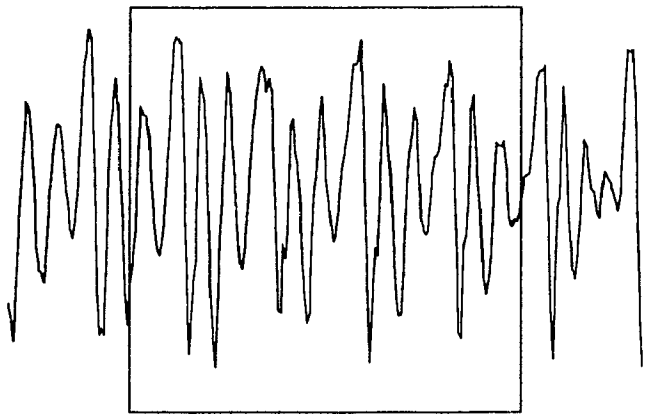
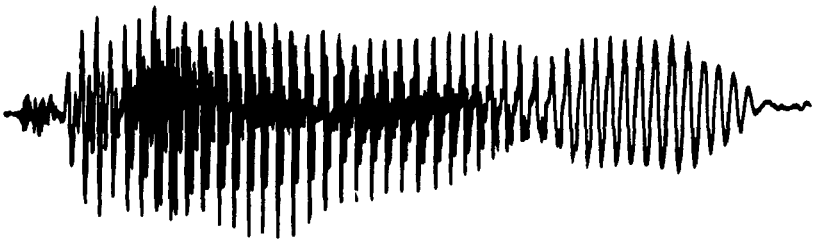
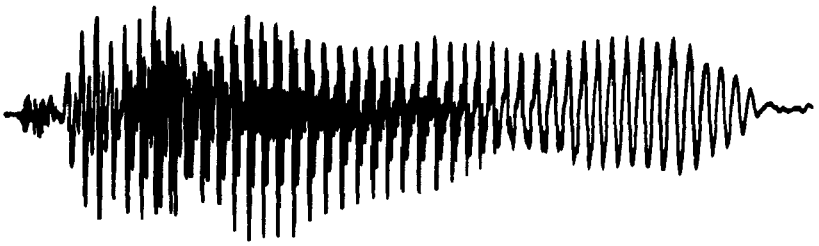


Figure 4.5: Energy fixup on a PWI frame

a) original



b) no fixup



c) after fixup

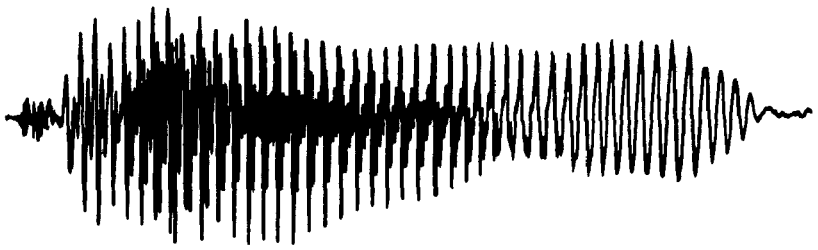
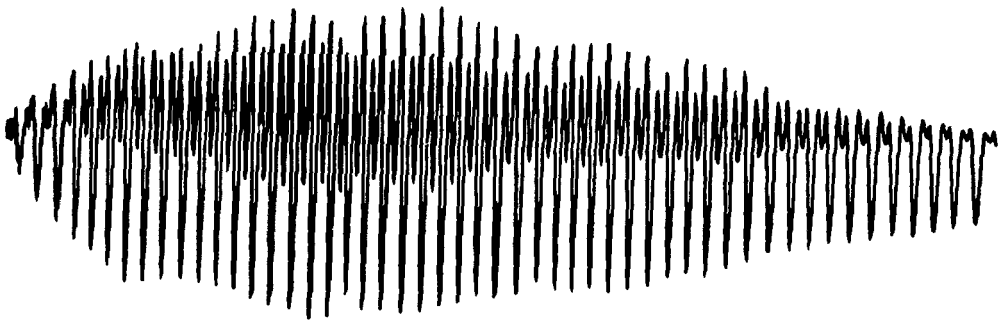


Figure 4.6: Energy fixup on a PWI segment

a) no fixup



b) after fixup

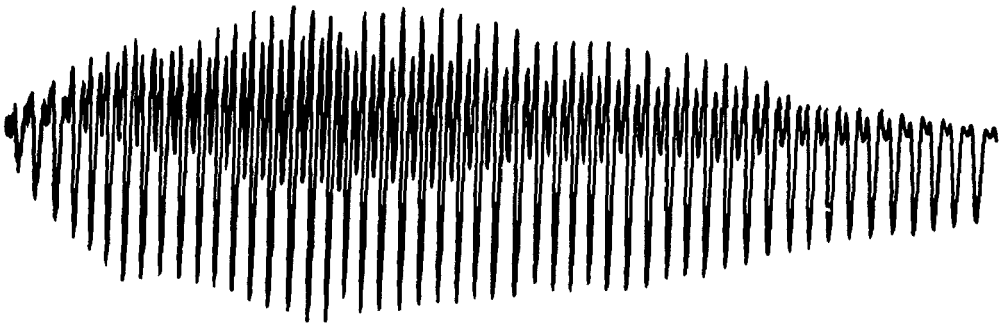


Figure 4.7: Energy fixup on a PWI segment with little effect

voiced-to-unvoiced transition and unvoiced frames was removed by making these samples identical to the original samples, allowing for an investigation of solely the PWI modelling effects. It was speculated that the time-varying nature of the formant filter was the source of the problem. This speculation was confirmed by performing the prototype extraction and interpolation in the original speech domain. The time-shift τ for prototype alignment and the cross-correlation between successive prototypes for making the PWI/CELP decision were determined in the excitation domain to keep them consistent with the previous implementation of performing PWI on the excitation. Performing PWI in the speech domain did indeed produce the smoothly evolving waveform that is desired, confirming the speculation that the undesired envelope variations were caused by the time-varying nature of the formant filter.

The simplest way to avoid the problems associated with the time-varying nature of the LP formant filter is not to use the filter and apply PWI on the clean speech signal. PWI, by definition, produces a linearly interpolated waveform when a linear interpolating function $\beta(k)$ is used. However, when a formant filter is first applied on the speech then the PWI method produces a linearly interpolated excitation but falls short of its objective to synthesize a smooth linearly evolving speech waveform. Applying PWI on the unfiltered speech ensures, by definition, that a linearly interpolated speech waveform is synthesized. Furthermore, pitch doubling is not as serious a problem since the successive pitch cycles of speech are likely to be highly correlated even under pitch doubling.

When applying PWI directly in the speech domain the same procedure discussed in Chapter 3 is applicable with two modifications. Firstly, prototypes are extracted in the speech domain so the DFT is used to represent prototypes of the speech and not the excitation signal. Secondly, the spectral weighting of the DFT coefficients in (3.22) must be modified for the prototypes of speech. If formant deemphasis is not desired then there is no weighting operation needed. To apply the weighting $W(z) = A(z)/A(z/\gamma)$ as in CELP then firstly the DFT coefficients of the prototypes in the excitation signal domain that would be obtained from filtering the speech with

the filter $A(z)$ are determined from the DFT coefficients $S_c(m)$ and $S_s(m)$ of the speech prototypes by:

$$G_c(m) = S_c(m)\sum_{l=0}^{N_f}\alpha_l \cos(2\pi ml/L_g) - S_s(m)\sum_{l=0}^{N_f}\alpha_l \sin(2\pi ml/L_g)$$
$$G_s(m) = S_c(m)\sum_{l=0}^{N_f}\alpha_l \sin(2\pi ml/L_g) + S_s(m)\sum_{l=0}^{N_f}\alpha_l \cos(2\pi ml/L_g). \tag{4.4}$$

as shown in Appendix B. Now (3.22) can be applied to perform the inverse filtering operation $1/A(z/\gamma)$ with $\gamma < 1$ for formant deemphasis.

Some examples of speech segments synthesized by applying PWI on the unfiltered speech are compared to the equivalent segments synthesized by applying PWI on the excitation in Fig. 4.8 and 4.9. The samples in the non-PWI frames were made identical to the original speech samples to allow for a direct comparison between the PWI frames of the two approaches. The speech segment from Fig. 4.2 is again shown in Fig. 4.8 with the undesired envelope successfully removed by applying PWI on the speech whereas the energy fixup of Section 4.2 had failed. The speech segment which suffers from pitch doubling/tripling in Fig. 4.3 is again shown in Fig. 4.9 in which the speech reconstructed applying PWI directly in the unfiltered speech domain maintains a smooth evolution of the amplitude. There was no audible distortion in the sample speech phrases reconstructed by applying PWI directly on the unfiltered speech when the prototypes were not quantized.

Although there was no quantization of the prototypes in the simulations performed for this thesis, a brief discussion of the subject is provided here. When applying PWI on the excitation signal, the formant filter provides for an efficient removal of redundancy in the speech but can also produce undesired results in the resynthesized speech. When applying PWI on the unfiltered speech, the prototypes of the unfiltered speech must be quantized. It is expected that an increased number of bits is not required in comparison to Kleijn's scheme. The efficient encoding offered by the formant filter can also be exploited in this case. The formant filter can be applied to filter the prototypes only, instead of filtering an entire update frame. The resid-

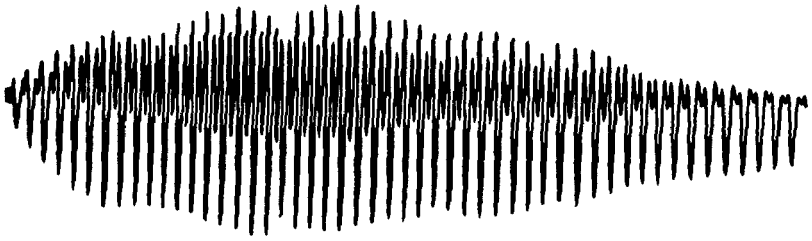
ual of the prototype is differentially encoded as in (3.35) and then the prototype is resynthesized by passing the coded residual through the inverse filter. The previous prototype should be used as past samples for filtering the current prototype since the decoder has not yet synthesized the samples in between the prototypes. Using the reconstructed speech prototypes, waveform interpolation is performed in the speech domain to guarantee a smoothly evolving speech waveform. This encoding method may even be more efficient than the case when applying PWI on the excitation since the LPC analysis is performed on only a single pitch cycle instead of an entire update frame so higher prediction gain may be achieved.

Another quantization scheme to investigate is to encode the DFT's of the speech domain prototypes directly using the differential encoding scheme in (3.35). Efficient encoding may be achieved since the successive prototypes are highly correlated to each other. The bits that would have been used for LSF quantization can instead be used for increasing the resolution of the gains, increasing the size fo the codebooks, and even increasing the number of codebooks.

An additional problem which requires careful investigation when full encoding and full integration to CELP is implemented is the performance of the coder in the voiced-to-unvoiced transitions. In particular, when PWI is applied on the unfiltered speech there is no excitation signal generated in PWI frames. At a voiced-to-unvoiced transition the coder must therefore produce a sufficient number of past excitation samples to be used by the LP pitch filter (or adaptive codebook) of CELP. The method chosen will produce excitation samples that slightly differ from those produced by applying PWI on the excitation. The voiced-to-unvoiced transition frame will not be identical to that produced by Kleijn's scheme, but is not necessarily inferior.

a) original

b) PWI on excitation



c) PWI on speech

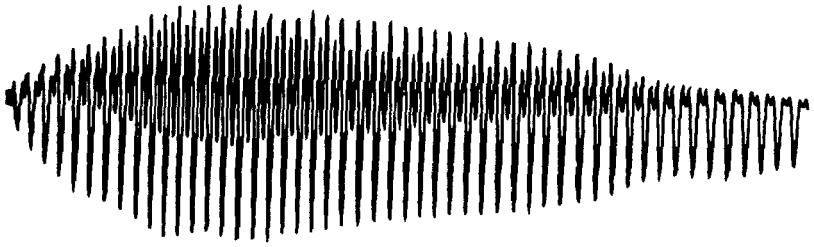
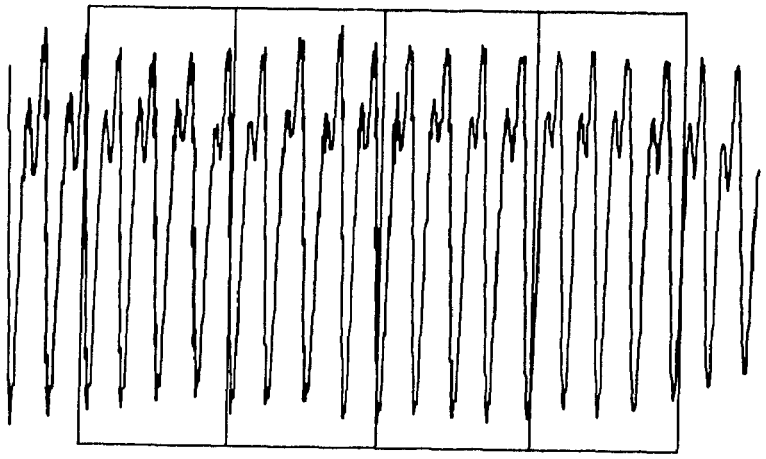


Figure 4.8: PWI applied directly on speech

a) original



b) PWI on speech

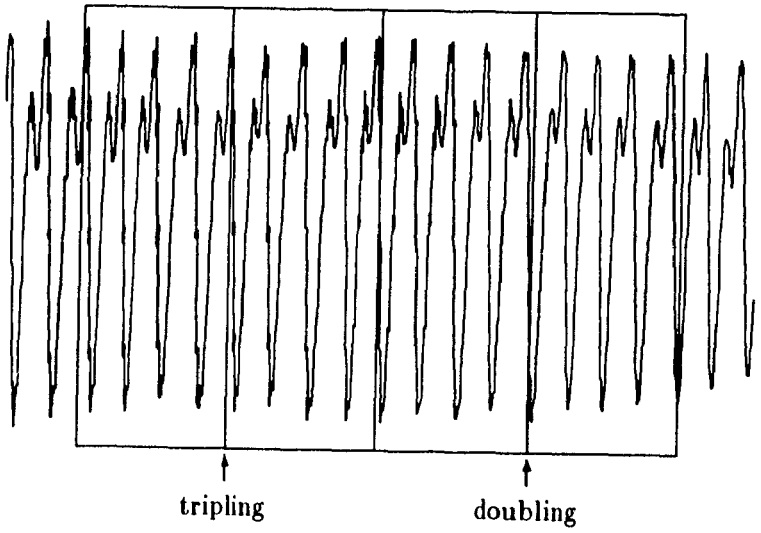


Figure 4.9: PWI applied directly on speech under pitch doubling

Chapter 5

Conclusion

The purpose of this thesis was to examine the implementation and application of a recently proposed speech coding method called prototype waveform interpolation (PWI) for encoding voiced segments of speech at low bit rates. The objective of PWI is not only to achieve a low bit rate but also to achieve higher quality, more natural sounding speech than current low rate coders.

Voiced speech is characterized by a high level of periodicity and a clean, noise-free waveform shape. Traditional coders are driven by weighted signal-to-noise ratio (SNR) criteria which do not control the level of periodicity in the reconstructed speech and therefore this fundamental characteristic of the voiced speech may not be well preserved. While these coders may shape the error by using error weighting filters and adaptive post-filtering, they do not provide a careful control over the level of periodicity. As a result, these coders do not achieve natural sounding voiced speech despite the high SNR scores which they can achieve in the voiced frames.

The PWI method encodes only a single pitch cycle prototype of voiced speech per update frame (the prototypes, however, must be represented at an upsampled 16 kHz resolution for good performance) to achieve a low coding rate and it maintains a high level of periodicity in the reconstructed speech to achieve high quality voiced speech reconstruction. SNR based error evaluation techniques are not used and sample-by-sample phase synchrony is not maintained to allow for the speech to be reconstructed

by a smooth interpolation between adjacent prototype waveforms. The interpolation maintains a high level of periodicity in the signal which is controlled by maintaining the cross-correlation between the quantized prototypes to be the same as the cross-correlation between the unquantized prototypes.

One of the contributions of this thesis is simply to provide some of the details needed to implement PWI that are not provided in the available literature. A practical implementation of PWI using the discrete Fourier transform (DFT) was presented which is an approximation to the conceptual PWI formulation in the continuous-time domain using the Fourier series (FS). The need re-establish sample-by-sample synchrony when the coder switches from PWI to code-excited linear prediction (CELP) was explained and one method for performing this re-synchronization was presented. Some of the details in the implementation of PWI used in this thesis are likely to differ from the implementation performed by Kleijn, but the principles do not differ.

The major contribution of this thesis was the detection of a problem in the PWI method and the identification of its source. Performing the interpolation procedure between unquantized prototypes resulted in reconstructed speech which could have an undesired envelope. While this effect was usually imperceptible, it was occasionally perceptible as a warble, especially in long voiced segments. The PWI method was not quite achieving its objective of producing a smoothly evolving waveform corresponding to a linear interpolation and therefore it was felt that there was a fundamental problem in the PWI model which should be investigated instead of jumping into bit encoding for comparing PWI to other coders.

The problem in PWI was identified to be caused by the time-varying nature of the short-term linear predictive (LP) filter, called the formant filter. PWI was applied on the excitation signal after the speech was formant filtered. The PWI process, by definition, produced a linearly interpolated excitation signal but this did not correspond to a linear interpolation in the speech domain after the reconstructed excitation was passed through the time-varying formant resynthesis filter. The scheme proposed by Kleijn provides control over the level of periodicity in the reconstructed excitation, but

this does not correspond to control over the level of periodicity in the reconstructed speech.

The PWI method proposed by Kleijn was applied inside a linear predictive coding (LPC) framework since the formant filter has been a popular choice for efficiently removing redundancy in a speech signal. Traditional coders perform a closed-loop analysis-by-synthesis optimization to overcome as much as possible the sensitivity of the formant filter to errors in the excitation signal and in the previous values needed for the filter memory. The analysis-by-synthesis approach is not, however, suitable for PWI since the basic approach of PWI is to construct an interpolated signal and not to perform an SNR based optimization search Furthermore, other low-rate speech coders can also suffer from an inconsistent reproduction of the speech amplitude envelope despite the "robustness" of the analysis-by-synthesis approach because of the coarse quantization used.

A post-processing algorithm was implemented to smooth out the undesired envelope in the reconstructed speech using an energy fixup. Unfortunately, the post-processor achieves only marginal improvements since it only reduces the amplitude of the envelope variations but rarely could remove them and therefore the characteristic of the signal that causes warble was not removed. While more sophisticated post-processors were contemplated, in addition to a prototype extraction procedure which tries to avoid as much as possible the time-varying nature of the formant filter, these approaches do not get to the root of the problem.

The root of the problem is removed by applying the PWI method on the input speech without formant filtering. The smooth interpolation desired in the reconstructed speech was achieved. With the prototypes left unquantized, there was no audible distortion in the reconstructed speech. It is preferable for the reconstructed speech to have an amplitude envelope which is smoother than the original, rather than one which is more varying and erratic.

The PWI coding method proposed by Kleijn may achieve good results compared to other speech coders (according to Kleijn's own simulation results) and has contributed

substantially to research in speech coding, however, further research on PWI should first take a step back and start off on the right foot. Quantization and bit allocation for PWI should be approached from the viewpoint of efficiently encoding prototypes of the unfiltered speech, not the excitation. The performance of the encoding method in the voiced-to-unvoiced transitions must also be considered. The method of applying first a formant filter and then performing PWI on the excitation signal is one option to be compared against others.

The motivation behind PWI is to produce natural sounding voiced speech by preserving fundamental characteristics of the speech – its level of periodicity and continuously evolving pitch period length. These, however, are only two characteristics of the speech. It was argued in this thesis that Kleijn's theory of having excessive correlation between successive pitch cycles is not a complete explanation for the improvement obtained in Kleijn's PWI simulations achieved by injecting noise in the reconstructed speech. It appears that in order to achieve truly high quality speech coding at very low bit rates, a further understanding and modelling of the properties of natural speech may be needed.

Appendix A

Cross-correlation Expressed in the Fourier Series Domain

The following is a proof that the normalized cross-correlation between the two waveforms $x(t)$ and $y(t)$ defined on the interval $0 \leq t \leq T$ can be expressed in terms of their FS coefficients $\mathbf{X}$ and $\mathbf{Y}$ as:

$$\frac{\int_{t=0}^{T} x(t)y(t)dt}{\left[\int_{t=0}^{T} x^2(t)dt \cdot \int_{t=0}^{T} y^2(t)dt\right]^{1/2}} = \frac{\mathrm{Re}\{\mathbf{X}^H\mathbf{Y}\}}{[\mathbf{X}^H\mathbf{X}\mathbf{Y}^H\mathbf{Y}]^{1/2}} \tag{A.1}$$

and that the time-shift of τ that phase aligns $y(t)$ to $x(t)$ to achieve the maximum cross-correlation is determined by:

$$\begin{aligned}\tau = \underset{0\leq\tau'\leq T}{\mathrm{argmax}} \sum_{m=0}^{M} & [X_c(m)Y_c(m) + X_s(m)Y_s(m)]\cos(2\pi m\tau') \\ & +[X_s(m)Y_c(m) - X_c(m)Y_s(m)]\sin(2\pi m\tau').\end{aligned} \tag{A.2}$$

Proof of (A.1):

Let the FS representation for a bandlimited signal $x(t), 0 \leq t \leq T$ be:

$$x(t) = \sum_{m=0}^{M} X_c(m)\cos(2\pi mt/T) + X_s(m)\sin(2\pi mt/T) \tag{A.3}$$

and the vector of FS coefficients be defined as:

$$\mathbf{X} = [X_c(0) + jX_s(0), \ldots, X_c(M) + jX_s(M)]^T. \tag{A.4}$$

The (un-normalized) cross-correlation between $x(t)$ and $y(t)$ can thus be expressed as:

$$\int_{t=0}^{T} x(t)y(t)dt = \int_{t=0}^{T} \sum_{m=0}^{M}[X_c(m)\cos(2\pi mt/T) + X_s(m)\sin(2\pi mt/T)] \cdot \sum_{n=0}^{M}[Y_c(n)\cos(2\pi nt/T) + Y_s(n)\sin(2\pi nt/T)]dt. \tag{A.5}$$

Noting the orthogonality over the interval $0 \leq t \leq T$ for $m \neq n$ between the functions $\cos(2\pi mt/T)$ and $\cos(2\pi nt/T)$, $\sin(2\pi mt/T)$ and $\sin(2\pi nt/T)$, and for any m, n between $\cos(2\pi mt/T)$ and $\sin(2\pi nt/T)$ then the surviving terms are:

$$\int_{t=0}^{T} x(t)y(t)dt = \sum_{m=0}^{M} \int_{t=0}^{T} X_c(m)Y_c(m)\cos^2(2\pi mt/T) + X_s(m)Y_s(m)\sin^2(2\pi mt/T)dt. \tag{A.6}$$

Performing the integration yields:

$$\begin{aligned} \int_{t=0}^{T} x(t)y(t)dt &= \frac{T}{2}\sum_{m=0}^{M} X_c(m)Y_c(m) + X_s(m)Y_s(m) \\ &= \frac{T}{2}\,\mathrm{Re}(\mathbf{X}^H\mathbf{Y}). \end{aligned} \tag{A.7}$$

Similarly, it can be easily shown that

$$\int_{t=0}^{T} x(t)x(t)dt = \frac{T}{2}(\mathbf{X}^H\mathbf{X}). \tag{A.8}$$

The normalized cross-correlation can therefore be expressed as:

$$\frac{\int_{t=0}^{T} x(t)y(t)dt}{\left[\int_{t=0}^{T} x^2(t)dt \cdot \int_{t=0}^{T} y^2(t)dt\right]^{1/2}} = \frac{\mathrm{Re}\{\mathbf{X}^H\mathbf{Y}\}}{[\mathbf{X}^H\mathbf{X}\mathbf{Y}^H\mathbf{Y}]^{1/2}}. \tag{A.9}$$

Proof of (A.2):

The waveforms $x(t)$ and $y(t)$ can be phase-aligned by time-shifting $y(t)$ by τ such that the normalized cross-correlation between $x(t)$ and $y(t+\tau)$ is maximized. The

FS of $y(t+\tau)$ can be expressed in terms of the FS coefficients of $y(t)$ by:

$$\begin{aligned} y(t+\tau) &= \sum_{m=0}^{M} Y_c(m)\cos(2\pi m(t+\tau)/T) + Y_s(m)\sin(2\pi m(t+\tau)/T) \\ &= \sum_{m=0}^{M} [Y_c(m)\cos(2\pi m\tau/T) - Y_s(m)\sin(2\pi m\tau/T)]\cos(2\pi mt/T) \\ &\quad +[Y_c(m)\sin(2\pi m\tau/T) + Y_s(m)\cos(2\pi m\tau/T)]\sin(2\pi mt/T) \end{aligned} \tag{A.10}$$

Therefore, the FS coefficients of $\tilde{y}(t) = y(t+\tau)$ are:

$$\begin{aligned} \tilde{Y}_c(m) &= Y_c(m)\cos(2\pi m\tau/T) - Y_s(m)\sin(2\pi m\tau/T) \\ \tilde{Y}_s(m) &= Y_c(m)\sin(2\pi m\tau/T) + Y_s(m)\cos(2\pi m\tau/T). \end{aligned} \tag{A.11}$$

The time-shift τ which maximizes the cross-correlation between $x(t)$ and $\tilde{y}(t)$ is determined by:

$$\begin{aligned} \tau &= \underset{0\le\tau'\le T}{\text{argmax}} \frac{\text{Re}\{\mathbf{X}^H\tilde{\mathbf{Y}}\}}{\left[\mathbf{X}^H\mathbf{X}\tilde{\mathbf{Y}}^H\tilde{\mathbf{Y}}\right]^{1/2}} \\ &= \underset{0\le\tau'\le T}{\text{argmax}}\, \text{Re}\{\mathbf{X}^H\tilde{\mathbf{Y}}\} \\ &= \underset{0\le\tau'\le T}{\text{argmax}} \sum_{m=0}^{M} [X_c(m)Y_c(m) + X_s(m)Y_s(m)]\cos(2\pi m\tau') \\ &\quad +[X_s(m)Y_c(m) - X_c(m)Y_s(m)]\sin(2\pi m\tau'). \end{aligned} \tag{A.12}$$

Appendix B

LP Filtering Using the Fourier Series Coefficients

The following proves a result used by Kleijn that the FS coefficients of an excitation waveform $g(t)$, $0 \leq t \leq T$ derived from LP filtering a speech waveform $s(t)$, $0 \leq t \leq T$ with the filter coefficients $(\alpha_0, \alpha_1, \alpha_2, \ldots, \alpha_{N_f}) = (1, -a_1, -a_2, \ldots, -a_{N_f})$ can be represented in terms of the FS coefficients of the speech waveform by:

$$G_c(m) = S_c(m)\sum_{l=0}^{N_f}\alpha_l \cos(2\pi mlD/T) - S_s(m)\sum_{l=0}^{N_f}\alpha_l \sin(2\pi mlD/T)$$
$$G_s(m) = S_c(m)\sum_{l=0}^{N_f}\alpha_l \sin(2\pi mlD/T) + S_s(m)\sum_{l=0}^{N_f}\alpha_l \cos(2\pi mlD/T). \quad \text{(B.1)}$$

where D is the sampling interval of the LP filter.

Furthermore, the FS coefficients of the speech can be recovered from the FS coef-

ficients of the excitation by:

$$S_c(m) = \frac{G_c(m)\sum_{l=0}^{N_f}\gamma^l\alpha_l\cos(\frac{2\pi mlD}{T}) + G_s(m)\sum_{l=0}^{N_f}\gamma^l\alpha_l\sin(\frac{2\pi mlD}{T})}{\left[\sum_{l=0}^{N_f}\gamma^l\alpha_l\cos(\frac{2\pi mlT}{T})\right]^2 + \left[\sum_{l=0}^{N_f}\gamma^l\alpha_l\sin(\frac{2\pi mlT}{T})\right]^2}$$

$$S_s(m) = \frac{-G_c(m)\sum_{l=0}^{N_f}\gamma^l\alpha_l\sin(\frac{2\pi mlD}{T}) + G_s(m)\sum_{l=0}^{N_f}\gamma^l\alpha_l\cos(\frac{2\pi mlD}{T})}{\left[\sum_{l=0}^{N_f}\gamma^l\alpha_l\cos(\frac{2\pi mlD}{T})\right]^2 + \left[\sum_{l=0}^{N_f}\gamma^l\alpha_l\sin(\frac{2\pi mlD}{T})\right]^2} \tag{B.2}$$

in which the formants can be deemphasized, if desired, by setting $\gamma < 1$.

Proof of (B.1):

Consider a speech waveform $s(t)$ on the interval $0 \le t \le T$. The FS representation is:

$$s(t) = \sum_{m=0}^{M}[S_c(m)\cos(2\pi mt/T) + S_s(m)\sin(2\pi mt/T)] \tag{B.3}$$

in which the FS coefficients are determined by:

$$\begin{aligned} S_c(m) &= \frac{1}{T}\int_0^T s(t)\cos(2\pi mt/T)dt \\ S_s(m) &= \frac{1}{T}\int_0^T s(t)\sin(2\pi mt/T)dt. \end{aligned} \tag{B.4}$$

Filtering the speech with an LP filter with the coefficients $(\alpha_0, \alpha_1, \alpha_2, \ldots, \alpha_{N_f}) = (1, -a_1, -a_2, \ldots, -a_{N_f})$ produces the excitation signal:

$$g(t) = s(t) - \sum_{l=1}^{N_f} a_l s(t - lD) = \sum_{l=0}^{N_f}\alpha_l s(t - lD). \tag{B.5}$$

Substituting the FS representation in (B.3) for $s(t)$ gives:

$$g(t) = \sum_{l=0}^{N_f}\alpha_l\left[\sum_{m=0}^{M} S_c(m)\cos(2\pi m(t - lD)/T) + S_s(m)\sin(2\pi m(t - lD)/T)\right]. \tag{B.6}$$

Applying the trigonometric identities

$$\begin{aligned} \cos(\theta - \phi) &= \cos\theta\cos\phi + \sin\theta\sin\phi \\ \sin(\theta - \phi) &= \sin\theta\cos\theta - cos\theta\sin\phi \end{aligned} \tag{B.7}$$

and re-arranging the terms gives:

$$g(t) = \sum_{m=0}^{M} \left[S_c(m)\sum_{l=0}^{N_f}\alpha_l \cos(2\pi mlD/T) - S_s(m)\sum_{l=0}^{N_f}\alpha_l \sin(2\pi mlD/T) \right] \cos(2\pi mt/T)$$
$$+ \left[S_c(m)\sum_{l=0}^{N_f}\alpha_l \sin(2\pi mlD/T) + S_s(m)\sum_{l=0}^{N_f}\alpha_l \cos(2\pi mlD/T) \right] \sin(2\pi mt/T). \tag{B.8}$$

The FS coefficients of the excitation in terms of the FS coefficients of the speech are thus:

$$G_c(m) = S_c(m)\sum_{l=0}^{N_f}\alpha_l \cos(2\pi mlD/T) - S_s(m)\sum_{l=0}^{N_f}\alpha_l \sin(2\pi mlD/T)$$
$$G_s(m) = S_c(m)\sum_{l=0}^{N_f}\alpha_l \sin(2\pi mlD/T) + S_s(m)\sum_{l=0}^{N_f}\alpha_l \cos(2\pi mlD/T). \tag{B.9}$$

Proof of (B.2):

The FS coefficients $S_c(m)$ and $S_s(m)$ can be recovered from the FS coefficients of the excitation using the follwing expression:

$$S_c(m) = \frac{G_c(m)\sum_{l=0}^{N_f}\alpha_l \cos(\frac{2\pi mlD}{T}) + G_s(m)\sum_{l=0}^{N_f}\alpha_l \sin(\frac{2\pi mlD}{T})}{\left[\sum_{l=0}^{N_f}\alpha_l \cos(\frac{2\pi mlT}{T})\right]^2 + \left[\sum_{l=0}^{N_f}\alpha_l \sin(\frac{2\pi mlT}{T})\right]^2}$$
$$S_s(m) = \frac{-G_c(m)\sum_{l=0}^{N_f}\alpha_l \sin(\frac{2\pi mlD}{T}) + G_s(m)\sum_{l=0}^{N_f}\alpha_l \cos(\frac{2\pi mlD}{T})}{\left[\sum_{l=0}^{N_f}\alpha_l \cos(\frac{2\pi mlD}{T})\right]^2 + \left[\sum_{l=0}^{N_f}\alpha_l \sin(\frac{2\pi mlD}{T})\right]^2} \tag{B.10}$$

This is proven below by substitution. Consider the numerator for the first expression in (B.10):

$$G_c(m)\sum_{l=0}^{N_f}\alpha_l \cos(\frac{2\pi mlD}{T}) + G_s(m)\sum_{l=0}^{N_f}\alpha_l \sin(\frac{2\pi mlD}{T}) = \tag{B.11}$$
$$\left[S_c(m)\sum_{l=0}^{N_f}\alpha_l \cos(2\pi mlD/T) - S_s(m)\sum_{l=0}^{N_f}\alpha_l \sin(2\pi mlD/T) \right] \sum_{n=0}^{N_f}\alpha_n \cos(2\pi mnD/T)$$

$$+ \quad \left[S_c(m) \sum_{l=0}^{N_f} \alpha_l \sin(2\pi m l D/T) + S_s(m) \sum_{l=0}^{N_f} \alpha_l \cos(2\pi m l D/T) \right] \sum_{n=0}^{N_f} \alpha_n \sin(2\pi m n D/T).$$

The terms can be rearranged to give:

$$\begin{aligned} G_c(m) \sum_{l=0}^{N_f} \alpha_l \cos(\frac{2\pi m l D}{T}) + G_s(m) \sum_{l=0}^{N_f} \alpha_l \sin(\frac{2\pi m l D}{T}) = \\ S_c(m) \quad \left[\sum_{l=0}^{N_f} \alpha_l \cos(2\pi m l D/T) \sum_{n=0}^{N_f} \alpha_n \cos(2\pi m n D/T) \right. \\ \left. + \sum_{l=0}^{N_f} \alpha_l \sin(2\pi m l D/T) \sum_{n=0}^{N_f} \alpha_n \sin(2\pi m n D/T) \right] \\ + S_s(m) \quad \left[\sum_{l=0}^{N_f} \alpha_l \cos(2\pi m l D/T) \sum_{n=0}^{N_f} \alpha_n \sin(2\pi m n D/T) \right. \\ \left. - \sum_{l=0}^{N_f} \alpha_l \sin(2\pi m l D/T) \sum_{n=0}^{N_f} \alpha_n \cos(2\pi m n D/T) \right] \end{aligned} \tag{B.12}$$

The terms for $S_s(m)$ cancel out to leave:

$$\begin{aligned} G_c(m) \sum_{l=0}^{N_f} \alpha_l \cos(\frac{2\pi m l D}{T}) + G_s(m) \sum_{l=0}^{N_f} \alpha_l \sin(\frac{2\pi m l D}{T}) = \\ S_c(m) \left\{ \left[\sum_{l=0}^{N_f} \alpha_l \cos(2\pi m l D/T) \right]^2 + \left[\sum_{l=0}^{N_f} \alpha_l \sin(2\pi m l D/T) \right]^2 \right\} \end{aligned} \tag{B.13}$$

Dividing by the denominator in (B.10) recovers $S_c(m)$. Similarly, the expression for recovering $S_s(m)$ in (B.10) can be proven by substitution.

Now consider if the LPC coefficients in (B.10) are modified such that:

$$\tilde{\alpha}_l = \gamma^l \alpha_l, \quad 0 \le l \le N_f. \tag{B.14}$$

Then with $\gamma < 1$ a deemphasized formant structure is put back into the FS represen-

tation by the expression:

$$S_c(m) = \frac{G_c(m)\sum_{l=0}^{N_f}\gamma^l\alpha_l\cos(\frac{2\pi mlD}{T}) + G_s(m)\sum_{l=0}^{N_f}\gamma^l\alpha_l\sin(\frac{2\pi mlD}{T})}{\left[\sum_{l=0}^{N_f}\gamma^l\alpha_l\cos(\frac{2\pi mlT}{T})\right]^2 + \left[\sum_{l=0}^{N_f}\gamma^l\alpha_l\sin(\frac{2\pi mlT}{T})\right]^2}$$

$$S_s(m) = \frac{-G_c(m)\sum_{l=0}^{N_f}\gamma^l\alpha_l\sin(\frac{2\pi mlD}{T}) + G_s(m)\sum_{l=0}^{N_f}\gamma^l\alpha_l\cos(\frac{2\pi mlD}{T})}{\left[\sum_{l=0}^{N_f}\gamma^l\alpha_l\cos(\frac{2\pi mlD}{T})\right]^2 + \left[\sum_{l=0}^{N_f}\gamma^l\alpha_l\sin(\frac{2\pi mlD}{T})\right]^2}. \quad \text{(B.15)}$$

References

[1] I. Gerson and M. A. Jasiuk, "Vector sum excited linear prediction (VSELP) speech coding at 8 kbps," *Proc. Int. Conf. Acoust., Speech and Signal Process.*, Albuquerque, pp. 461–464, 1990.

[2] P. Mermelstein, "Outlook for high quality 4 kb/s speech coding for cellular mobile," *Proc. Global Telecomm. Conf.*, Phoenix, pp. 1865 1868, 1991.

[3] T. E. Tremain, J. P. Campbell, V. C. Welch, "Federal Standard (FED-STD) 1016: analog-to-digital conversion of voice by 4800 bits/sec code excited linear prediction (CELP) coding," U.S. Government, Dept. of Defense, R5, Fort Meale, Maryland 20755-600, U.S.A., Aug. 1989.

[4] D. O'Shaughnessy, *Speech Communication*, Addison-Wesley Series in Electrical Engineering, 1981.

[5] W. Granzow and B. S. Atal, "High quality digital speech at 4 kb/s," *Proc. Global Telecomm. Conf. Conf.*, San Diego, pp. 941-945, 1990.

[6] R. J. McAulay and T. F. Quatieri, "Speech analysis/synthesis based on a sinusoidal representation," *IEEE Trans. Acoust., Speech and Signal Process.*, vol. ASSP-34, pp. 744–754, Aug. 1986.

[7] W. B. Kleijn, "Continuous representations in linear predictive coding," *Proc. Int. Conf. Acoust., Speech and Signal Process.*, Toronto, pp. 201 204, 1991.

[8] W. B. Kleijn, "Speech coding below 4 kb/s using waveform interpolation," *Proc. Global Telecomm. Conf.*, Phoenix, pp. 1879–1883, 1991.

[9] W. B. Kleijn, "Methods for waveform interpolation in speech coding," *Digital Signal Processing*, pp. 215–230, Sept. 199.

[10] W. B. Kleijn, "Analysis-by-synthesis coding based on relaxed waveform-matching constraints," Ph.D. thesis, Delft University of Technology, Holland, Dec. 1991.

[11] R. Ramachandran and P. Kabal, "Pitch prediction filters in speech coding," *IEEE Trans. Acoust., Speech and Signal Process.*, vol. ASSP-37, pp. 467–478, April 1989.

[12] R. Ramachandran and P. Kabal, "Stability and performance analysis of pitch filters in speech coders," *IEEE Trans. Acoust., Speech and Signal Process.*, vol. ASSP-35, pp. 937–945, July 1987.

[13] P. Kroon and B. S. Atal, "Pitch predictors with high temporal resolution," *Proc. Int. Conf. Acoust, Speech and Signal Process.*, Albuquerque, pp. 661–664, 1990.

[14] F. K. Soong and B. H. Juang, "Optimal quantization of LSP parameters," *Proc. Int. Conf. Acoust., Speech and Signal Process.*, New York, pp. 394–397, 1988.

[15] H. W. Schussler, "A stability theorem for discrete systems," *IEEE Trans. Acoust., Speech and Signal Process.*, vol. ASSP-24, pp. 87–89, Feb. 1976.

[16] F. K. Soong and B. H. Juang, "Line spectrum pair (LSP) and speech data compression," *Proc. Int. Conf. Acoust., Speech, and Signal Process.*, San Diego, pp. 1.10.1–1.10.4, 1984.

[17] G. S. Kang and L. S. Fransen, "Application of line spectrum pairs to low bit rate speech encoders," *Proc. Int. Conf. Acoust., Speech and Signal Process.*, Tampa, pp. 7.3.1–7.3.4, 1985.

[18] P. Kabal and R. Ramachandran, "The computation of line spectral frequencies using Chebyshev polynomials," *IEEE Trans. Acoust., Speech and Signal Process.*, vol. ASSP-34, pp. 1419–1426, Dec. 1986.

[19] J. H. Chen and A. Gersho, "Real-time vector APC speech coding at 4800 bps with adaptive postfiltering," *Proc. Int. Conf. Acoust., Speech and Signal Process.*, Dallas, pp. 2185–2188, 1987.

[20] J. S. Marques, I. M. Trancoso, J.M. Tribolet, and L.B. Almeida, "Improved pitch prediction with fractional delays in CELP coding," *Proc. Int. Conf. Acoust., Speech and Signal Process.*, Albuquerque, pp. 661-664, 1990.

[21] M. M. Sondhi, "New methods of pitch extraction," *IEEE Trans. Audio and Electroacoustics*, vol. AU-16, pp. 262-266, June 1968.

[22] J. D. Markel, "The SIFT algorithm for fundamental frequency estimation," *IEEE Trans. Audio and Electroacoustics*, vol. AU-20, pp. 367 377, Dec. 1972.

[23] K. K. Paliwal and B. S. Atal, "Efficient vector quantization of LPC parameters at 24 bits/frame," *Proc. Int. Conf. Acoust., Speech and Signal Process.*, Toronto, pp. 661–664, 1991.

[24] Y. Shoham, "Constrained-stochastic excitation coding of speech at 4.8 kb/s," in *Advances in Speech Coding*, ed. B.S. Atal, V. Cuperman and A. Gersho, Kluwer Academic Publishers, Dordrecht, Holland, pp. 339-348, 1991.

[25] T. Taniguchi, M. Johnson and Y. Ohta, "Pitch sharpening for perceptually improved CELP, and the sparse-delta codebook for reduced computation," *Proc. Int. Conf. Acoust., Speech and Signal Process.*, Toronto, pp. 241 244, 1991.