Pitch Filtering in Adaptive Predictive Coding of Speech

by

Ravi Prakash Ramachandran
B.Eng. (Electrical)

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
Montréal, Canada
March, 1986

Abstract

This thesis investigates the problem of stability of pitch filters in speech coding. The concern is on adaptive predictive coders employing pitch predictors.

A new algorithm that estimates the pitch period is coupled with the covariance formulation of determining the pitch predictor coefficients in order to realize a transversal structured filter. Since this approach does not guarantee the stability of the corresponding synthesis filter, a computationally efficient stability test based on a simple but tight sufficient condition is formulated. This is much less computationally demanding than utilizing a set of necessary and sufficient conditions derived from known stability tests. From the sufficient condition, a stabilization technique that ensures a stable pitch filter is introduced. An alternate method of deriving the filter coefficients such that stability is guaranteed at the outset is obtained by applying the Burg algorithm in realizing a lattice structured filter. For a lattice predictor, the pitch period is estimated in a different way than for a transversal predictor. The effect of the presence of unstable pitch filters on decoded speech is also investigated.

At the analysis stage, the formant and pitch predictors may be placed in either order. Both configurations are compared with regards to the stability and performance of pitch filters. Recommendations for future research are given.

Sommaire

Ce mémoire examine les problèmes de stabilité dans les filtres de périodicité tels qu'employés par les systèmes de codage prédictif adaptif avec prédiction du fondamental.

On propose un nouvel algorithme de détection du fondamental qui, lorsque combiné avec la méthode de calcul par covariance des coefficients du prédicteur, permet la réalisation d'un filtre à structure transverse. Puisque cette approche ne garantit pas la stabilité du filtre de synthèse correspondant, on formule un test de stabilité qui est basé sur une condition suffisante simple mais sévère et qui est efficace du point de vue du temps de calcul. Cette solution est beaucoup plus efficace que ne l'est l'utilisation d'un ensemble de conditions nécessaires et suffisantes obtenu à partir des tests de stabilité déjà connus. En se basant sur la condition suffisante, on propose une technique qui assure la stabilité du filtre de synthèse. Si l'on applique l'algorithme de Burg lors de la réalisation du filtre en treillis, on parvient à une autre solution qui, elle, garantit que les coefficients produits correspondent à un filtre stable. On estime le fondamental de façon différente pour un prédicteur en treillis que pour un filtre à structure transverse. Enfin, on analyse l'effet qu'a la présence d'instabilités dans les filtres de synthèse sur le signal décodé.

Au niveau de l'analyse, on peut placer les prédicteurs de formants et de fondamental dans n'importe quel ordre. Ce travail compare ces deux configurations au niveau de la stabilité de leurs filtres de synthèse et de la performance de leurs filtres de prédiction du fondamental. Finalement, on offre des recommendations au sujet de recherches futures possibles.

Acknowledgements

I would like to sincerely thank my supervisor Dr. Peter Kabal for his invaluable guidance. The laboratory facilities given by INRS is greatly appreciated. Also, a special thanks goes to Dr. Johnson for his help in using the computer to generate the figures for this thesis.

The scholarship awarded by the Natural Sciences and Engineering Research Council of Canada is appreciated. I owe a very special thanks to my parents for their continuous support and encouragement throughout my studies.

Table of Contents

List of Figures

List of Tables

Chapter 1 Introduction

1.1 Adaptive Predictive Coding of Speech

Speech coding concerns itself with transmitting a speech waveform over a digital channel. Although a speech waveform is not necessarily bandlimited, lowpass filtering ensures that the resulting waveform has essentially a finite bandwidth. The cutoff frequency of the filter is chosen so that this finite bandwidth includes perceptually important frequencies in the speech signal. The finite bandwidth of the filtered waveform allows it to be sampled at a rate which avoids aliasing at the receiver. After obtaining the sampled signal, the goal is to code the speech at a low bit rate while maintaining high quality in the reconstructed speech.

The required bit rate can be lowered by exploiting redundancies in the speech signal. These redundant components can be predicted from the recent history of the signal. This concept is known as predictive coding and was pioneered by Atal in the early 1970's [1,2].

In predictive coding, two types of prediction are used. One is based on the spectral envelope (formant prediction) and the other based on the spectral fine structure (pitch prediction). Formant prediction removes near-sampled-based redundancies introduced by the vocal tract filtering while pitch prediction removes distant-sampled-based redundancies caused by the rhythmic glottal excitation (pitch period excitation) [3].

Both formant and pitch predictors must be adaptive since a speaker's vocal tract shape and pitch are time varying. Therefore, the coefficients of these predictors change within a short period of time and are reset every 10-25 milliseconds. The use of adaptive predictors in speech coding gives rise to the name *Adaptive Predictive Coding* (APC). The performance measure for a predictor is the prediction gain. The prediction gain is the ratio of the energies of the input and output signals of the predictor. In APC, the formant and pitch predictors may be cascaded in either order. However, it has been observed that the total prediction gain of either serial combination of the predictors is less than the sum of the prediction gains of each predictor acting alone [4].

After performing effective formant and pitch prediction, a residual signal is obtained. Effective prediction is achieved when the coefficients of the filters are chosen with the objective of minimizing the energy of the residual. The operation of an APC system (details are given in next chapter) involves the transmission of the quantized residual, the predictor coefficients and the step size of the quantizer. The transmission of the residual occupies the largest portion of the total number of bits used, but also requires fewer bits/sample than the original speech signal. The quantizer for the residual signal often uses non-uniformly spaced step sizes.

A derivative of APC, namely CELP (Code-Excited Linear Prediction) [5], does not use sample-by-sample quantization of the residual (details given in next chapter). The residual signal, after gain normalization, has a cumulative probability distribution closely resembling a Gaussian distribution with the same mean and variance [6]. Therefore, the residual is tested for its resemblance to one of many Gaussian waveforms stored in a dictionary. This is equivalent to a vector quantization scheme. It is the index of the selected waveform in the dictionary that is transmitted.

1.1.1 Formant and Pitch Predictors

Both the formant predictor and the pitch predictor are FIR (finite impulse response) digital filters that are usually implemented in direct form or lattice form. The order of a formant predictor is typically between 8 and 16 for low bit rate encoders. Simple pitch prediction uses only one coefficient (one tap prediction) where the tap delay of the predictor corresponds to the estimated pitch period. However, a 3 tap predictor performs better than a 1 tap predictor. In a 3 tap predictor, the middle coefficient is associated with a delay equal to the estimated pitch period. A 3 tap predictor also allows interpolation of the speech samples in the delayed version to more precisely match the undelayed version. This is very useful since the pitch period is not necessarily an integral number of samples.

At the transmitter, the stability of the formant and pitch predictors is guaranteed since they are FIR filters. At the receiver, formant and pitch synthesis filters restore the near-sampled-based and distant-sampled-based redundancies to yield the decoded speech waveform. These synthesis filters are all-pole IIR filters and hence, stability is an issue. The stability of a formant synthesis filter is guaranteed if the predictor coefficients are derived using the autocorrelation [7] or modified covariance method [8].

In practice, unstable pitch synthesis filters arise frequently since the analysis techniques do not guarantee their stability. This thesis provides a stability test for 1, 2 and 3 tap pitch synthesis filters based on a tight sufficient condition and devises analysis algorithms that guarantee stability. The effect of instability on decoded speech is also investigated.

1.2 Scope and Organization of the Thesis

The entire thesis is organized into six chapters. After the introduction, Chapter 2 describes two speech coders, APC and CELP.

Chapter 3 deals with the stability issues that emerge when a pitch filter is used in these coders. First, the cause of instability is examined. Then, known stability tests are discussed briefly. Finally, a stability test for 2 and 3 tap filters is formulated. This test is based on a tight sufficient condition and can be implemented in a real-time environment.

In Chapter 4, experiments involving a CELP system in which a formant predictor is followed by a pitch predictor are performed. A new method of estimating the pitch period is derived. This method can be coupled with the covariance formulation to calculate the pitch predictor coefficients. Since the solution does not guarantee stability of the pitch filter, a technique to stabilize the filter while minimizing the loss in prediction gain is suggested. The Burg algorithm is also used to generate a lattice structured pitch filter which is guaranteed to be stable. The prediction gains achieved by the stabilization and the Burg methods are compared with that of the covariance algorithm. Finally, the effect of having unstable pitch filters on decoded speech is examined.

Chapter 5 deals with a CELP coder that has a pitch predictor followed by a formant predictor. This arrangement is compared with the previous configuration in order to determine which arrangement of the two filters is superior. The chapter ends with some suggestions for future research. Chapter 6 records the conclusions of the investigation.

Chapter 2 APC and CELP Coders

This chapter describes the operation of APC and CELP coders. Both coders use formant and pitch predictors to remove the redundancies in the speech signal. Formant predictors have a transfer function:

$$F(z) = \sum_{k=1}^{p} a_k z^{-k} \tag{2.1}$$

The order p is typically between 8 and 16 for low bit rate coding. Pitch predictors consist of 1, 2 or 3 taps having transfer functions:

$$P(z) = \begin{cases} \beta z^{-M} & \text{1 tap} \\ \beta_1 z^{-M} + \beta_2 z^{-(M+1)} & \text{2 tap} \\ \beta_1 z^{-(M-1)} + \beta_2 z^{-M} + \beta_3 z^{-(M+1)} & \text{3 tap} \end{cases} \tag{2.2}$$

where M is the estimated pitch period in samples. Therefore, a pitch predictor has a small number of taps that are placed after a long delay and centered around the pitch period. The polynomials $1 - F(z)$ and $1 - P(z)$ are the transfer functions of the prediction error filters. At the synthesis stage, a formant synthesis filter $H_F(z)$ and pitch synthesis filter $H_P(z)$ are used. Their transfer functions are:

$$H_F(z) = \frac{1}{1 - F(z)} \quad \text{and} \quad H_P(z) = \frac{1}{1 - P(z)} \tag{2.3}$$

2.1 APC System

To clearly understand the operation of an APC system, first assume that only a formant predictor is used. The block diagram of such a coder is shown below.

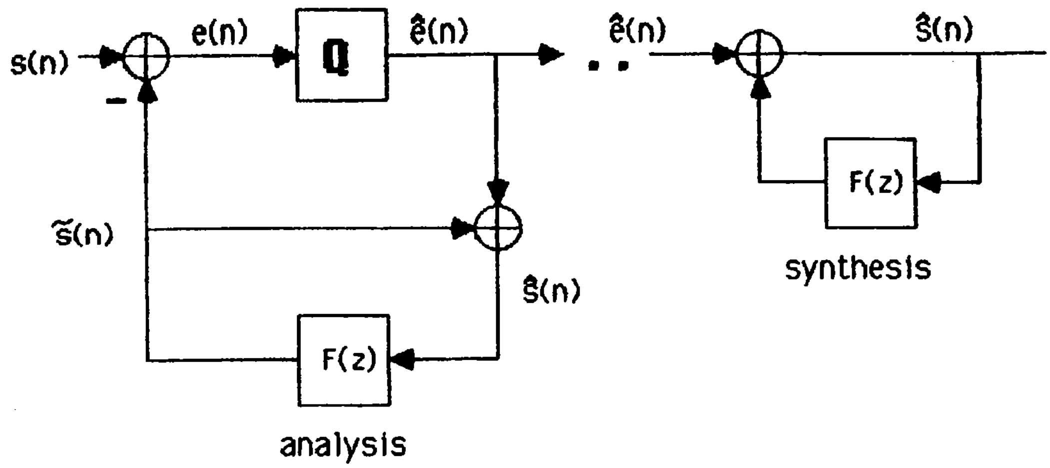


Fig. 2.1 Block Diagram of an APC Coder

The formant predictor is part of the feedback loop and predicts the input signal $s(n)$ on the basis of previously reconstructed samples $\hat{s}(n-k)$ for $k = 1$ to p. The error signal $e(n)$ is expressed as:

$$e(n) = s(n) - \tilde{s}(n) \tag{2.4}$$

The predicted signal $\tilde{s}(n)$ is a weighted sum of previously reconstructed samples:

$$\tilde{s}(n) = \sum_{k=1}^{p} a_k \hat{s}(n-k) \tag{2.5}$$

The error signal is quantized to give $\hat{e}(n)$. Then, any reconstructed sample $\hat{s}(n)$ is given by:

$$\begin{aligned} \hat{s}(n) &= \hat{e}(n) + \tilde{s}(n) \\ &= \hat{e}(n) + \sum_{k=1}^{p} a_k \hat{s}(n-k) \end{aligned} \tag{2.6}$$

Manipulating the above equations give:

$$\begin{aligned} \hat{e}(n) &= \hat{s}(n) - \sum_{k=1}^{p} a_k \hat{s}(n-k) \\ &= (\hat{s}(n) - s(n)) + (s(n) - \sum_{k=1}^{p} a_k \hat{s}(n-k)) \\ &= e(n) + q(n) \end{aligned} \tag{2.7}$$

The quantity $q(n)$ is the quantization error. This error is interpreted as being a source of additive noise. Assuming an error-free channel, this is the only noise introduced in the APC system.

The synthesis step is merely the inverse operation of the analysis step. Since the only source of noise is quantization error, the reconstructed speech signal $\hat{s}(n)$ is expressed as:

$$\hat{s}(n) = s(n) + q(n) \tag{2.8}$$

Therefore, the reconstructed samples are equal to the original samples plus the noise arising from the quantization of the error signal.

Figure 2.2 shows an APC coder with both a formant and pitch predictor. Here, formant prediction is done before pitch prediction. The positions of the two predictors may be interchanged. The mathematical development of the analysis is just an extension of the development given for a coder with just a formant predictor. Again, the only source of noise is quantization error.

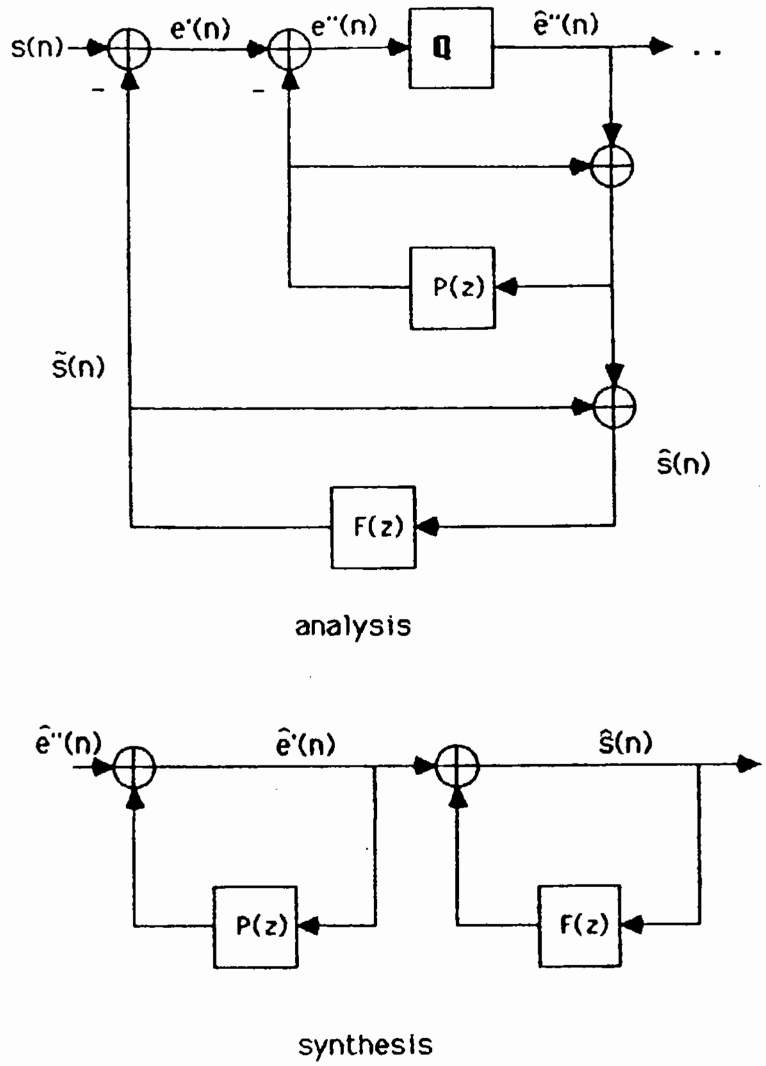
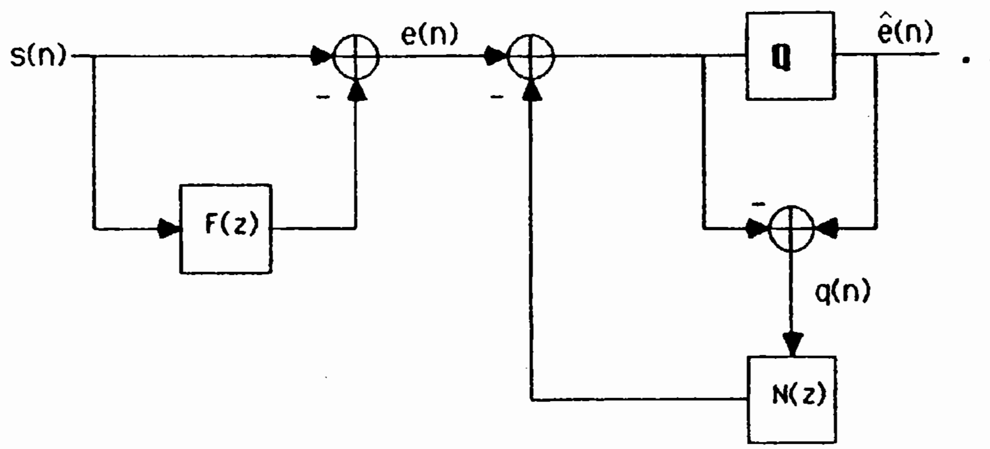


Fig. 2.2 Complete APC Coder

2.1.1 Noise Shaping

Modification of the spectrum of the quantization error can reduce the perceptual distortion in the decoded speech [8], [9]. An alternate description of the APC coder with a noise feedback filter $N(z)$ is shown in Fig. 2.3 [10]. The open-loop predictor configuration is used in CELP (discussed in next section) for both the formant and pitch filters. The quantization error is fed to a feedback filter $N(z)$. The output of this filter is subtracted from the residual and again fed to the quantizer. It can be shown that the spectrum of the coder output noise can be modeled by the quantization error filtered by $(1 - N(z))/(1 - F(z))$.



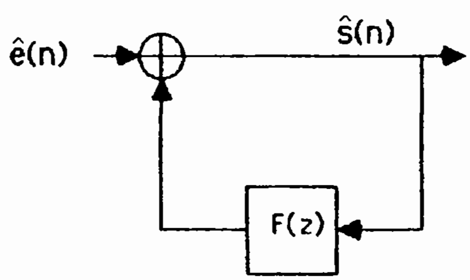


Fig. 2.3 APC Coder with Noise Shaping

The filter $N(z)$ is often selected to be $N(z) = F(z/\alpha)$ where the weighting factor α is between 0 and 1. If $\alpha = 0$, then $N(z) = 0$ and the quantization error has the same spectral envelope as the original speech. If $\alpha = 1$. then $N(z) = F(z)$ and the noise retains its flat spectrum. This produces a high SNR (signal-to-noise ratio) at the formant frequencies and a low SNR at the valleys of the spectral envelope of the speech. By selecting α between 0 and 1 (common values are between 0.75 and 0.85), the SNR is improved in the valleys but at the expense of decreased SNR at the formants. Although the overall unweighted SNR decreases if α is decreased from 1, noise shaping reduces the perceptual distortion of the output speech.

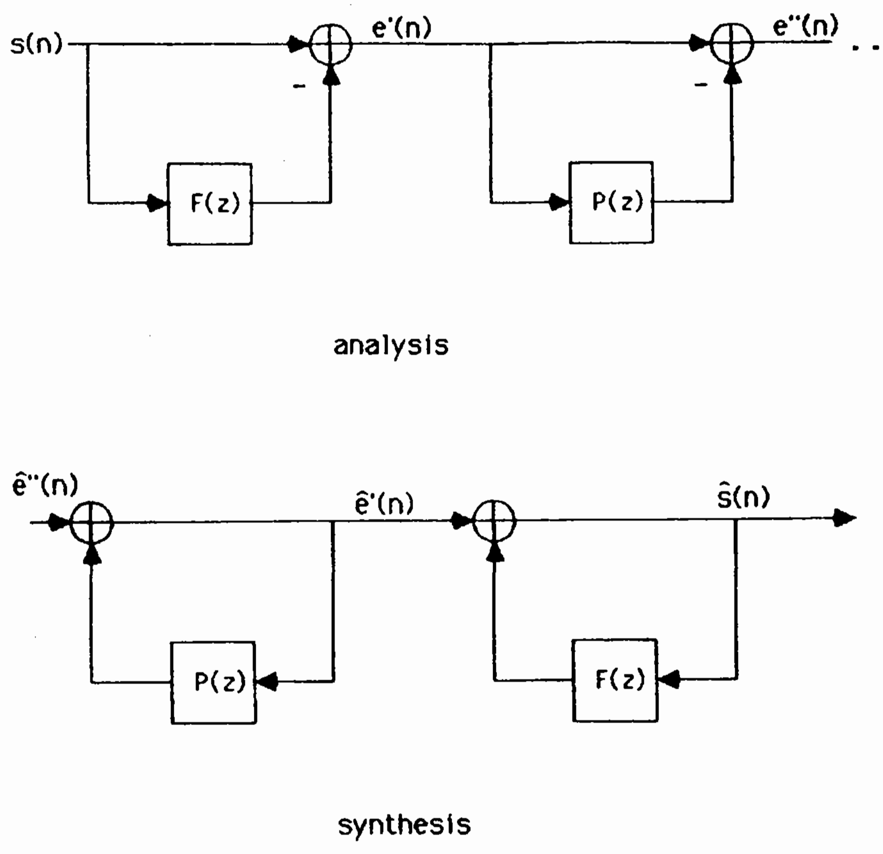


Fig. 2.4 Block Diagram of CELP System

2.2 Code Excited Linear Predictive (CELP) Coding

Figure 2.4 illustrates a CELP coder. In CELP [5], a residual signal is generated by formant and pitch prediction. The residual (after gain normalization) is compared to each waveform in a codebook constructed of Gaussian random numbers with unit variance. A Gaussian distribution is used since the gain normalized residual signal has a distribution which is nearly Gaussian.

To perform the comparison, each entry in the codebook is first filtered by a pitch synthesis and formant synthesis filter and subtracted from the original speech signal to form a difference signal. The difference signal is then passed through a perceptual weighting filter $W(z)$ where:

$$W(z) = \frac{1 - F(z)}{1 - N(z)} \tag{2.9}$$

This filter de-emphasizes the frequencies which contribute less to perceptual error and emphasizes the frequencies which contribute more to perceptual error. Then, the weighted mean-squared error is formed by squaring and averaging the filtered difference signal. This entire process is depicted in Fig. 2.5.

The entry in the codebook that gives the least perceptual error represents the residual. Its index is transmitted. This is equivalent to a vector quantization scheme. The synthesis operation is the inverse of the analysis operation. The codeword representing the residual is filtered by $H_P(z)$ and $H_F(z)$ to generate the decoded speech.

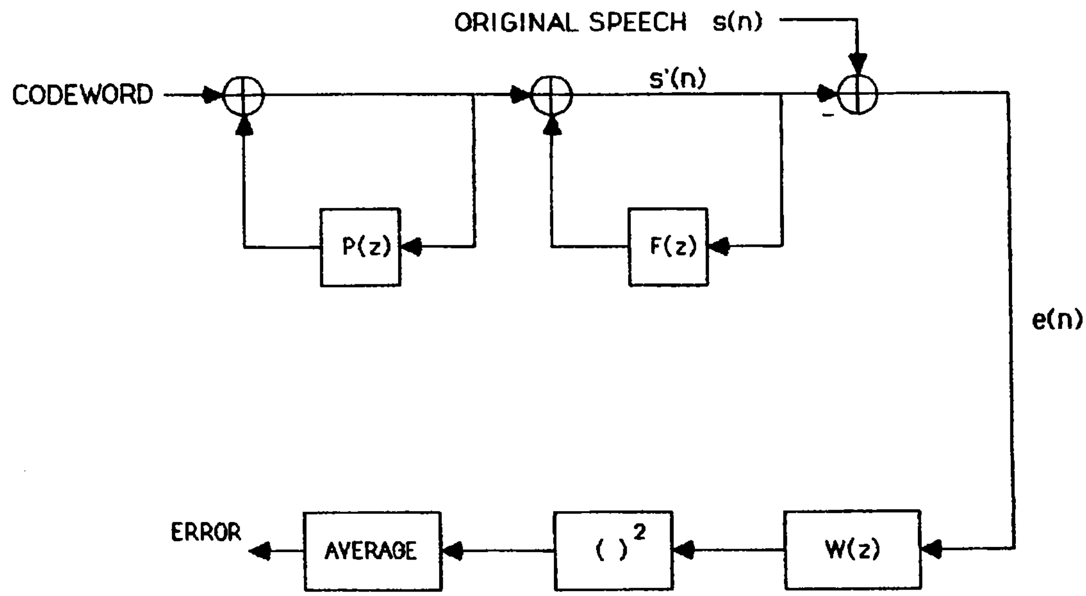


Fig. 2.5 Calculating the Weighted Error

2.3 Similarities and Differences Between APC and CELP

In both APC and CELP, formant and pitch prediction generate a residual signal. In an APC coder, prediction is based on past reconstructed samples whereas in CELP, prediction is based directly on the past input samples.

Transmission of the predictor coefficients occurs both in APC and CELP. Usually, a transformation of these coefficients to a new set of quantities having superior quantization properties (such as reflection coefficients or line spectral frequencies for formant predictors) is performed before quantization. Scalar quantization of each residual sample is done in APC whereas a vector quantization scheme is used in CELP to code a block of residual samples.

A weighting filter that modifies the noise spectrum can be used in both coders although the implementation of this filter as part of the overall system is different. In APC, the only source of error is quantization error. Similarly, it can be shown that the output of a CELP coder consists of the original speech signal plus quantization error.

Unlike APC, CELP requires the storage of a codebook consisting of different waveforms with a Gaussian distribution. This imposes a large memory requirement on the system. A considerable computational requirement is also present since the residual must be compared with each waveform in a codebook. However, CELP can code speech at lower bit rates but at the expense of extra computation and memory.

Chapter 3 Stability Analysis of Pitch Filters

This chapter deals with the stability of pitch synthesis filters. For analysis frame sizes of interest, using either the modified covariance or autocorrelation methods in determining the coefficients of a formant predictor ensures the stability of $H_F(z)$ without significant loss in prediction gain as compared to the optimal covariance method. No equivalent algorithm for a pitch predictor that gives a stable $H_P(z)$ is available. Since an unstable $H_P(z)$ can occur in some frames of speech, this chapter introduces a stability test for pitch synthesis filters.

3.1 Definition of Stability

Consider a digital filter with rational transfer function $H(z)$ where:

$$H(z) = \frac{C(z)}{D(z)} \tag{3.1}$$

The impulse response of $H(z)$ is a discrete-time sequence $h(n)$. A stable system is one where every bounded input $d(n)$ produces a bounded output $c(n)$. This is referred to as bounded-input bounded-output (BIBO) stability. An equivalent condition tests the impulse response [11]:

$$\sum_{k=-\infty}^{\infty} |h(k)| < \infty \tag{3.2}$$

In the z-domain, the condition for BIBO stability is that the poles of $H(z)$ or equivalently the zeros of $D(z)$ lie within the unit circle, $|z| = 1$. For low order $D(z)$ (order ≤ 10), one can use root finding algorithms to determine the location of the zeros and establish whether or not all are within the unit circle. Pitch synthesis filters have an order around M, where M is the estimated pitch period in samples. The pitch period is in the range 25 to 80 for female speakers and 40 to 110 for male speakers when the sampling frequency is 8 kHz. Determining the root positions of such a large degree polynomial is not practical even though it has only a small number of non-zero coefficients. For implementation as part of a coding system, a stability test should be simple and avoid excessive computational overhead on the coder. A simple stability test that can be implemented as part of the overall coding system is introduced.

3.2 Origins of Instability

In a coder, it is useful to know how an unstable pitch synthesis filter is encountered. For purposes of simplicity, this is illustrated for a 1 tap filter. However, the concept carries over to filters with a larger number of taps.

A 1 tap pitch synthesis filter has a system function:

$$H_P(z) = \frac{1}{1 - \beta z^{-M}} = \frac{z^M}{z^M - \beta} \tag{3.3}$$

In this case the order of the denominator polynomial is very high but solving for its roots is simple. The zeros of the denominator polynomial are given by:

$$z_k = |\beta|^{\frac{1}{M}} \exp(\frac{j2\pi k}{M}) \quad \text{for } k = 1 \text{ to } M \tag{3.4}$$

In the z-plane, the zeros z_k form a circle of radius $|\beta|^{\frac{1}{M}}$ and are separated form each other in angular frequency by $\frac{2\pi}{M}$ radians. For all the zeros z_k to be within the unit circle, it is necessary and sufficient that $|\beta| < 1$.

To investigate how instability occurs, assume that the pitch prediction is performed on a signal $s(n)$. In a 1 tap filter, β is obtained by minimizing the energy of the residual over a block of N samples,

$$\beta = \frac{\sum_{k=1}^{N} s_k s_{k-M}}{\sum_{k=1}^{N} s_{k-M}^2} \tag{3.5}$$

The derivation of the above expression and the determination of the value of M are discussed in the next chapter. Here, only the issue of stability is being studied. An unstable pitch synthesis filter arises when the absolute value of the numerator of Eq. (3.5) is greater than the denominator ($|\beta| > 1$). This usually arises when a transition from an unvoiced to a voiced segment takes place. Such a transition is marked by an increase in the signal energy. When processing a voiced frame that occurs just after an unvoiced frame, the denominator quantity $\sum s_{k-M}^2$ involves the sum of squares of amplitudes in the unvoiced segment and does not reach a very large value. On the other hand, the numerator quantity $\sum s_k s_{k-M}$ involves the sum of products of the higher amplitudes from the voiced frame and the lower amplitudes from the unvoiced frame. Therefore, often the numerator can be larger in magnitude than the denominator giving $|\beta| > 1$.

From the analysis above, one can conclude that unstable pitch synthesis filters can arise when the signal energy shows a sudden increase. Figure 3.1 shows an actual speech waveform, the energy in each analysis frame and indicates frames having unstable 1 tap pitch filters (those with a non-zero indicator function).

Figure 3.2 shows an enlarged plot of frames 16 to 18 where a transition from an unvoiced frame (16) to a voiced frame (17) occurs. An increase in signal energy is depicted and the voiced frame 17 has an unstable pitch synthesis filter. This type of root cause of instability in 1 tap pitch synthesis filters carries over to 2 and 3 tap filters.

3.3 Stability Tests

A simple stability test is not available for 2 or 3 tap pitch filters although there are many different tests that may be used to obtain a set of necessary and sufficient conditions. A pitch synthesis filter has a denominator polynomial which is sparse (high order but with only few non-zero coefficients). A test that takes advantage of this sparse nature is the Schur-Cohn test [12,13]. Appendix A gives the general form of this test and shows how it can be applied to pitch synthesis filters. The implementation is computationally efficient.

Another form of the Schur-Cohn test [12] is given in matrix form. Given a polynomial of order n, the determinants of a set of $2k$ by $2k$ matrices (for $k = 1$ to

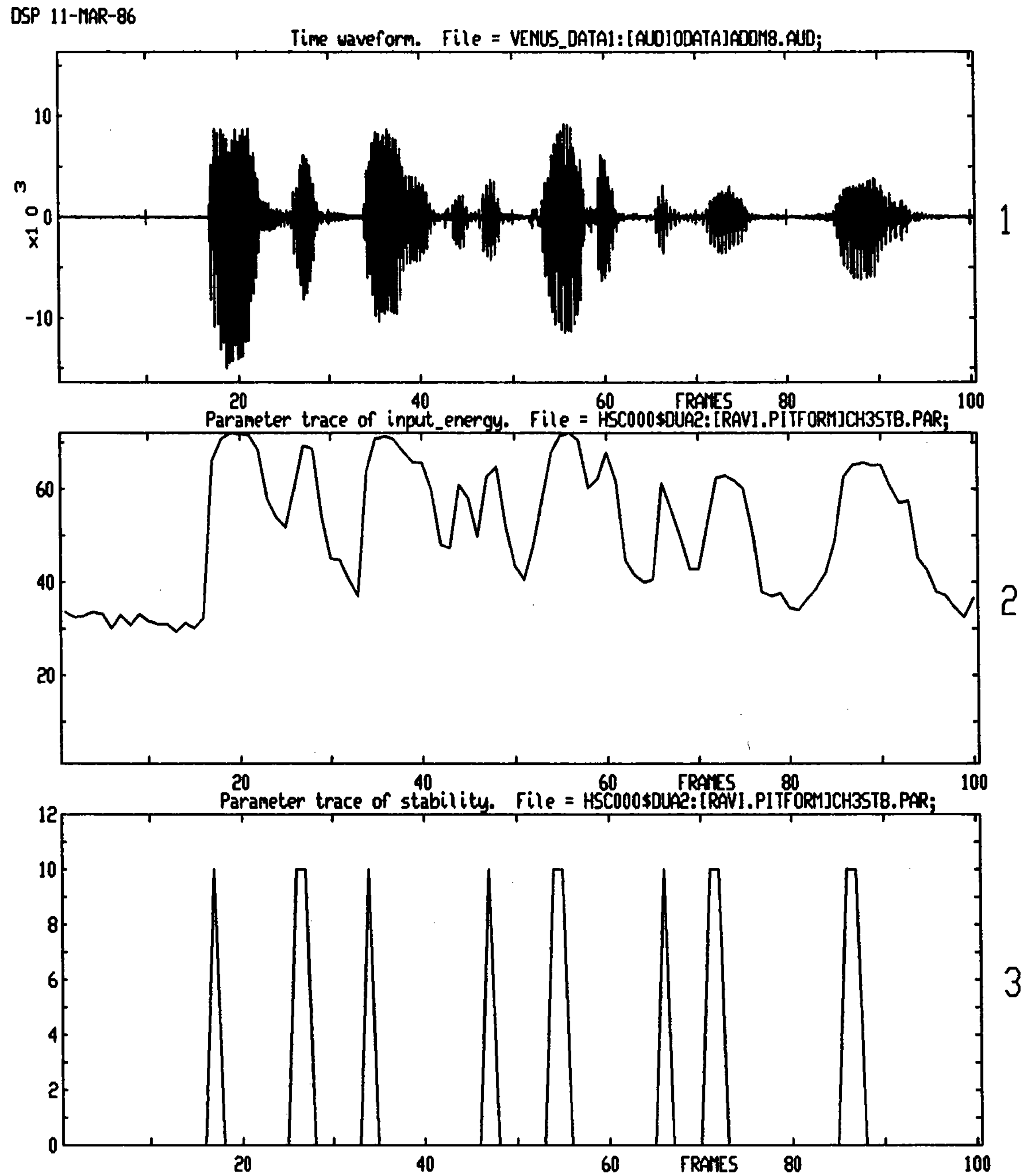


Fig. 3.1 Indications of Instability (1) Speech Data, (2) Input Energy in Each Frame (db) and (3) Frames Having Unstable Pitch Synthesis Filters

n) must be evaluated. To ensure stability, the determinants must alternate in sign with the first one being negative. For pitch filters, the $2k$ by $2k$ matrices can be

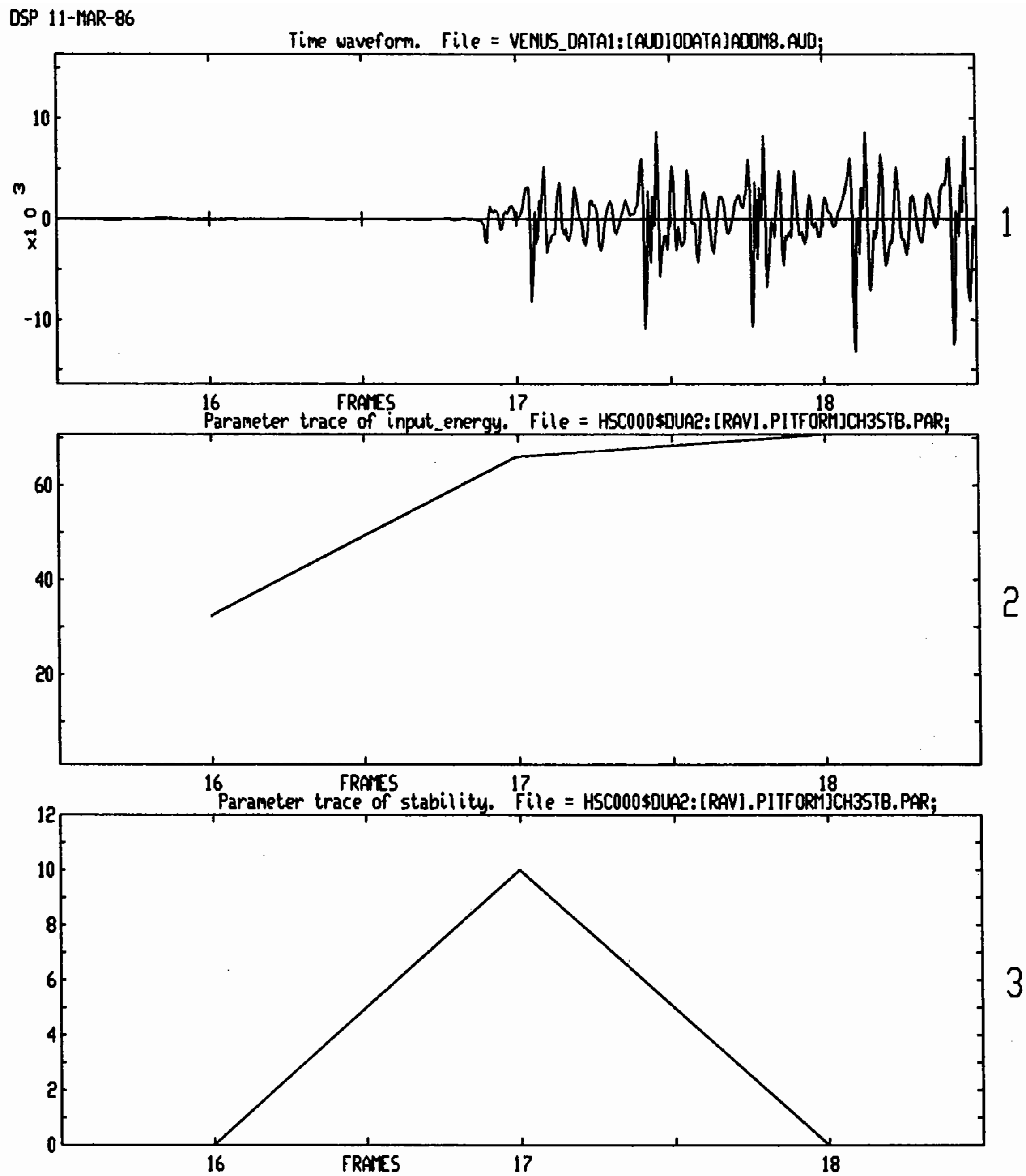


Fig. 3.2 Enlarged Plot of Frames 16 to 18 (1) Speech Data, (2)Input Energy in Each Frame (dB) and (3) Frames Having Unstable Pitch Synthesis Filters

transformed into k by k matrices without altering their determinants. These new matrices are tridiagonal and pentadiagonal for 2 and 3 tap filters respectively. The determinants of a tridiagonal matrix can be evaluated by a simple recursion. The pentadiagonal matrices can be recursively reduced to tridiagonal form by elementary row and column transformations to facilitate the calculation of the determinants. No actual storage of the matrices is necessary. Although the determinant form of the Schur-Cohn test can be reduced in complexity for pitch filters, the original form is still easier to use. Jury's simplifications of the Schur-Cohn test complicate the formulation for this problem and are therefore not considered.

Schussler's stability theorem [14,15] also gives a set of necessary and sufficient conditions. A symmetric and anti-symmetric polynomial is defined in terms of the denominator polynomial. If the roots of the symmetric and anti-symmetric polynomials are simple, on the unit circle and separate each other, the denominator polynomial has all its roots within the unit circle. If this theorem is used, both the symmetric and anti-symmetric polynomials formed are of high order. Finding their roots and ensuring that they are on the unit circle, are simple and separate each other requires a large amount of computation even if the symmetric and anti-symmetric nature of the polynomials is exploited. Recently, a test in table form that uses the symmetric and anti-symmetric polynomials defined by Schussler has been proposed [16]. In the case of pitch filters, it is not as efficient as the Schur-Cohn test since it does not take advantage of the sparse nature of the characteristic polynomial.

Even though a computationally efficient implementation of the Schur-Cohn test is possible (described in Appendix A), a comparatively simple alternative test is now formulated. This alternative test is based on a tight sufficient condition and is used in the experiments. This does not mean that the implementation of the Schur-Cohn test should not be used. If it is used, a set of necessary and sufficient conditions are available at the cost of increased computational load. The computational load increases with the order of the filter. Also, if the simple sufficient test is used and the filter is judged to be possibly unstable, stabilization is very easy (this idea is explored in Chapter 4).

3.4 Stability Test for Pitch Filters

In this section, a stability test based on a tight sufficient condition is presented. The advantages of the stability test are that it is not computationally demanding, does not involve the evaluation of transcendental functions and is independent of the pitch period (approximately equal to the highest degree or order of the characteristic polynomial). Independence of the order is useful since in each analysis frame, the estimated pitch period can change.

3.4.1 Sufficient Test

Consider a general denominator polynomial $D(z)$ of the form,

$$D(z) = z^n - B(z) \tag{3.6}$$

where

$$B(z) = \sum_{i=0}^{n-1} b_i z^i. \tag{3.7}$$

Then, $D(z) = z^n - B(z) = z^n(1 - z^{-n}B(z))$. The polynomial $D(z)$ has roots $z_k = r_k e^{jw_k}$ in the z-plane. It is desired that all the roots z_k be inside the unit circle. Therefore, the conditions are that $1 - z^{-n}B(z) \neq 0$ or $z^{-n}B(z) \neq 1$ on and outside the unit circle $z = e^{j\theta}$. By the maximum modulus theorem [17], $z^{-n}B(z)$ has its maximum modulus on the contour surrounding any region in which it is analytic. The expression $z^{-n}B(z)$ being a polynomial in z^{-1} is analytic on and outside the unit circle in the z-plane. Therefore, a sufficient condition for stability is that $|z^{-n}B(z)| < 1$ on and outside the unit circle in the z-plane. This condition is further expressed as

$$|z^{-n}B(z)| < 1 \quad \text{for} \quad z = e^{j\theta} \tag{3.8}$$

or simply

$$|B(e^{j\theta})| < 1. \tag{3.9}$$

For $B(z)$ as defined in Eq. (3.7),

$$\begin{aligned} |B(e^{j\theta})| &= |b_0 + b_1 e^{j\theta} + \cdots + b_{n-1} e^{j(n-1)\theta}| \\ &\leq |b_0| + |b_1| + \cdots + |b_{n-1}| \end{aligned} \tag{3.10}$$

The sufficient condition for stability is that the sum of the moduli of the coefficients is less than 1. This simple test can be applied to any predictor. Specifically for pitch filters, the sufficient condition for stability becomes:

$$\begin{aligned} |\beta| &< 1 \quad \text{1 tap} \\ |\beta_1| + |\beta_2| &< 1 \quad \text{2 tap} \\ |\beta_1| + |\beta_2| + |\beta_3| &< 1 \quad \text{3 tap} \end{aligned} \tag{3.11}$$

By performing a more detailed examination of the expression $|B(e^{j\theta})|$, an attempt to tighten the sufficient condition is made. The goal is to develop a test that is independent of the order and asymptotically tight. For 1 tap predictors, the condition is both necessary and sufficient as shown earlier. For 2 tap filters, it will be shown that the test is also asymptotically necessary ($n \rightarrow \infty$). A tighter sufficient test is developed for 3 tap pitch predictors.

3.4.2 Tight Sufficient Test

2 Tap Filters

Although the test for 3 tap filters subsumes the test for 2 tap filters, a detailed derivation of the 2 tap case is given in order to illustrate how the expression $|B(e^{j\theta})|$ is examined.

For a 2 tap filter:

$$\begin{aligned} B(z) &= \beta_1 z + \beta_2 \\ &= \sqrt{z}(\sqrt{z}\beta_1 + \beta_2/\sqrt{z}) \end{aligned} \tag{3.12}$$

Then, on the unit circle:

$$B(e^{j\theta}) = e^{j\frac{\theta}{2}}\Big[(\beta_1+\beta_2)\cos\frac{\theta}{2} + j(\beta_1-\beta_2)\sin\frac{\theta}{2}\Big] \tag{3.13}$$

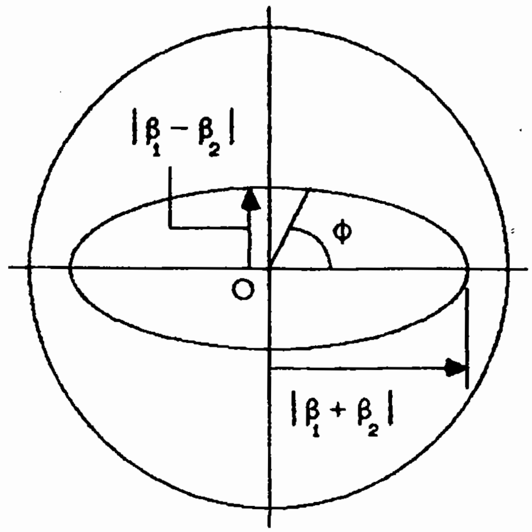
If $\phi = \theta/2$ then,

$$B(e^{j\phi}) = e^{j\phi}B'(e^{j\phi}) \tag{3.14}$$

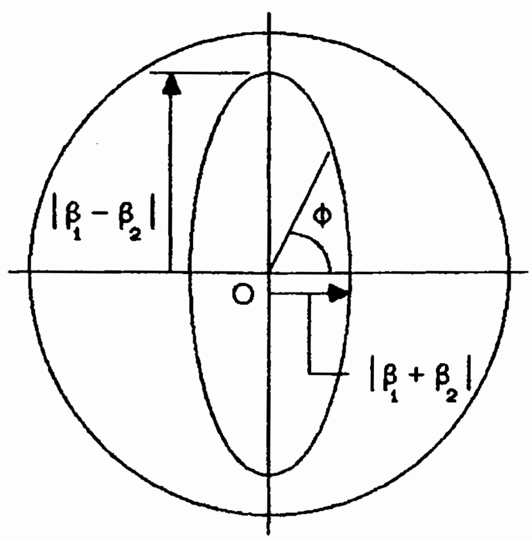
where

$$B'(e^{j\phi}) = \Big[(\beta_1+\beta_2)\cos\phi + j(\beta_1-\beta_2)\sin\phi\Big]. \tag{3.15}$$

The expression $B'(e^{j\phi})$ defines an ellipse with major axis $|\beta_1+\beta_2|$ if β_1 and β_2 have the same sign or $|\beta_1-\beta_2|$ if β_1 and β_2 have different signs. The two cases are illustrated in Fig. 3.3.



(a)   Horizontal Axis is the Major Axis



(b)   Vertical Axis is the Major Axis

Fig. 3.3 Illustration of the Stability Test Ellipse for 2 Taps

Stability is ensured if the ellipse lies entirely within the unit circle. This requires that:

$$\begin{aligned} &|\beta_1 + \beta_2| < 1 \quad \text{when } \beta_1 \text{ and } \beta_2 \text{ have the same signs} \\ &|\beta_1 - \beta_2| < 1 \quad \text{when } \beta_1 \text{ and } \beta_2 \text{ have opposite signs} \end{aligned} \tag{3.16}$$

Combining the two conditions gives the sufficient condition $|\beta_1| + |\beta_2| < 1$ for the stability of a 2 tap filter. This is the same as the sum of the moduli condition found above. The asymptotic tightness of this test for 2 tap filters is shown later.

3 Tap Filters

In a 3 tap filter:

$$\begin{aligned} B(z) &= \beta_1 z^2 + \beta_2 z + \beta_3 \\ &= z(\beta_1 z + \beta_2 + \beta_3 z^{-1}) \end{aligned} \tag{3.17}$$

On the unit circle,

$$B(e^{j\theta}) = e^{j\theta} B'(e^{j\theta}) \tag{3.18}$$

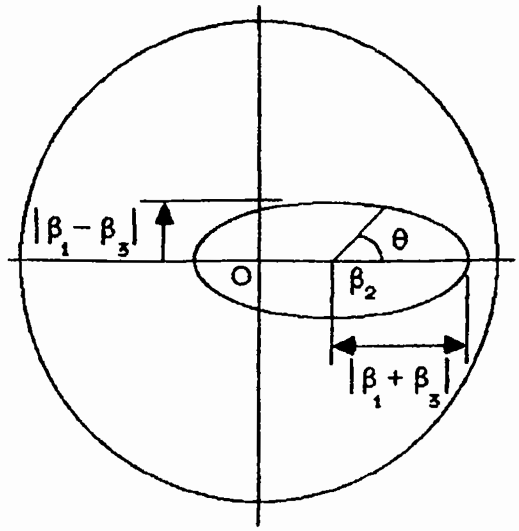
where

$$B'(e^{j\theta}) = \Big[\beta_2 + (\beta_1 + \beta_3)\cos\theta + j(\beta_1 - \beta_3)\sin\theta\Big]. \tag{3.19}$$

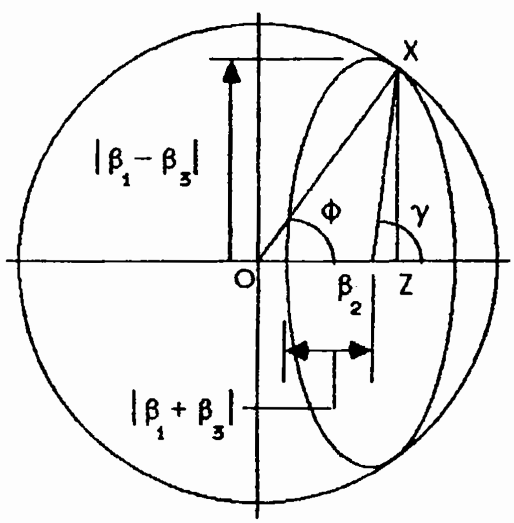
Again, $B'(e^{j\theta})$ defines an ellipse with center β_2 and major axis $|\beta_1 + \beta_3|$ if β_1 and β_3 have the same signs or $|\beta_1 - \beta_3|$ if β_1 and β_3 have opposite signs. The two cases are illustrated in Fig. 3.4. The cases $\beta_2 > 0$ and $\beta_2 < 0$ are symmetrical. For now, the $\beta_2 > 0$ is considered. Later, the analysis is extended to $\beta_2 < 0$.

If β_1 and β_3 have the same signs, the ellipse lies entirely within the unit circle if,

$$|\beta_2| + |\beta_1 + \beta_3| < 1 \tag{3.20}$$



(a) Horizontal Axis is the Major Axis



(b) Vertical Axis is the Major Axis

Fig. 3.4 Illustration of Stability Test Ellipse for 3 Taps or equivalently,

$$|\beta_1| + |\beta_2| + |\beta_3| < 1. \tag{3.21}$$

If β_1 and β_3 have opposite signs, the analysis is more complicated. Even if all points on the major axis $|\beta_1 - \beta_3|$ lie within the unit circle, the ellipse may touch the circle at a point of tangency X as shown in Fig. 3.4(b). Using the substitutions $a = |\beta_1 + \beta_3|$ and $b = |\beta_1 - \beta_3|$, the condition $\beta_2 + a < 1$ ensures that no point on the minor axis lies outside the circle. Similarly, no point on the major axis is outside the circle if $\beta_2^2 + b^2 < 1$. These are necessary conditions for the ellipse to lie within the unit circle and will be assumed to be satisfied for the following discussion.

The ellipse with center β_2 has an equation $\beta_2 + a\cos\theta + jb\sin\theta$. Therefore, the slope of any line tangent to the ellipse is given by:

$$\frac{\frac{d}{d\theta} b\sin\theta}{\frac{d}{d\theta} a\cos\theta} = -\frac{b\cos\theta}{a\sin\theta} \tag{3.22}$$

Let the value of b for which the ellipse is tangent to the unit circle be denoted as $b_{\max}$. If $b < b_{\max}$, the ellipse lies entirely within the circle. If $b > b_{\max}$, a portion of the ellipse lies outside the circle. At $b = b_{\max}$, the slope of line OX is

$$\frac{b_{\max}\sin\gamma}{\beta_2 + a\cos\gamma} = \tan\phi \tag{3.23}$$

where γ is the value of the angle θ which gives tangency. The slope of the tangent line to the ellipse at point X is $-(\beta_2 + a\cos\gamma)/(b_{\max}\sin\gamma)$. Equating appropriate slopes gives:

$$-\frac{\beta_2 + a\cos\gamma}{b_{\max}\sin\gamma} = -\frac{b_{\max}\cos\gamma}{a\sin\gamma} \tag{3.24}$$

Solving Eq. (3.24) for $\cos\gamma$ gives:

$$\cos\gamma = \frac{a\beta_2}{b_{\max}^2 - a^2} \tag{3.25}$$

where $\sin\gamma \neq 0$. Tangency also occurs if $\sin\gamma = 0$ or $\gamma = 0$. If $\gamma = 0$, $\beta_2 + a = 1$. But, the condition $\beta_2 + a < 1$ must be satisfied. For $\beta_2 > 0$, it suffices to consider $0 < |\gamma| < \frac{\pi}{2}$. Otherwise, no point of tangency exists.

In addition to the requirement on the angle γ, a point of tangency exists if the length of OX equals unity. Therefore, regarding $\triangle$OXZ:

$$(\beta_2 + a\cos\gamma)^2 + (b_{\max}\sin\gamma)^2 = 1 \tag{3.26}$$

From Eqs. (3.25) and (3.26) and using the identity $\sin^2\gamma = 1 - \cos^2\gamma$, it can be shown that $b_{\max}$ can be found by solving $f(b_{\max}^2) = 0$ where:

$$f(b^2) = b^4 + b^2(\beta_2^2 - a^2 - 1) + a^2 \tag{3.27}$$

Equation (3.27) describes a quadratic in b^2. The properties of $f(b^2)$ are discussed in Appendix B. The properties are that the two roots of $f(b^2)$ are real and positive and that the desired solution of $f(b^2) = 0$ is the larger of the two roots. In fact, the smaller root is not permissible due to the bound $0 < \cos\gamma < 1$. The ellipse shown in Fig. 3.4(b) is entirely within the unit circle if $b < b_{\max}$. To check that $b < b_{\max}$, one need not solve the equation $f(b_{\max}^2) = 0$. If $b^2 \leq a\beta_2 + a^2$, the ellipse is within the circle. Note that just because $b_{\max}^2$ cannot be less than $a\beta_2 + a^2$ does not mean that b^2 cannot be less than $a\beta_2 + a^2$. If $b^2 > a\beta_2 + a^2$, one must check that $f(b^2) < 0$. If one of the two above conditions hold, there is no point of tangency ($b < b_{\max}$) and the ellipse lies entirely within the circle.

The analysis shown above has used the condition $\beta_2 > 0$. The analysis for the case $\beta_2 < 0$ is very similar and the resulting conditions merely involve replacing β_2 by $|\beta_2|$. Therefore, the implementation of a stability test for a 3 tap pitch synthesis filter is given below.

Stability Test

Let $a = |\beta_1 + \beta_3|$ and $b = |\beta_1 - \beta_3|$.

1. If $a \geq b$, then check:

   (a) $|\beta_1| + |\beta_2| + |\beta_3| < 1$

2. If $a < b$, then check:

   (a) $|\beta_2| + a < 1$

   (b) $\beta_2^2 + b^2 < 1$

   (c) (i) $b^2 \leq a^2 + a|\beta_2|$ or

   (ii) $f(b^2) = b^4 + b^2(\beta_2^2 - a^2 - 1) + a^2 < 0$

Depending on the relationship between a and b, part 1 or 2 must be checked. If the latter is to be checked, conditions (a), (b) and parts (i) or (ii) of condition (c) must be satisfied. When $a \geq b$, the test ensures that all points on the major axis are inside the circle. When $a < b$, conditions (a) and (b) ensure that all points on the major and minor axes are within the circle. Part (i) of condition (c) deals with the situation when the bound $0 < \cos\gamma < 1$ does not hold. Part (ii) of condition (c) is checked when $0 < \cos\gamma < 1$.

This stability test is tighter than the test $|\beta_1|+|\beta_2|+|\beta_3| < 1$ since it can detect the stability of certain filters which the other test cannot. As a simple example, consider $D(z) = z^3 - 0.3z^2 - 0.7z + 0.1$. This polynomial has all its zeros within the unit circle. In this case, $\beta_1 = 0.3$, $\beta_2 = 0.7$, $\beta_3 = -0.1$, $a = 0.2$ and $b = 0.4$. The expression $|\beta_1| + |\beta_2| + |\beta_3| = 1.1 > 1.0$. But using the stability test given above yields:

2. $a < b$

   (a) $|\beta_2| + a = 0.9 < 1$
   (b) $\beta_2^2 + b^2 = 0.65 < 1$
   (c) (i) $b^2 = 0.16 < a^2 + a|\beta_2| = 0.18$

The tighter test indicates that $D(z)$ has all its zeros within the unit circle while simply checking the sum of the moduli of the coefficients does not. On the other hand, if $|\beta_1| + |\beta_2| + |\beta_3| < 1$ (which is the test given in Eq. (3.10)), the conditions in this stability test will always be satisfied.

The stability test for 3 tap filters subsumes the test for 2 tap filters. By setting $\beta_1 = 0$, a 3 tap filter becomes a 2 tap filter. Then, $a = b$ and one must test that $|\beta_2| + |\beta_3| < 1$. This is precisely the test for 2 tap filters.

3.5 Further Analysis of the Sufficient Condition

The stability tests for 2 and 3 tap pitch synthesis filters constitute a set of sufficient conditions. Now, the intention is to examine how tight the sufficient condition is.

The sufficient test defines a region in $(\beta_1, \beta_2, \beta_3)$ space within which stability is guaranteed. The boundaries of the exact necessary and sufficient conditions depend on the order n. However, it will be shown that as $n \to \infty$, the boundaries of the necessary and sufficient region converge to those of the sufficient test. A 3 tap filter is analyzed.

Consider a combination of β_1, β_2 and β_3 or equivalently $|\beta_2|$, a and b which result in a stable filter. Then, the roots of $D(z)$ are inside the unit circle. Let $|\beta_2|$, a and/or b increase to the point at which the ellipse touches the unit circle. Then, $|B(e^{j\theta})| = 1$.

A root of the characteristic polynomial $1 - z^{-n}B(z)$ is on the unit circle if:

$$1 - z^{-n}B(z) = 0 \quad \text{for } z = e^{j\theta}$$

$$\Longleftrightarrow 1 - e^{-j(n-1)\theta}[\beta_1 e^{j\theta} + \beta_2 + \beta_3 e^{-j\theta}] = 0 \tag{3.28}$$

$$\Longleftrightarrow 1 - e^{-j(n-1)\theta}|B'(e^{j\theta})|e^{j\alpha} = 0$$

The expression $\beta_1 e^{j\theta} + \beta_2 + \beta_3 e^{-j\theta}$ has magnitude $|B'(e^{j\theta})|$ and phase angle α. The phase angle α is expressed as:

$$\tan\alpha = \frac{(\beta_1 - \beta_3)\sin\theta}{\beta_2 + (\beta_1 + \beta_3)\cos\theta} \tag{3.29}$$

A root lies on the unit circle if $|B'(e^{j\theta})| = 1$ (the ellipse touches the unit circle in Fig. 3.4) and:

$$e^{-j(n-1)\theta}e^{j\alpha} = 1$$

$$\Longleftrightarrow -(n-1)\theta + \alpha = \pm 2k\pi \quad (k = 0, 1, \cdots) \tag{3.30}$$

$$\Longleftrightarrow \frac{\alpha}{n-1} = \theta \pm \frac{2k\pi}{n-1}$$

As n gets very large, the expression $\theta + \frac{2k\pi}{n-1}$ defines a set of parallel lines that forms a dense grid since these lines are separated by a very small distance.

Assume that the ellipse just touches the unit circle at a point of tangency. At this point, there is a unique α and θ which may or may not satisfy Eq. (3.30). If the size of the ellipse is increased. there are two ($a \geq b$) or four ($a < b$) points at which the ellipse intersects the unit circle ($|B(e^{j\theta})| = 1$). As the ellipse emerges from inside the circle, θ splits into two or four angles corresponding to the now separate points at which the ellipse intersects the unit circle. The angle θ sweeps through a range of values as the ellipse emerges. A root of the characteristic polynomial will cross the unit circle before the ellipse emerges to any extent since there are a large number of solutions to Eq. (3.30) when n is large. In the limit of large n, the ellipse must lie entirely within the unit circle for all the roots of the characteristic polynomial to lie within the unit circle in the z-plane. The sufficient condition $|B(e^{j\theta})| < 1$ actually becomes necessary and sufficient.

In the same fashion, it can be argued that as n gets large, the condition $|B(e^{j\theta})| < 1$ is both necessary and sufficient for ensuring the stability of a 2 tap pitch synthesis filter. However, the test for a 3 tap filter subsumes the 2 tap case. Therefore, the test for a 2 tap filter becomes both necessary and sufficient in the limit of large n.

It can be concluded that the stability test obtained for 2 and 3 tap pitch synthesis filters is very tight considering that it is independent of n. This tight condition was obtained by closely examining $|B(e^{j\theta})|$ rather than being satisfied with the condition $\sum_i |\beta_i| < 1$. In the 2 tap case, the test $|\beta_1| + |\beta_2| < 1$ is the tight condition.

The region in $(\beta_1, \beta_2, \beta_3)$ space defined by $|\beta_1| + |\beta_2| + |\beta_3| = 1$ has eight flat surfaces. Any set of coefficients within this region guarantees stability. The boundaries of the region representing the necessary and sufficient conditions coincide with some of the flat surfaces depending on the signs of β_1, β_2 and β_3. Consider the situation when β_1 and β_3 have the same sign ($a \geq b$) and $|\beta_1| + |\beta_2| + |\beta_3| = 1$. If β_1, β_2 and β_3 are all positive, the polynomial $1 - z^{-n}B(z)$ has a root on the unit circle. Then, $\beta_1 + \beta_2 + \beta_3 < 1$ is a condition for stability and one of the boundaries of the region representing the necessary and sufficient conditions coincides with one of the flat surfaces. Two other instances when this phenomenon occurs are given below.

1. If $\beta_2 > 0$, $\beta_1, \beta_3 < 0$ and n is odd, $-\beta_1 + \beta_2 - \beta_3 < 1$ is required for stability.
2. If $\beta_2 < 0$, $\beta_1, \beta_3 > 0$ and n is even, $\beta_1 - \beta_2 + \beta_3 < 1$ is required for stability.

The other boundaries of the region representing the necessary and sufficient conditions asymptotically converge to the flat surfaces defined by $|\beta_1| + |\beta_2| + |\beta_3| = 1$.

If β_1 and β_3 have opposite signs ($a < b$), no boundary of the necessary and sufficient region is a flat surface. However, it has been shown that these boundaries asymptotically converge to those defined defined by the tight sufficient test rather than the test which simply compares the sum of the moduli of the coefficients with 1.

3.6 Pitch Filters With More Taps

Since the condition $|B(e^{j\theta})| < 1$ is tight, it can be used to obtain a stability test for pitch synthesis filters having more than 3 taps. Again, the test is independent of the order n.

In the case of a 4 tap filter, $D(z) = z^n - \beta_1 z^3 - \beta_2 z^2 - \beta_3 z - \beta_4$. The stability test is formulated in the same way as was done for 2 and 3 tap filters. Therefore, only an outline of the derivation is presented. The polynomial $D(z) = z^n - B(z)$ and $B(z) = \beta_1 z^3 + \beta_2 z^2 + \beta_3 z + \beta_4 = z^{\frac{3}{2}}(\beta_1 z^{\frac{3}{2}} + \beta_2 z^{\frac{1}{2}} + \beta_3 z^{-\frac{1}{2}} + \beta_4 z^{-\frac{3}{2}})$. Then:

$$B(e^{j\theta}) = e^{\frac{j3\theta}{2}}\Big[(\beta_1+\beta_4)\cos\frac{3\theta}{2} + j(\beta_1-\beta_4)\sin\frac{3\theta}{2} + (\beta_2+\beta_3)\cos\frac{\theta}{2} + j(\beta_2-\beta_3)\sin\frac{\theta}{2}\Big] \tag{3.31}$$

If $\phi = \theta/2$ then,

$$B(e^{j\phi}) = e^{j3\phi}B'(e^{j\phi}) \tag{3.32}$$

where

$$B'(e^{j\phi}) = (\beta_1+\beta_4)\cos 3\phi + j(\beta_1-\beta_4)\sin 3\phi + (\beta_2+\beta_3)\cos\phi + j(\beta_2-\beta_3)\sin\phi. \tag{3.33}$$

Two functions $F_1(\phi)$ and $F_2(\phi)$ are defined as:

$$\begin{aligned} F_1(\phi) &= (\beta_1+\beta_4)\cos 3\phi + j(\beta_1-\beta_4)\sin 3\phi \\ F_2(\phi) &= (\beta_2+\beta_3)\cos\phi + j(\beta_2-\beta_3)\sin\phi. \end{aligned} \tag{3.34}$$

Then, $B'(e^{j\phi}) = F_1(\phi) + F_2(\phi)$. Now, $F_2(\phi)$ again defines an ellipse with major axis $|\beta_2+\beta_3|$ or $|\beta_2-\beta_3|$ depending on the signs of β_2 and β_3. The curve defined by $F_1(\phi)$ is more complicated to describe but has some crucial properties. If $|\beta_1+\beta_4| > |\beta_1-\beta_4|$, $F_1(\phi)$ has a maximum magnitude $|\beta_1+\beta_4|$ at $\phi = \frac{\pi k}{3}$ $(k = 0,\cdots,5)$ and a minimum magnitude $|\beta_1-\beta_4|$ at $\phi = \frac{\pi(2k+1)}{6}$ $(k = 0,\cdots,5)$. Similarly if $|\beta_1-\beta_4| > |\beta_1+\beta_4|$, the angles that give the maximum and minimum magnitudes are interchanged. If $|\beta_2+\beta_3|$ is the major axis, $F_2(\phi)$ has a maximum magnitude at $\phi = 0$ and π. Similarly if $|\beta_1-\beta_3|$ is the major axis, the maximum magnitude occurs at $\phi = \frac{\pi}{2}$ and $\frac{3\pi}{2}$.

Using the substitutions $a = |\beta_1+\beta_4|$, $b = |\beta_1-\beta_4|$, $c = |\beta_2+\beta_3|$ and $d = |\beta_2-\beta_3|$, four different cases occur. It is desired that when the curves described by $F_1(\phi)$ and

$F_2(\phi)$ are added, the resulting curve lies entirely within the unit circle. Therefore, $|B(e^{j\phi})| = |B'(e^{j\phi})| = |F_1(\phi)+F_2(\phi)| < 1$. If $a \geq b$ and $c \geq d$, the ellipse defined by $F_2(\phi)$ has a maximum magnitude c at $\phi = 0$. At $\phi = 0$, $F_1(\phi)$ also has a maximum magnitude a. Therefore, $a + c < 1$. The inequality $b + d < 1$ is the condition when $a \leq b$ and $c \leq d$. In either case, it must be checked that the sum of the moduli of the coefficients is less than 1.

When $a > b$ and $c < d$, one must find the angle ϕ for which $|B(e^{j\phi})|$ or more conveniently $|B(e^{j\phi})|^2 = |F_1(\phi) + F_2(\phi)|^2$ is a maximum and test whether this maximum is less than 1. The two expressions $F_1(\phi)$ and $F_2(\phi)$ do not by themselves have a maximum magnitude at the same angle ϕ. Now, $|B(e^{j\phi})|^2 = G(\phi) = (a\cos 3\phi + c\cos\phi)^2 + (b\sin 3\phi + d\sin\phi)^2$. To find the angle ϕ for which $G(\phi)$ is a maximum, $\frac{dG(\phi)}{d\phi}$ is set equal to zero. The resulting equation is:

$$\begin{aligned}\frac{dG(\phi)}{d\phi} &= 3(b^2 - a^2)\cos 3\phi \sin 3\phi + (d^2 - c^2)\cos\phi\sin\phi \\ &\quad + (3bd - ac)\cos 3\phi\sin\phi + (bd - 3ac)\sin 3\phi\cos\phi \\ &= 0\end{aligned} \tag{3.35}$$

Using the relations $\cos 3\phi = (2\cos 2\phi - 1)\cos\phi$ and $\sin 3\phi = (2\cos 2\phi + 1)\sin\phi$, Eq. (3.35) simplifies to:

$$\begin{aligned}\frac{dG(\phi)}{d\phi} &= 3(b^2 - a^2)(2\cos 2\phi - 1)(2\cos 2\phi + 1) + (d^2 - c^2) \\ &\quad + (3bd - ac)(2\cos 2\phi - 1) + (bd - 3ac)(2\cos 2\phi + 1) \\ &= 0\end{aligned} \tag{3.36}$$

Equation (3.36) holds if $\phi \neq 0, \frac{\pi}{2}, \pi$ and $\frac{3\pi}{2}$. If $\phi = 0, \frac{\pi}{2}, \pi$ or $\frac{3\pi}{2}$, it suffices to check that $a+c < 1$ and $b+d < 1$ in the stability test. By letting $2\cos 2\phi = x$, a quadratic equation in x is formed.

$$g(x) = (3b^2 - 3a^2)x^2 + (4bd - 4ac)x$$
$$+ (d^2 - c^2 - 3b^2 + 3a^2 - 2bd - 2ac) \quad (3.37)$$
$$= 0$$

There are two solutions x_1 and x_2 resulting in two angles ϕ_1 and ϕ_2. Since the derivative of $G(\phi)$ is set to zero, one angle corresponds to its minimum value and the other to its maximum value. One must check that $\max(G(\phi_1), G(\phi_2)) < 1$. Since $2\cos 2\phi = x$, then $-2 < x < 2$. Also, $G(\phi)$ is not a constant function of ϕ and hence has a maximum and minimum value. Therefore, a solution of Eq. (3.37) in the range $-2 < x < 2$ will exist. The same analysis holds for $a < b$ and $c > d$. The implementation of a stability test for a 4 tap filter is now given.

Stability Test

1. If $a \geq b$ and $c \geq d$ OR $a \leq b$ and $c \leq d$, then:
   (a) $|\beta_1| + |\beta_2| + |\beta_3| + |\beta_4| < 1$
2. If $a > b$ and $c < d$ OR $a < b$ and $c > d$, then:
   (a) $a + c < 1$
   (b) $b + d < 1$
   (c) Solve Eq. (3.37) for x to obtain ϕ_1 and ϕ_2 where:

$$\phi_1 = \frac{\arccos(\frac{x_1}{2})}{2}$$

$$\phi_2 = \frac{\arccos(\frac{x_2}{2})}{2}$$

   Then check that $\max(G(\phi_1), G(\phi_2)) < 1$.

The test for a 4 tap filter is again tight but uses transcendental functions not only to solve Eq. (3.37) but also to obtain the angles ϕ_1 and ϕ_2 and continue to compute $G(\phi_1)$ and $G(\phi_2)$. If a stability test is formulated for filters having more than 4 taps, transcendental functions will again be necessary. In a real time environment, it may be better to use the condition $\sum_i |\beta_i| < 1$ for filters with more than 3 taps. However, pitch synthesis filters having more than 3 taps are not used in practice. The stability tests given for 1, 2 and 3 tap filters are useful and can be easily implemented in a real time environment.

Chapter 4 Formant-Pitch Sequence

The performance of a pitch predictor is analyzed when it is placed after the formant predictor in a CELP coder. First, a technique for estimating the pitch period is introduced. Then, a stabilization technique for the unstable filters is formulated. This results in a sub-optimum predictor which guarantees the stability of the corresponding pitch synthesis filter. The efficiency of the stabilization technique is evaluated in terms of the loss in prediction gain accompanying this process. Finally, an algorithm that ensures the stability of pitch synthesis filters is presented.

4.1 Covariance Formulation

Given any prediction error filter $A(z)$ of the form

$$A(z) = 1 - \sum_{k=1}^{p} \beta_k z^{-M_k}, \tag{4.1}$$

the covariance formulation chooses the coefficients β_k that minimizes the mean-squared error of the output signal for any particular frame. The mean-squared error ε^2 is expressed as:

$$\begin{aligned}\varepsilon^2 &= \sum_{n=0}^{N-1} e_n^2 \\ &= \sum_{n=0}^{N-1} (d_n - \sum_{k=1}^{p} \beta_k d_{n-M_k})^2\end{aligned} \tag{4.2}$$

The signal d_n is the input signal (in this case the signal derived after formant prediction) and e_n is the output signal. The summation is over N samples where N is the frame length. Minimizing ε^2 is equivalent to setting $\frac{d\varepsilon^2}{d\beta_k} = 0$ for $k = 1$ to p. This leads to a linear system of equations,

$$\sum_{k=1}^{p} \beta_k \phi(M_k, M_l) = \phi(0, M_k) \text{ for } l = 1 \text{ to } p \tag{4.3}$$

where

$$\phi(M_i, M_j) = \sum_{n=0}^{N-1} d_{n-M_i} d_{n-M_j}. \tag{4.4}$$

In matrix form, the above system of equations ($\boldsymbol{\Phi\beta} = \boldsymbol{\alpha}$) is:

$$\begin{bmatrix} \phi(M_1,M_1) & \phi(M_1,M_2) & \cdots & \phi(M_1,M_p) \\ \phi(M_2,M_1) & \phi(M_2,M_2) & \cdots & \phi(M_2,M_p) \\ \vdots & \vdots & & \vdots \\ \phi(M_p,M_1) & \phi(M_p,M_2) & \cdots & \phi(M_p,M_p) \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \vdots \\ \beta_p \end{bmatrix} = \begin{bmatrix} \phi(0,M_1) \\ \phi(0,M_2) \\ \vdots \\ \phi(0,M_p) \end{bmatrix} \tag{4.5}$$

The matrix $\boldsymbol{\Phi}$ is symmetric and positive semi-definite although in practice it is positive definite [18]. The Cholesky decomposition can be used to solve for the predictor coefficients. In the 1 tap case, the equation $\phi(M,M)\beta = \phi(0,M)$ must be solved. When 2 taps are used, $M_1 = M$ and $M_2 = M + 1$. For a 3 tap predictor, $M_1 = M - 1$, $M_2 = M$ and $M_3 = M + 1$.

4.1.1 Estimation of Pitch Period

Solving the above system of equations results in an optimum pitch predictor for a particular value of M. However, the value of M (which is an estimate of the pitch period) must be determined.

Given that a speech waveform is sampled at 8 kHz (used in all experiments), the pitch period is between 20 and 120 samples. This sets a minimum and maximum value for M and covers the range for both male and female speakers. The true pitch period is not necessarily an integral number of samples. However, M must be an integer. A 3 tap predictor has an advantage over a 1 tap predictor in being able to interpolate between samples. Atal [8] describes one method of estimating the pitch period. A correlation array $\tau(M)$ is first calculated where:

$$\tau(M) = \frac{\phi(0, M)}{\sqrt{\phi(0,0)\phi(M, M)}} \tag{4.6}$$

The correlation array is searched for local maxima and parabolic interpolation is used on triplets of correlation values centered at the local maxima. Local peaks (not necessarily integer values) are located at points at which the interpolated functions are a maximum. The pitch period M is the nearest integer value of the largest local peak. This value of M is used in conjunction with the covariance formulation to determine the predictor coefficients. Minimizing the mean-squared error is done in two steps, one involving the choice of M and the other involving the solution of a system of linear equations.

Other pitch determination methods are also available. Lee and Morf [19] have used least-square lattice variables to estimate the pitch period. This method has been later used [20] in conjunction with a least mean-square transversal algorithm to estimate the pitch predictor coefficients recursively. The only disadvantage of this method is that errors in the pitch period estimate cause the predictor coefficients to fluctuate rather than converge to steady-state values [20].

Now, an alternative to Atal's method of choosing M is presented. First, the 1 tap case is considered. Then, $A(z) = 1 - \beta z^{-M}$ and $\beta = \frac{\phi(0,M)}{\phi(M,M)}$. Since this value of β results in an optimum predictor, the resultant mean-square error ε^2 is:

$$\varepsilon^2 = \phi(0,0) - \frac{\phi^2(0,M)}{\phi(M,M)} \tag{4.7}$$

One can choose M to maximize $\frac{\phi^2(0,M)}{\phi(M,M)}$. This is equivalent to maximizing $r^2(M)$.

To verify this claim, an experiment that compared the overall prediction gains of the pitch predictor when choosing M according to Atal's method and the new method was conducted. The overall prediction gain is the ratio (in dB) of the energy of the input signal to the pitch predictor (formed after formant prediction) to the energy of the residual. Throughout the thesis, each speech file is split up into frames of length 80 samples (10 ms). Also, a tenth order formant predictor whose coefficients are derived form the modified covariance method is used. The contents of each speech file are given in Appendix C. The results are shown in Table 4.1. The prediction gain achieved by the new method is the maximum attainable gain because ε^2 reaches its minimum value when $\frac{\phi^2(0,M)}{\phi(M,M)}$ is maximized. In contrast to Atal's method, no parabolic interpolation is used in the new method. Now, a method for choosing M in the case of a 2 and 3 tap filter is presented.

In the case of a 3 tap filter, the equation $\boldsymbol{\Phi}\boldsymbol{\beta} = \boldsymbol{\alpha}$ is solved. Then, $\boldsymbol{\beta} = \boldsymbol{\Phi}^{-1}\boldsymbol{\alpha}$ and the resulting mean-squared error is $\varepsilon^2 = \phi(0,0) - \boldsymbol{\beta}^T\boldsymbol{\alpha}$. The value of M should be chosen so as to maximize $\boldsymbol{\beta}^T\boldsymbol{\alpha}$. For notational convenience, $\boldsymbol{\Phi}$ and $\boldsymbol{\alpha}$ are expressed

Speech File	Atal's Method	New Method
CATM8	4.15	4.16
ADDM8	4.00	4.01
PIPM8	3.46	3.47
TOMF8	6.03	6.06
OAKF8	2.56	2.59
THVF8	4.75	4.78

Table 4.1 Prediction Gains (dB) Depending on Choice of M (1 Tap)

as:

$$\begin{aligned} \boldsymbol{\Phi} &= \begin{bmatrix} \phi(M-1,M-1) & \phi(M-1,M) & \phi(M-1,M+1) \\ \phi(M,M-1) & \phi(M,M) & \phi(M,M+1) \\ \phi(M+1,M-1) & \phi(M+1,M) & \phi(M+1,M+1) \end{bmatrix} \\ &= \begin{bmatrix} x_1 & x_4 & x_5 \\ x_4 & x_2 & x_6 \\ x_5 & x_6 & x_3 \end{bmatrix} \\ \boldsymbol{\alpha} &= \begin{bmatrix} \phi(0,M-1) \\ \phi(0,M) \\ \phi(0,M+1) \end{bmatrix} = \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \end{bmatrix} \end{aligned} \tag{4.8}$$

Since $\boldsymbol{\beta} = \boldsymbol{\Phi}^{-1}\boldsymbol{\alpha}$, $\boldsymbol{\beta}^T\boldsymbol{\alpha} = \frac{f(x_i,\alpha_i)}{\det \boldsymbol{\Phi}}$ where:

$$\begin{aligned} f(x_i,\alpha_i) = {} & \alpha_1^2(x_2x_3 - x_6^2) + \alpha_2^2(x_1x_3 - x_5^2) + \\ & \alpha_3^2(x_1x_2 - x_4^2) + 2\alpha_1\alpha_2(x_6x_5 - x_3x_4) + \\ & 2\alpha_1\alpha_3(x_4x_6 - x_2x_5) + 2\alpha_2\alpha_3(x_4x_5 - x_1x_6) \text{ and} \\ \det \boldsymbol{\Phi} = {} & x_1x_2x_3 - x_1x_6^2 - x_3x_4^2 + 2x_4x_5x_6 - x_2x_5^2 \end{aligned} \tag{4.9}$$

Maximizing such an expression involves a great deal of computation in each frame of speech. Therefore, certain approximations must be made. Since formant prediction has been accomplished, the near-sample-based redundancies have been removed to a large extent. Therefore, the off-diagonal terms (x_4,x_5 and x_6) in the matrix $\boldsymbol{\Phi}$ can be neglected. Now, $f(x_i, \alpha_i) = \alpha_1^2 x_2 x_3 + \alpha_2^2 x_1 x_3 + \alpha_3^2 x_1 x_2$ and det $\Phi = x_1 x_2 x_3$.

$$\begin{aligned}\boldsymbol{\beta}^T \boldsymbol{\alpha} &= \frac{\alpha_1^2}{x_1} + \frac{\alpha_2^2}{x_2} + \frac{\alpha_3^2}{x_3} \\ &= \frac{\phi^2(0, M-1)}{\phi(M-1, M-1)} + \frac{\phi^2(0, M)}{\phi(M, M)} + \frac{\phi^2(0, M+1)}{\phi(M+1, M+1)} \\ &= \phi(0,0)(\tau^2(M-1) + \tau^2(M) + \tau^2(M+1))\end{aligned} \tag{4.10}$$

The value of M that maximizes $\tau^2(M-1) + \tau^2(M) + \tau^2(M+1)$ is chosen.

The derivation for the 2 tap case is similar. In this case, the value of M that maximizes $\tau^2(M)+\tau^2(M+1)$ is chosen. Experimental results are given in Tables 4.2 and 4.3. In these tables, results derived from an exhaustive search are also shown. An exhaustive search of the value of M in each frame is performed by finding the predictor coefficients for each value between 20 and 120, filtering the signal to form the residual and measuring the prediction gain. The value between 20 and 120 that gives the highest prediction gain is then chosen as the value of M. Such a search should never be done in practice but is performed here just to compare it with Atal's method and the new method. The exhaustive search reveals the maximum attainable prediction gain.

Speech File	Atal's Method	New Method	Exhaustive Search
CATM8	5.66	5.68	5.69
ADDM8	5.22	5.23	5.24
PIPM8	4.68	4.75	4.78
TOMF8	6.86	6.88	6.89
OAKF8	3.37	3.44	3.45
THVF8	5.90	5.96	6.00

Table 4.2 Prediction Gains (dB) Depending on Choice of M (2 Tap)

Speech File	Atal's Method	New Method	Exhaustive Search
CATM8	5.94	6.20	6.26
ADDM8	5.47	5.62	5.67
PIPM8	5.00	5.21	5.25
TOMF8	7.11	7.24	7.37
OAKF8	3.50	3.70	3.77
THVF8	6.13	6.27	6.38

Table 4.3 Prediction Gains (dB) Depending on Choice of M (3 Tap)

The new method of choosing M consistently shows a higher prediction gain than Atal's method and serves as a good compromise between Atal's method and the exhaustive search. If no approximations were made in choosing the value of M that maximizes $\boldsymbol{\beta}^T\boldsymbol{\alpha}$, the achieved prediction gain would be the same as the gain derived from the exhaustive search. A feasible alternative to Atal's method has been formulated.

4.2 Stabilization of the Pitch Synthesis Filter

The pitch predictor coefficients can be calculated by using the new method of choosing M and the covariance formulation. This approach does not guarantee the stability of the corresponding pitch synthesis filter. Using the stability test derived in Chapter 3, a stabilization technique is employed.

In each frame of speech, the predictor coefficients are calculated. Then, the stability test is used to determine whether a stable synthesis filter results. If the filter is found to be stable, no modification of the coefficients is made and the optimality is preserved. Otherwise, each coefficient is scaled by a common factor c to force the synthesis filter to be stable. This stabilization technique will be optimal in that c is calculated so as to minimize the loss in prediction gain of the pitch predictor.

4.2.1 Theoretical Development

Solving the system $\boldsymbol{\Phi\beta} = \boldsymbol{\alpha}$ results in an optimum predictor. After scaling by a factor c, the vector of predictor coefficients is $\boldsymbol{\beta}' = c\boldsymbol{\beta} = \boldsymbol{\beta} + \boldsymbol{\delta}$. This results in a sub-optimum predictor [3] where the energy of the output signal ε^2 is:

$$\begin{aligned}\varepsilon^2 &= \phi(0,0) - 2\boldsymbol{\beta}'^T\boldsymbol{\alpha} + \boldsymbol{\beta}'^T\boldsymbol{\Phi}\boldsymbol{\beta}' \\ &= \phi(0,0) - 2(\boldsymbol{\beta}^T + \boldsymbol{\delta}^T)\boldsymbol{\alpha} + (\boldsymbol{\beta}^T + \boldsymbol{\delta}^T)\boldsymbol{\Phi}(\boldsymbol{\beta} + \boldsymbol{\delta}) \\ &= (\phi(0,0) - 2\boldsymbol{\beta}^T\boldsymbol{\alpha} + \boldsymbol{\beta}^T\boldsymbol{\Phi}\boldsymbol{\beta}) + (\boldsymbol{\beta}^T\boldsymbol{\Phi}\boldsymbol{\delta} + \boldsymbol{\delta}^T\boldsymbol{\Phi}\boldsymbol{\beta} - 2\boldsymbol{\delta}^T\boldsymbol{\alpha}) + \boldsymbol{\delta}^T\boldsymbol{\Phi}\boldsymbol{\delta}\end{aligned} \tag{4.11}$$

The minimum mean-square error resulting form an optimum predictor is $\varepsilon^2_{\text{min}} = \phi(0,0) - 2\boldsymbol{\beta}^T\boldsymbol{\alpha} + \boldsymbol{\beta}^T\boldsymbol{\Phi}\boldsymbol{\beta}$. Continuing the above derivation gives:

$$\begin{aligned}\varepsilon^2 &= \varepsilon^2_{\text{min}} + (\boldsymbol{\beta}^T\boldsymbol{\Phi}\boldsymbol{\delta} + \boldsymbol{\delta}^T\boldsymbol{\Phi}\boldsymbol{\beta} - 2\boldsymbol{\delta}^T\boldsymbol{\alpha}) + \boldsymbol{\delta}^T\boldsymbol{\Phi}\boldsymbol{\delta} \\ &= \varepsilon^2_{\text{min}} + ((\boldsymbol{\Phi}^{-1}\boldsymbol{\alpha})^T\boldsymbol{\Phi}\boldsymbol{\delta} + \boldsymbol{\delta}^T\boldsymbol{\alpha} - 2\boldsymbol{\delta}^T\boldsymbol{\alpha}) + \boldsymbol{\delta}^T\boldsymbol{\Phi}\boldsymbol{\delta} \\ &= \varepsilon^2_{\text{min}} + \boldsymbol{\delta}^T\boldsymbol{\Phi}\boldsymbol{\delta}\end{aligned} \tag{4.12}$$

Since $\boldsymbol{\Phi}$ is positive definite, $\boldsymbol{\delta}^T\boldsymbol{\Phi}\boldsymbol{\delta} > 0$. The quantity $\boldsymbol{\delta}^T\boldsymbol{\Phi}\boldsymbol{\delta}$ represents the excess mean-squared error resulting from the sub-optimum predictor. Since $\boldsymbol{\delta} = (c-1)\boldsymbol{\beta}$, $\varepsilon^2 - \varepsilon^2_{\text{min}} = (c-1)^2\boldsymbol{\beta}^T\boldsymbol{\Phi}\boldsymbol{\beta}$. In minimizing $(c-1)^2\boldsymbol{\beta}^T\boldsymbol{\Phi}\boldsymbol{\beta}$, it is observed that as c deviates further away from 1, the loss in prediction gain increases. Therefore, c must be as close to 1 as possible and at the same time stabilize the pitch synthesis filter. It can be shown that $0 < c < 1$ can assure a stable synthesis filter. If $c < 0$, a stable filter can arise but the loss in prediction is more than if $0 < c < 1$. In this stabilization technique, it is assumed that the value of M is unaltered although the algorithm used in choosing it makes use of the fact that $\boldsymbol{\Phi\beta} = \boldsymbol{\alpha}$. Scaling the predictor coefficients shrinks the magnitude of the poles of the system.

1 Tap

In a 1 tap filter, the only requirement on the coefficient β is that $|\beta| < 1$. Also, c must be as close to 1 as possible. If $\beta > 1$, scaling it such that it becomes equal to 1 makes c as close to 1 as possible and ensures marginal stability. Similarly, if $\beta < -1$, its new value should be equal to -1. The stabilization technique used is:

$$\beta' = \begin{cases} +1 & \text{if } \beta > 1 \\ -1 & \text{if } \beta < -1 \end{cases} \tag{4.13}$$

2 Tap

In a 2 tap filter, the scaling factor c must force $|\beta_1| + |\beta_2|$ to be at most equal to 1 in order to achieve marginal stability. Therefore, the value of c which gives marginal stability is:

$$c = \frac{1}{|\beta_1| + |\beta_2|} \tag{4.14}$$

3 Tap

In a 3 tap filter, $a = |\beta_1 + \beta_3|$ and $b = |\beta_1 - \beta_3|$. When $a \geq b$, c must force $|\beta_1| + |\beta_2| + |\beta_3|$ to be at most equal to 1. The formula for c which gives marginal stability is:

$$c = \frac{1}{|\beta_1| + |\beta_2| + |\beta_3|} \tag{4.15}$$

When $a < b$, the stability test as derived in Chapter 3 is:

2. (a) $a + |\beta_2| < 1$
   (b) $b^2 + {\beta_2}^2 < 1$
   (c) (i) $b^2 \leq a^2 + a|\beta_2|$ or
       (ii) $f(b^2) = b^4 + b^2(\beta_2^2 - a^2 - 1) + a^2 < 0$

For unstable filters, $|\beta_1| + |\beta_2| + |\beta_3| \geq 1$ and $s = \frac{1}{|\beta_1|+|\beta_2|+|\beta_3|}$ can always be used as a scaling factor. But, a value of $c \geq s$ that is derived from the tighter test can be found. Since $c \geq s$, the loss in prediction gain when using c is less than or equal to the loss when using s.

Assume that the stability check is violated. In the stabilization procedure, it must first be checked whether or not $b^2 \leq a^2 + a|\beta_2|$. This condition is equivalent to checking $\cos\gamma < 1$. Note that scaling the coefficients does not alter this relationship. If $b^2 \leq a^2 + a|\beta_2|$, marginal stability is achieved when the stability test ellipse for the scaled coefficients is tangent to the circle with $\sin\gamma = 0$. The appropriate scaling factor then forces

$$c(a + |\beta_2|) = 1 \tag{4.16}$$

or

$$c = \frac{1}{a + |\beta_2|}. \tag{4.17}$$

Under these conditions, it is straightforward to show that all points along the major axis of the scaled stability test ellipse lie inside the unit circle (condition 2. (b)) when the minor axis is inside the circle (condition 2. (a)).

Now, if $b^2 > a^2 + a|\beta_2|$, there is a point of tangency for the scaled ellipse (for some c, $0 < c < 1$) for which $0 < |\gamma| < \frac{\pi}{2}$. For that value of c, the quadratic stability function ($f(b^2)$ in condition 2. (c) (ii)) with scaled coefficients is equal to zero.

$$c^4 b^4 + c^2 b^2 (c^2 \beta_2^2 - c^2 a^2 - 1) + c^2 a^2 = 0 \tag{4.18}$$

The solution $c = 0$ is not admissible, giving:

$$c = \sqrt{\frac{b^2 - a^2}{b^4 + b^2 \beta_2^2 - b^2 a^2}} \tag{4.19}$$

With this value of c, it can be shown that all points along the minor and major axes of the scaled ellipse are within the unit circle (conditions 2. (a) and 2. (b) respectively are satisfied). The entire stabilization procedure is summarized in Table 4.4.

Conditions	Scaling Factor c
$a \geq b$	$\frac{1}{\lvert\beta_1\rvert + \lvert\beta_2\rvert + \lvert\beta_3\rvert}$
$a < b$ and $b^2 \leq a^2 + a\lvert\beta_2\rvert$	$\frac{1}{a + \lvert\beta_2\rvert}$
$a < b$ and $b^2 > a^2 + a\lvert\beta_2\rvert$	$\sqrt{\frac{b^2 - a^2}{b^4 + b^2\beta_2^2 - b^2 a^2}}$

Table 4.4 Stabilization Procedure for 3 Tap Filters

4.2.2 Experimental Results for Stabilization

Theoretically, marginal stability has been assured. In practice, complete stability is desired. Therefore, in the case of a 1 tap filter, β' is set to 0.99 or -0.99 to ensure complete stability. The stabilization procedure for 2 and 3 tap filters assures marginal stability in the sense that sometimes the ellipses in Figs. 3.3 and 3.4 are tangent to the circle. Calculating c as required and subtracting a very small quantity ϵ will assure complete stability. Table 4.5 shows the experimental results when the stabilization algorithm is applied to a 1, 2 and 3 tap filters. The prediction gains recorded when no stabilization was performed correspond to those achieved by the covariance formulation. When stabilizing 2 and 3 tap filters, c is computed as required and the minute quantity $\epsilon = 0.001$ subtracted from it.

In 1, 2 and 3 tap filters, the average loss in prediction gain associated with stabilization is 0.03, 0.26 and 0.21 dB respectively. This loss is a very small sacrifice especially if stability is guaranteed. Speech files having male voices show a higher loss in prediction gain as compared to those having female voices.

4.3 Lattice Structured Pitch Predictor

Lattice methods have been employed in linear prediction and are useful in implementing a lattice structured formant predictor [21]. A lattice structured predictor consisting of p stages is an all-zero filter as depicted in Fig. 4.1. The input signal is $d(n)$ and the final error signal is $f_p(n) = e(n)$. Any stage i has a reflection coefficient K_i and forms both the forward residual $f_i(n)$ and the so called backward residual $b_i(n)$. The minimization of the mean-square value of the forward and/or backward residuals is accomplished stage by stage such that after each stage, the mean-square value is reduced. Techniques that minimize only the mean-square value of the forward or backward residual alone are available. However, neither of these methods

Speech File	No Stabilization	After Stabilization
1 Tap		
CATM8	4.16	4.12
ADDM8	4.01	3.95
PIPM8	3.47	3.41
TOMF8	6.06	6.03
OAKF8	2.59	2.57
THVF8	4.78	4.77
2 Tap		
CATM8	5.68	5.34
ADDM8	5.23	4.89
PIPM8	4.75	4.37
TOMF8	6.88	6.59
OAKF8	3.44	3.38
THVF8	5.96	5.78
3 Tap		
CATM8	6.20	5.83
ADDM8	5.62	5.36
PIPM8	5.21	4.88
TOMF8	7.24	7.13
OAKF8	3.70	3.62
THVF8	6.27	6.19

Table 4.5 Prediction Gains (dB) With and Without Stabilization

ensure that the magnitude of the resulting reflection coefficients is bounded by 1, thereby not assuring the stability of the corresponding synthesis filter.

To assure the stability of the synthesis filter at the outset, the Burg method is used. This algorithm, at each stage, minimizes the sum of the mean-square values of the forward and backward residuals and each reflection coefficient K_i is calculated

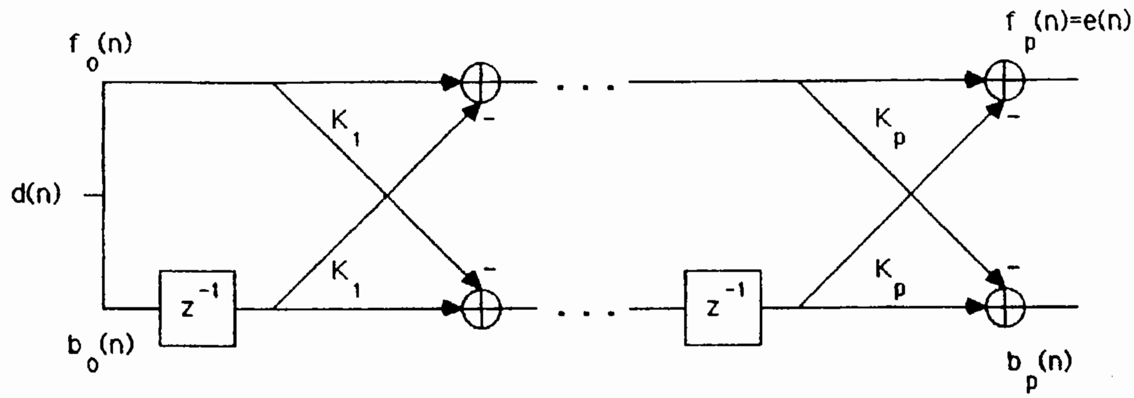


Fig. 4.1 All-Zero Lattice Filter

as,

$$K_i = \frac{2C_{i-1}(n)}{F_{i-1}(n) + B_{i-1}(n-1)} \tag{4.20}$$

where

$$\begin{aligned} C_{i-1}(n) &= \sum_{n=0}^{N-1} f_{i-1}(n) b_{i-1}(n-1) \\ F_{i-1}(n) &= \sum_{n=0}^{N-1} f_{i-1}^2(n) \\ B_{i-1}(n-1) &= \sum_{n=0}^{N-1} b_{i-1}^2(n-1) \end{aligned} \tag{4.21}$$

By using this method, the new mean-square values of the forward and backward residuals are given by:

$$\begin{aligned} F_i(n) &= (1 - K_i^2) F_{i-1}(n) \\ B_i(n) &= (1 - K_i^2) B_{i-1}(n-1) \end{aligned} \tag{4.22}$$

Since this method works well for a formant predictor, the purpose here is to apply this technique in developing a lattice implementation for a pitch predictor. The stability of the corresponding pitch synthesis filter is guaranteed at the outset and hence, no stabilization procedure is necessary. The next section compares this method to the stabilization technique described in the previous section.

When implementing a 1 tap pitch predictor, the value of the pitch period M is first determined, the reflection coefficients K_i for $i = 1$ to $M-1$ are set to zero and K_M is given by:

$$\begin{aligned} K_M &= \frac{2C_{M-1}(n)}{F_{M-1}(n) + B_{M-1}(n-1)} \\ &= \frac{2\phi(0,M)}{\phi(0,0) + \phi(M,M)} \end{aligned} \tag{4.23}$$

The transfer function of the prediction error filter is $A(z) = 1 - K_M z^{-M}$. Since $K_i = 0$ for $i = 1$ to $M-1$, $f_{M-1}(n) = f_{M-2}(n) = \cdots = f_0(n) = d(n)$ and $b_{M-1}(n-1) = b_0(n-M) = d(n-M)$. The resulting mean-square error ε^2 of the forward residual is:

$$\begin{aligned} \varepsilon^2 &= F_M(n) \\ &= (1 - K_M^2)F_{M-1}(n) \\ &= (1 - \frac{4\phi^2(0,M)}{(\phi(0,0) + \phi(M,M))^2})F_{M-1}(n) \end{aligned} \tag{4.24}$$

The value of M that minimizes ε^2 also maximizes K_M^2 or $\frac{\phi^2(0,M)}{(\phi(0,0)+\phi(M,M))^2}$. The choice of M (or equivalently the estimate of the pitch period) when implementing a 1 tap lattice filter is not necessarily the same as that when using a 1 tap transversal filter. In the case of a transversal filter, the value of M that maximizes $\tau^2(M)$ is chosen. The experimental results obtained when using the Burg method are shown in the next section since it is compared with a stabilized filter. An experiment involving the choice of M that maximizes $\tau^2(M)$ in conjunction with the lattice structured predictor was also performed. An average difference in prediction gain of only 0.02 dB was found when the two methods of choosing M were compared. Obviously, the choice of M that maximized K_M^2 consistently revealed a higher gain. This shows that even when using the Burg method, using the value of M that maximizes $\tau^2(M)$ is a feasible choice.

Estimating the pitch period is an issue for 2 and 3 tap filters. In a 2 tap filter, $K_i = 0$ for $i = 1$ to $M-1$ and two reflection coefficients K_M and K_{M+1} are to be determined. The resulting mean-square error is:

$$\begin{aligned}\varepsilon^2 &= (1-K_M^2)(1-K_{M+1}^2)F_{M-1}(n) \\ &= (1-K_M^2-K_{M+1}^2+K_M^2K_{M+1}^2)F_{M-1}(n)\end{aligned} \tag{4.25}$$

Theoretically, choosing the value of M that maximizes $K_M^2+K_{M+1}^2-K_M^2K_{M+1}^2$ results in the minimum value of ε^2. The formula $K_M = \frac{2\phi(0,M)}{\phi(0,0)+\phi(M,M)}$ has already been developed. After the Mth stage, the forward and backward residuals are given by:

$$\begin{aligned}f_M(n) &= f_{M-1}(n) - K_M b_{M-1}(n-1) \\ &= d(n) - K_M d(n-M) \\ b_M(n) &= b_{M-1}(n-1) - K_M f_{M-1}(n) \\ &= d(n-M) - K_M d(n)\end{aligned} \tag{4.26}$$

Then, K_{M+1} is computed as,

$$K_{M+1} = \frac{2C_M(n)}{F_M(n)+B_M(n-1)} \tag{4.27}$$

where

$$\begin{aligned}C_M(n) &= \phi(0,M+1) - K_M\phi(M,M+1) - K_M\phi(0,1) + K_M^2\phi(1,M) \\ F_M(n) &= \phi(0,0) - 2K_M\phi(0,M) + K_M^2\phi(M,M) \text{ and} \\ B_M(n-1) &= \phi(M+1,M+1) - 2K_M\phi(1,M+1) + K_M^2\phi(1,1)\end{aligned} \tag{4.28}$$

Maximizing $K_M^2+K_{M+1}^2-K_M^2K_{M+1}^2$ involves a great deal of computation and is not practically feasible. Even though the near-sample-based redundancies have been removed, not many terms can be neglected. In view of this difficulty, a method for estimating the pitch period is derived empirically.

When determining the value of M for 2 tap filters, it must be ensured that the overall prediction gain must be consistently higher than the 1 tap case. This can be guaranteed by choosing M in the same way as for the 1 tap filter and implementing two lattice stages having coefficients K_M and K_{M+1}. Since maximizing $r^2(M)$ was feasible in the 1 tap case, it should be determined whether or not maximizing $r^2(M) + r^2(M+1)$ is useful for a 2 tap filter. Three different methods for choosing M as shown in Table 4.6 were investigated.

Method	Choice of M	Non-zero Coefficients
1	$\max(r^2(M) + r^2(M+1))$	K_M and K_{M+1}
2	Same as 1 Tap Case	K_M and K_{M+1}
3	Same as 1 Tap Case	K_{M-1} and K_M

Table 4.6 Methods of Choosing M in 2 Tap Lattice Predictors

Although only method (2) theoretically guarantees a higher overall prediction gain than a 1 tap filter, all three methods consistently showed a greater gain. Furthermore, method (1) consistently showed the highest prediction gain which was on the average 0.60 and 0.36 dB more than methods (2) and (3) respectively. A final test compares method (1) with an exhaustive search. An exhaustive search, for each value of M between 20 and 120, finds the reflection coefficients, filters the input signal, measures the prediction gain and then finally selects the value of M that gives the highest gain. The comparison is done to determine how effective the proposed method is since an exhaustive search reveals the maximum possible gain that can be realized. The average difference between method (1) and the exhaustive search is just 0.12 dB making it the method of choice.

Speech File	Method (1)	Exhaustive Search
CATM8	5.05	5.14
ADDM8	4.96	5.04
PIPM8	4.28	4.43
TOMF8	6.95	7.09
OAKF8	3.43	3.49
THVF8	5.80	6.02

Table 4.7 Prediction Gains (dB) for 2 Tap Lattice Predictor

Estimating the pitch period for a 3 tap predictor must again be done empirically. All of the methods considered are given in Table 4.8. Methods (1), (2) and (3) will show higher gains than their 2 tap counterparts because of the stage by stage minimization procedure. Since it has been decided that method (1) should be used for 2 tap filters due to its good performance, it is anticipated that this method will perform well for 3 tap predictors.

Method	Choice of M	Non-zero Coefficients
1	$\max(\tau^2(M) + \tau^2(M+1))$	K_M, K_{M+1} and K_{M+2}
2	Same as 1 Tap Case	K_M, K_{M+1} and K_{M+2}
3	Same as 1 Tap Case	K_{M-1}, K_M and K_{M+1}
4	$\max(\tau^2(M-1) + \tau^2(M) + \tau^2(M+1))$	K_{M-1}, K_M and K_{M+1}

Table 4.8 Methods of Choosing M in 3 Tap Lattice Predictors

Experimental evidence indicates that method (4) is definitely not useful and method (1) again consistently gives the best performance. In fact, the performance using method (1) is superior to both methods (2) and (3) (on the average) by 0.58 and 0.12 dB respectively. Finally, results that compare method (1) to the exhaustive search are shown in Table 4.9. An average difference of just 0.28 dB renders method (1) as a good technique. To summarize, one must choose M that maximizes $\tau^2(M) + \tau^2(M+1)$ and let the Mth stage be the first with a non-zero reflection coefficient for both 2 and 3 tap predictors.

Speech File	Method (1)	Exhaustive Search
CATM8	5.30	5.66
ADDM8	5.24	5.37
PIPM8	4.55	4.79
TOMF8	7.11	7.52
OAKF8	3.86	4.03
THVF8	6.25	6.59

Table 4.9 Prediction Gains (dB) for 3 Tap Lattice Predictor

The 2 and 3 tap prediction error filters derived from the Burg method do not have exactly the same transfer functions as those derived from the covariance formulation. The transfer functions are given by:

$$\text{2 tap} \quad A(z) = 1 + K_M K_{M+1} z^{-1} - K_M z^{-M} - K_{M+1} z^{-(M+1)}$$

$$\begin{aligned}\text{3 tap} \quad A(z) = 1 &+ (K_M K_{M+1} + K_{M+1} K_{M+2}) z^{-1} + K_M K_{M+1} z^{-2} - K_M z^{-M} \\ &- (K_{M+1} + K_M K_{M+1} K_{M+2}) z^{-(M+1)} - K_{M+2} z^{-(M+2)}\end{aligned} \tag{4.29}$$

The difference is due to the presence of z^{-1} and z^{-2} terms. In spite of this difference, an extremely good performance is achieved and as shown in the next section, the amplitude of the pitch pulses are greatly reduced. The main purpose of a pitch predictor is to shrink the pitch pulses.

A computationally efficient procedure called the covariance-lattice method [21], calculates the reflection coefficients using Eq. (4.20) but expresses it in terms of the covariance of the input signal. This efficient algorithm is applied in formant prediction and can also be used in pitch prediction.

4.4 Comparison of the Three Methods

Two methods that guarantee the stability of a pitch synthesis filter are provided. One approach uses the stability test formulated in Chapter 3 to modify the predictor coefficients derived form the covariance formulation. The other approach implements a lattice structured predictor such that at each stage, the magnitude of the reflection coefficient is bounded by 1. This section compares these two methods with the covariance formulation in an attempt to decide whether or not one of the approaches is clearly better than the other. The prediction gains are tabulated in Table 4.10.

When utilizing a 1 tap predictor, use of the stabilization technique consistently shows a higher gain than the lattice method. However, in 2 and 3 tap filters, the same statement does not hold. In 2 tap filters, the lattice method shows a higher average gain of 0.02 dB. In 3 tap filters, the situation is reversed since the stabilization technique reveals a greater average gain of 0.12 dB. The differences are definitely very small and hence, both methods are feasible. In some speech files, the lattice method outperforms the covariance formulation mainly because the transfer functions of the two predictors are different.

Speech File	Covariance	After Stabilization	Lattice Predictor
1 Tap			
CATM8	4.16	4.12	3.84
ADDM8	4.01	3.95	3.81
PIPM8	3.47	3.41	3.20
TOMF8	6.06	6.04	5.89
OAKF8	2.59	2.57	2.50
THVF8	4.78	4.77	4.67
2 Tap			
CATM8	5.68	5.34	5.05
ADDM8	5.23	4.89	4.96
PIPM8	4.75	4.37	4.28
TOMF8	6.88	6.59	6.95
OAKF8	3.44	3.38	3.43
THVF8	5.96	5.78	5.80
3 Tap			
CATM8	6.20	5.83	5.30
ADDM8	5.62	5.36	5.24
PIPM8	5.21	4.88	4.55
TOMF8	7.24	7.13	7.11
OAKF8	3.70	3.62	3.86
THVF8	6.27	6.19	6.25

Table 4.10 Comparison of the Prediction Gains (dB) for Each Method

Figure 4.2 shows six plots of which the first two are the speech waveform TOMF8 and its energy in each frame. Plots (3), (4) and (5) show the prediction gains in each frame when the covariance formulation, stabilization technique and lattice method are used to implement a 3 tap predictor. In plot (6), frames in which unstable pitch synthesis filters are present are indicated when the function value equals 10. As expected, the prediction gains are much higher in the voiced segments. Plots showing the prediction gains all follow approximately the same contour.

Figure 4.3 shows different waveforms during frames 97 to 104. Waveforms (1) and (2) depict the original signal TOMF8 and the formant predicted residual. Waveforms (3), (4) and (5) are the residuals formed after pitch prediction when the covariance formulation, stabilization technique and lattice method are employed. This section of the speech waveform was chosen since frames 97 to 101 had unstable pitch synthesis filters when the covariance approach was used. All the pitch predicted residuals have a much lower energy than the formant predicted residual and no pitch pulses of significant amplitudes remain.

4.5 Effect of Instability on Decoded Speech

This section examines how unstable pitch synthesis filters affect decoded speech. A CELP coder was simulated using 3 tap pitch filters. After formant and pitch prediction, each residual was compared to a dictionary of $2^{10} = 1024$ Gaussian waveforms. The comparisons were made over blocks of data of length 40 samples. The spectral weighting factor used was 0.8 (value of α as described in section 2.1.1). Figure 4.4 depicts four waveforms of which the first is the original signal CATM8 and the next three are decoded signals when using the covariance formulation, stabilization technique and lattice method. Plot (5) indicates frames having unstable filters.

All three decoded waveforms are distorted due to quantization noise filtered by $H_P(z)$ and $H_F(z)$. If $H_P(z)$ is unstable, the energy of the noise is enhanced and

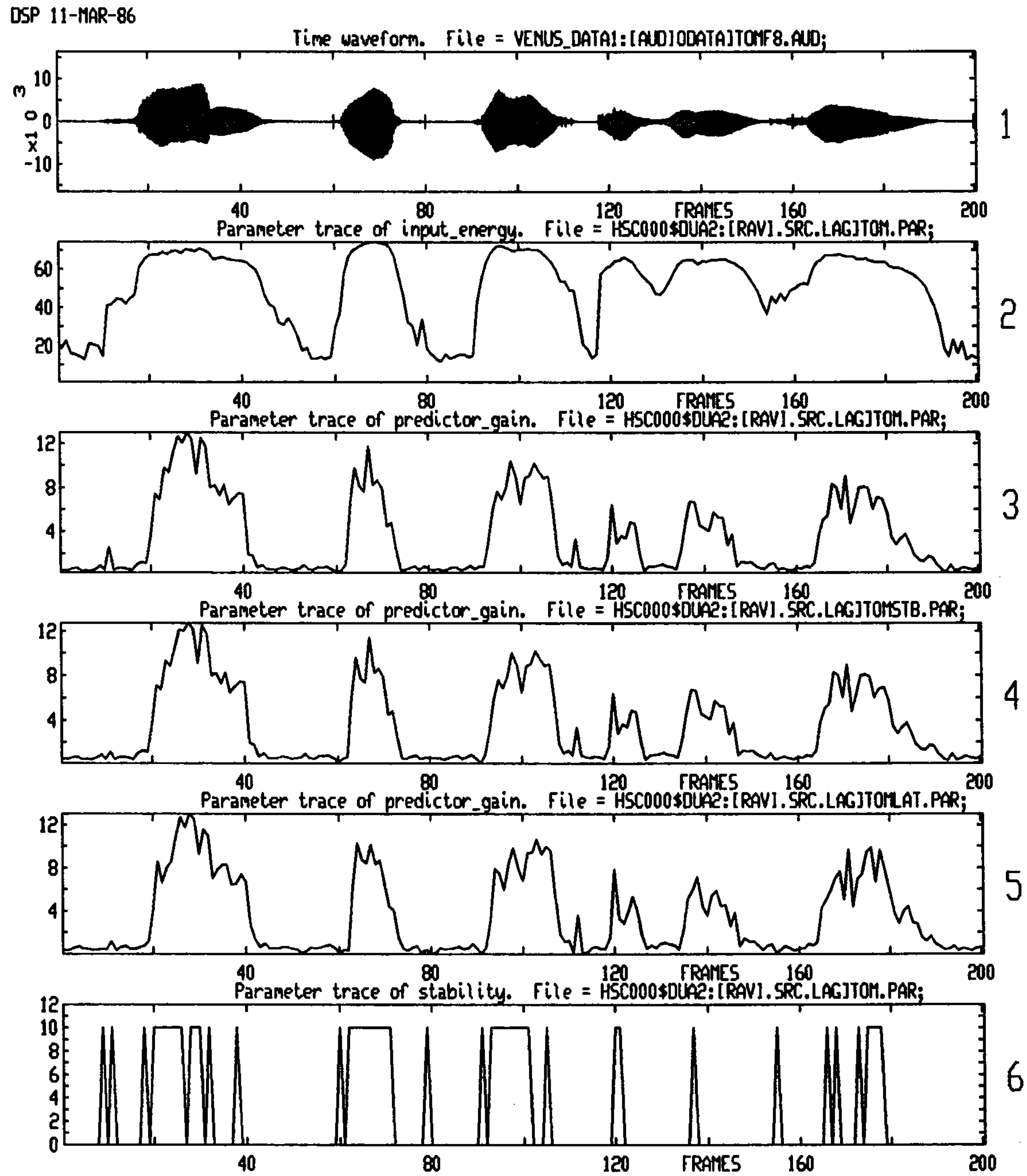


Fig. 4.2 Prediction Gains in Each Frame (3 Tap) (1) Speech Data, (2) Input Energy in Each Frame, (3) Prediction Gain for Covariance Formulation, (4) Prediction Gain for Stabilization Technique, (5) Prediction Gain for Lattice Method and (6) Frames Having Unstable Pitch Synthesis Filters

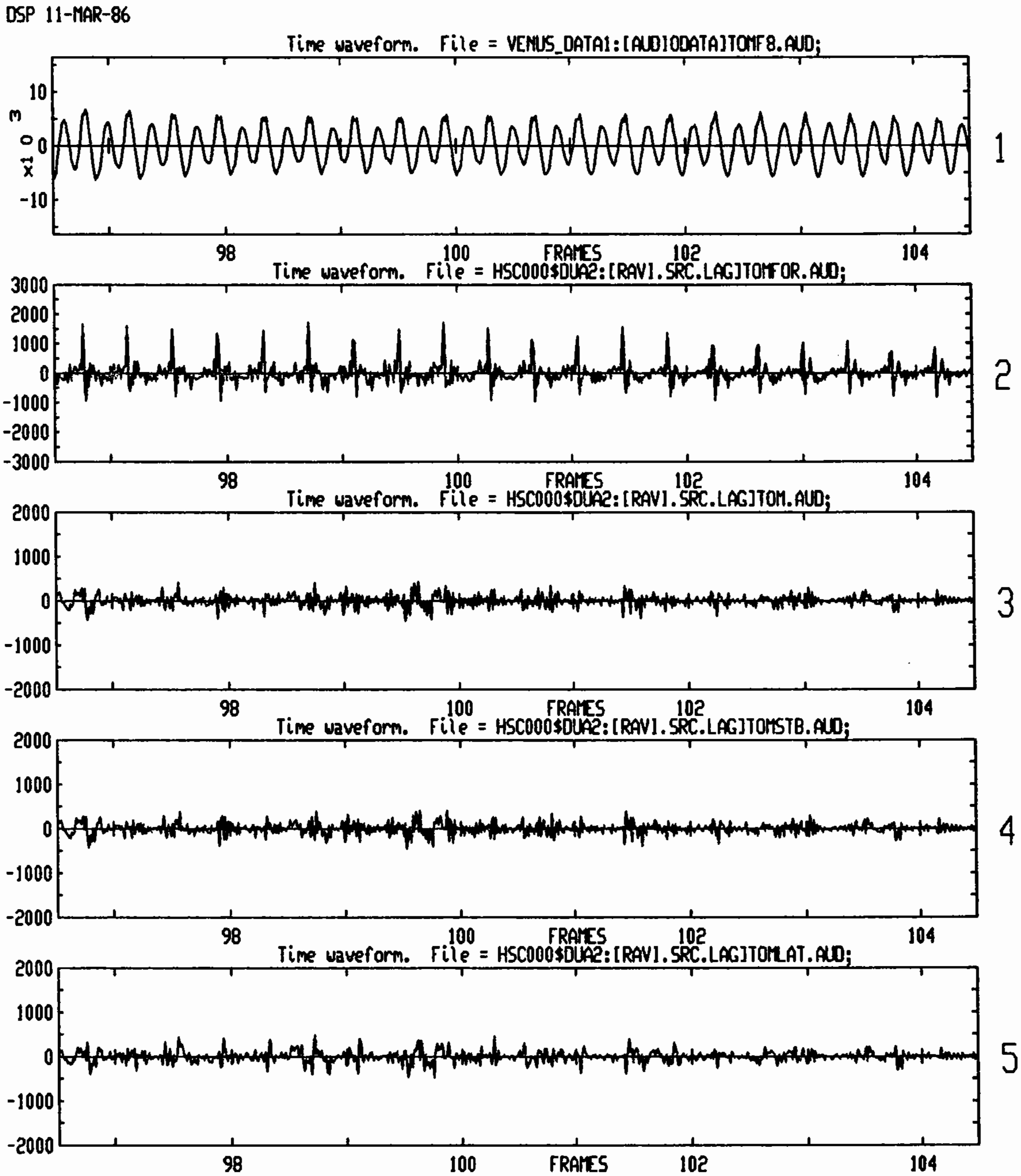


Fig. 4.3 Formant and Pitch Predicted Residuals (3 Tap) (1) Speech Data, (2) Formant Predicted Residual, (3) Pitch Predicted Residual for Covariance Formulation, (4) Pitch Predicted Residual for Stabilization Technique and (5) Pitch Predicted Residual for Lattice Method

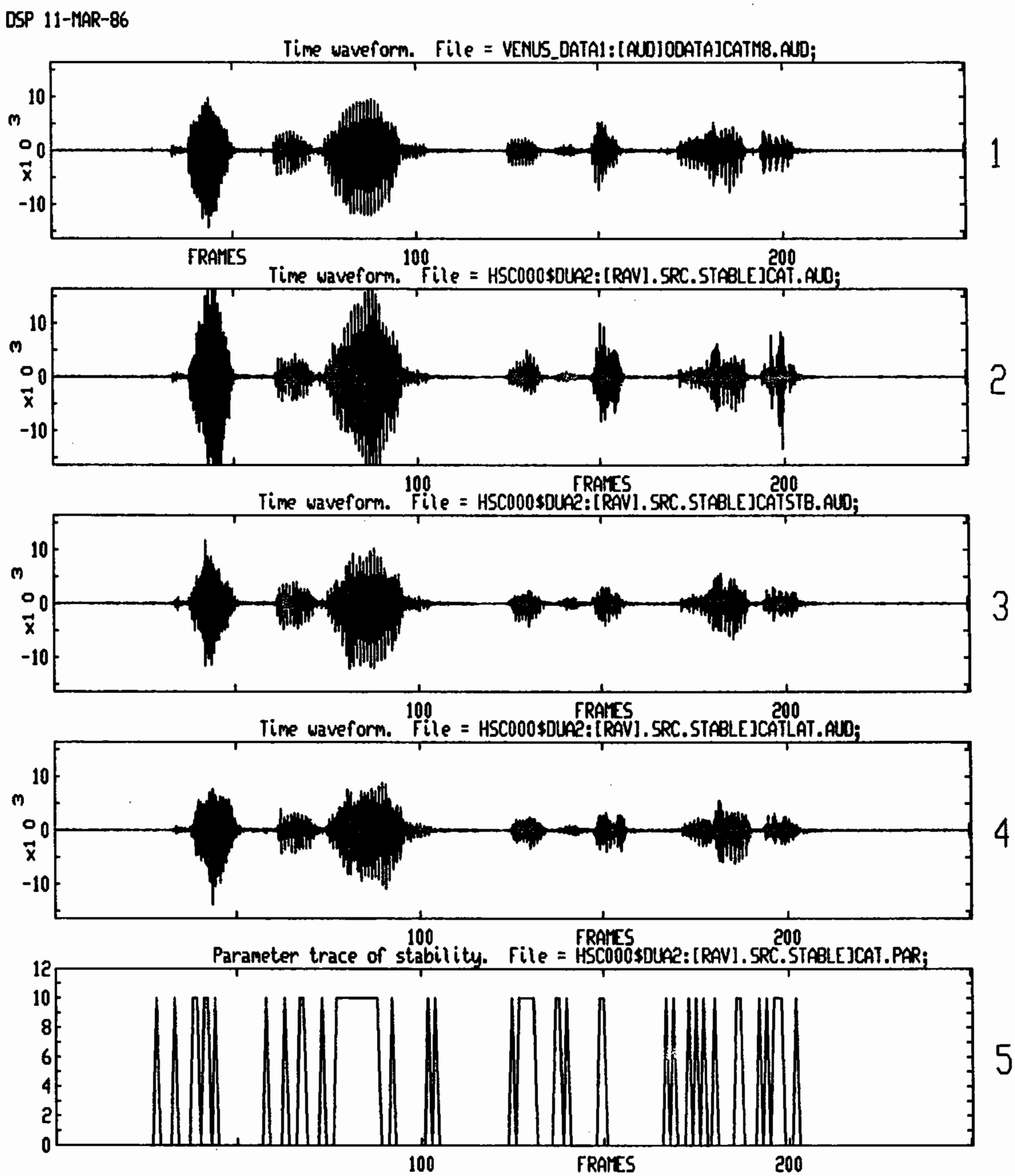


Fig. 4.4 Original and Decoded Speech Signals (1) Speech Data, (2) Decoded Signal for Covariance Formulation, (3) Decoded Signal for Stabilization Technique, (4) Decoded Signal for Lattice Method and (5) Frames Having Unstable Pitch Synthesis Filters

causes further degradation as observed in waveform (2). Although a perceptual test showed that all of the decoded signals were intelligible, waveform (2) suffers from pops or clicks that can be annoying. Also, background noise is more dominant in portions of waveform (2) than in waveforms (3) and (4).

Degradations in the output speech is perceptible if a sequence of consecutive frames of high input energy have unstable filters or if the numerical value of $\sum_i |\beta_i|$ is high. Frames 77 to 88 consist of high-energy voiced speech and have unstable pitch synthesis filters. The quantization noise continues to build up causing the energy of the output signal to keep rising. This background noise is perceptible. Although the average value of $\sum_i |\beta_i|$ is just 1.3, distortion is present since this segment has a high input energy. Figure 4.5 shows the original and output signals during frames 75 to 90. The rising energy in waveform (2) is clearly visible.

Certain isolated frames of high input energy having unstable filters also demonstrate a rising output energy (frames 38 to 50). Frames 38 and 39 have filters whose value of $\sum_i |\beta_i|$ equals 2.36 and 2.46 respectively. These big values are responsible for some distortion. This distortion continues since frames 41, 42 and 44 have unstable filters where $\sum_i |\beta_i|$ equals 1.45, 1.57 and 1.54 respectively.

The degradation is not serious when unstable filters with low values of $\sum_i |\beta_i|$ are present in frames of relatively low energy. This situation occurs during frames 127, 128 and 129 in which $\sum_i |\beta_i|$ equals 1.11, 1.53 and 1.35. If an unstable filter with a very high value of $\sum_i |\beta_i|$ occurs even in a frame of low input energy, an impulse-type distortion that is heard as a pop or click is present. Frames 149 and 150 with values of $\sum_i |\beta_i|$ equal to 4.43 and 2.70 clearly depict this. Another example

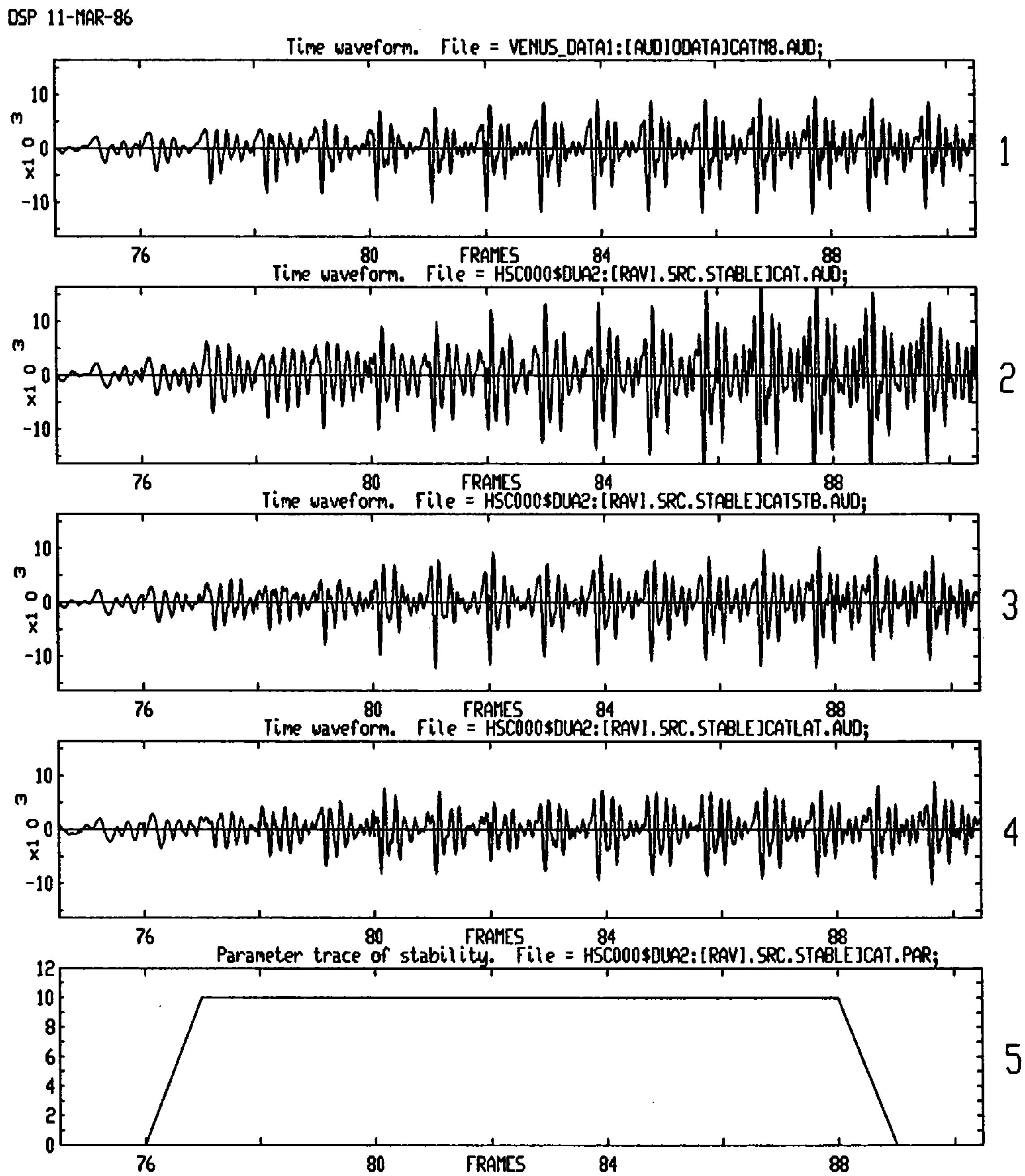


Fig. 4.5 View of Frames 75 to 90 (1) Speech Data, (2) Decoded Signal for Covariance Formulation, (3) Decoded Signal for Stabilization Technique, (4) Decoded Signal for Lattice Method and (5) Frames Having Unstable Pitch Synthesis Filters

of this phenomenon is during frames 196 to 198. Here, $\sum_i |\beta_i|$ equals 2.37, 4.02 and 2.23. Figure 4.6 shows the input and decoded signals during frames 195 to 200. The distortion which is visible in waveform (2) has spread to frame 199 and a clear pop sound is heard. No glaring distortion is present in waveforms (3) and (4).

Since stable pitch synthesis filters do not cause as much perceptible distortion as unstable filters, it is highly recommended that they be used. The stabilization technique and lattice method guarantee stability and generate decoded signals (3) and (4) as shown in Fig. 4.4. Perceptually, waveforms (3) and (4) are highly intelligible and do not sound much different. Neither has the undesirable pops, clicks or enhanced background noise. Waveform (4) appears to have a lower overall energy than waveform (3). Another difference is that the pitch pulses in waveform (4) sometimes have lower amplitudes than the pulses in waveform (3) as can be seen in Fig. 4.5 (this phenomenon does not occur over the entire speech signal). However, these differences cause no perceptual effect. Therefore, any of the methods that achieves stability can be used.

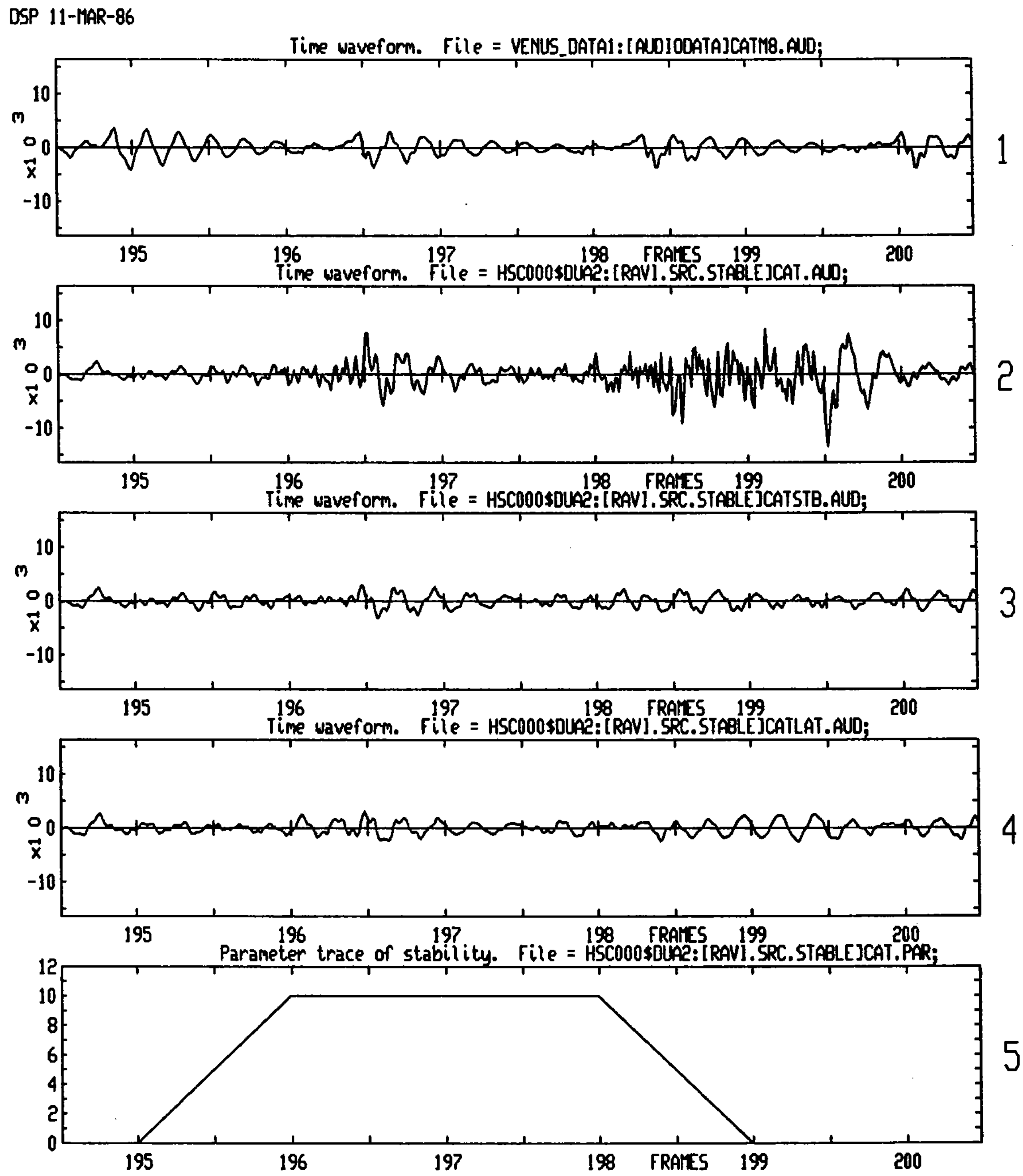


Fig. 4.6 View of Frames 195 to 200 (1) Speech Data, (2) Decoded Signal for Covariance Formulation, (3) Decoded Signal for Stabilization Technique, (4) Decoded Signal for Lattice Method and (5) Frames Having Unstable Pitch Synthesis Filters

Chapter 5 Pitch-Formant Sequence

In contrast to Chapter 4, a pitch predictor which precedes a formant predictor in a CELP coder is now considered. Stability and performance analyses are again performed with the objective of comparing this arrangement (P-F sequence) with the previous configuration (F-P sequence). This investigation enables one to determine which sequence is to be preferred. In particular, the comparisons focus on 3 tap pitch predictors.

5.1 Stability and Performance Issues

Estimating the pitch period M when the pitch predictor is applied directly to the speech signal cannot be done in the same way as before. The only exception is the 1 tap case where the pitch period is the value of M that maximizes $\tau^2(M)$. For a 3 tap predictor, the approximations that were previously made no longer hold and an exact choice of the best M can be computationally heavy. Since the value of M chosen in the 1 tap case is a reliable estimate, it can be used in conjunction with the covariance formulation to implement a 3 tap filter.

Experimental evidence indicates that both the formant and pitch predictors achieve a high gain (actual results given in next section). An exhaustive search of the value of M that maximizes the gain of the pitch predictor (or simply pitch gain) only shows an average improvement of 0.45 dB. In some speech files, the corresponding gain of the formant predictor (formant gain) decreases. Therefore, the total gain (sum of pitch and formant gains) does not always improve and on the average, is nearly the same.

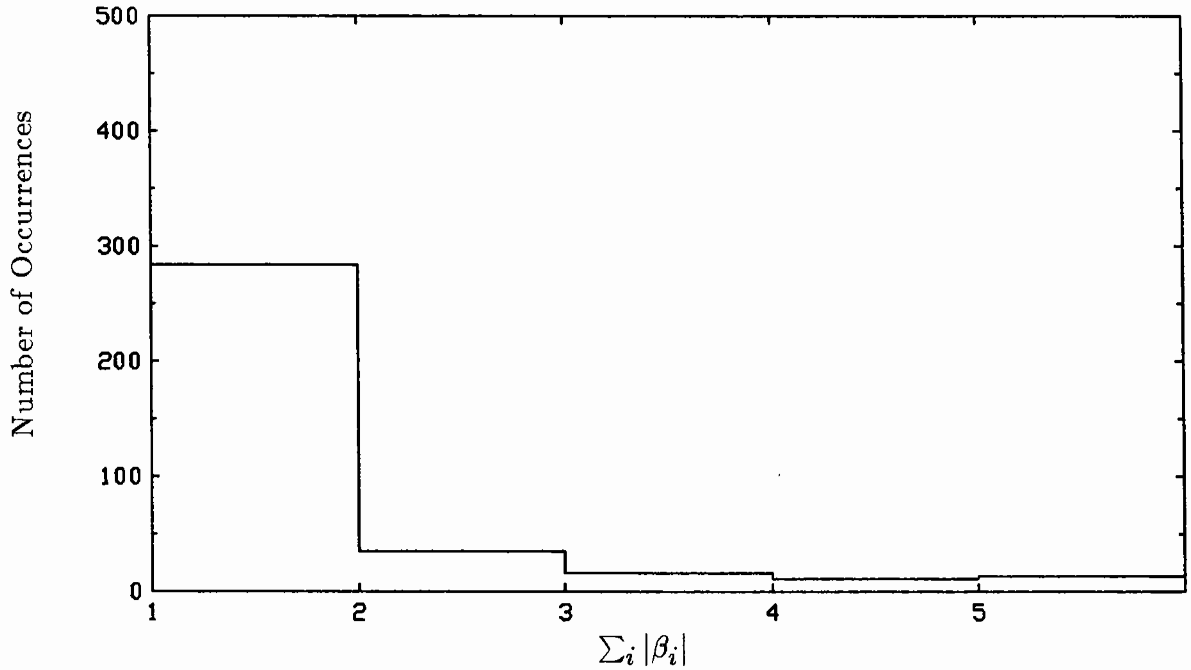
An exhaustive search of the value of M that maximizes the total gain is also performed. When compared to the covariance formulation, the pitch gain is consistently reduced and the formant gain increased such that an average improvement of only 0.91 dB results. The two experiments involving an exhaustive search suggest that by choosing M to only maximize the pitch gain may sometimes diminish the total gain. Obviously, selecting M in order to maximize the total gain is a complex problem and does not achieve a significant improvement. Therefore, the proposed method of determining the pitch period is feasible.

A pitch filter used in a P-F sequence is more susceptible to instability than when used in a F-P arrangement. As mentioned in Chapter 3, frames in which the input energy show a sudden increase usually give rise to an unstable filter. The original speech waveform (input to the pitch predictor in a P-F sequence) has a higher input energy that shows greater variations from frame to frame than the residual after formant prediction (input signal in F-P sequence). When using a F-P sequence, an average of 23% of the frames had unstable pitch synthesis filters. In a P-F sequence, the number increased by more than a factor of 2, namely 49%. This is almost half of the total number of frames processed.

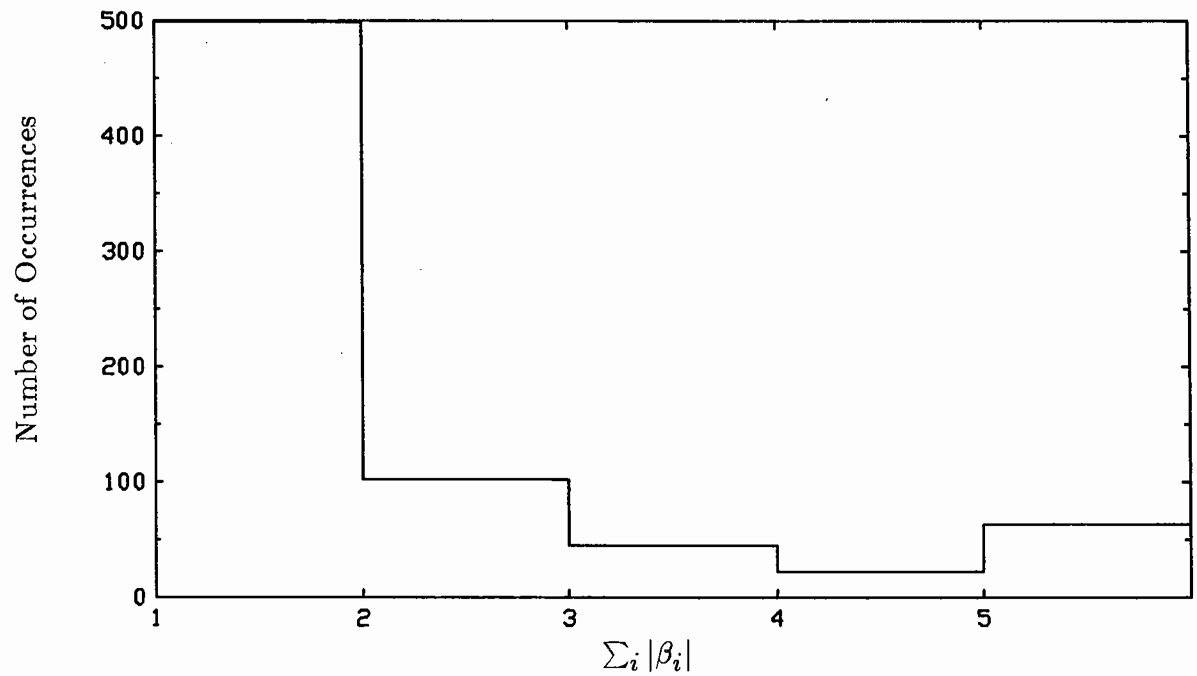
When discussing stability, the parameter $\sum_i |\beta_i|$ is important since as its value increases, the loss in prediction gain associated with stabilization also increases and if the filter is not stabilized, more distortion is introduced in the decoded speech. In frames having unstable filters, the number of occurrences of the value of $\sum_i |\beta_i|$ between 1 and 5 (in steps of 1) and greater than 5 are shown in Fig. 5.1 by a histogram. The data was gathered by using the six speech files employed throughout the thesis.

In both the F-P and P-F configurations, the value of $\sum_i |\beta_i|$ is mostly between 1 and 2 when an unstable filter arises. In a P-F sequence, the number of frames in which $\sum_i |\beta_i| > 5$ is approximately five times more than in a F-P sequence. Also, the number of occurrences of the value of $\sum_i |\beta_i|$ in any bin of the histogram is consistently higher in a P-F arrangement. A clear disadvantage of the P-F configuration is that the pitch filter becomes unstable more often. As revealed in the next section, the loss in prediction gain due to stabilization also increases.

A lattice structured predictor derived from the Burg method can be utilized. The pitch period is determined by a technique that is different from the one used in a F-P sequence. Again, a reliable algorithm must be determined experimentally. In a 1 tap lattice predictor, selecting the value of M that maximizes $\mu^2(M) = \frac{\phi^2(0,M)}{(\phi(0,0)+\phi(M,M))^2}$ is theoretically, the best approach. Choosing M in this fashion and adding two lattice stages guarantees a higher pitch gain. Also, choosing the value of M that maximizes $r^2(M)$ is worth investigating. Four different approaches as given in Table 5.1 were examined.



(a) F-P Sequence



(b) P-F Sequence

Fig. 5.1 Histograms of the Value of $\sum_i |\beta_i|$

Only methods (2) and (4) are feasible to implement. Method (4) shows only a slightly higher total gain than method (2) and is preferred since it is consistent with

Method	Choice of M	Non-Zero Coefficients
1	$\max(\tau^2(M))$	K_{M-1}, K_M and K_{M+1}
2	$\max(\tau^2(M))$	K_M, K_{M+1} and K_{M+2}
3	$\max(\mu^2(M))$	K_{M-1}, K_M and K_{M+1}
4	$\max(\mu^2(M))$	K_M, K_{M+1} and K_{M+2}

Table 5.1 Techniques to Estimate Pitch Period (3 Tap Lattice)

the optimal solution for a 1 tap lattice predictor. When performing an exhaustive search of the value of M that maximizes the pitch gain, the total gain consistently improves but only by an average of 0.45 dB. An exhaustive search of the value of M that maximizes the total gain shows an average increase of only 0.95 dB. Therefore, method (4) performs well.

5.2 Experimental Results

Table 5.1 shows all the results (prediction gains) obtained for a P-F sequence and the total gain achieved by a F-P sequence. The F-P arrangement consistently reveals a higher average gain of 1.8, 2.0 and 1.5 dB when using the covariance, stabilization and lattice algorithms respectively. Even from a performance point of view, the F-P configuration is more advantageous than its P-F counterpart. The prediction gains may be interpreted as being rather high. This is because in a CELP system, the input signal is predicted from its actual previous samples rather than the reconstructed samples which are used in an APC coder.

In a P-F sequence, stabilization of the pitch filter results in average loss of 2.6 dB (much more than in the F-P case). But, a high percentage of this loss

Speech File		Covariance	After Stabilization	Lattice Predictor
CATM8	Pitch	7.86	6.37	9.63
	Formant	8.87	9.89	6.86
	Total (P-F)	16.7	16.3	16.5
	Total (F-P)	18.4	18.0	17.5
ADDM8	Pitch	7.80	6.32	8.87
	Formant	7.51	8.92	6.14
	Total (P-F)	15.3	15.2	15.0
	Total (F-P)	16.9	16.7	16.5
PIPM8	Pitch	6.91	5.17	9.19
	Formant	8.56	10.12	6.09
	Total (P-F)	15.5	15.3	15.3
	Total (F-P)	18.4	18.1	17.8
TOMF8	Pitch	18.29	15.22	18.52
	Formant	7.08	9.63	5.88
	Total (P-F)	25.4	24.9	24.4
	Total (F-P)	25.8	25.7	25.6
OAKF8	Pitch	12.56	9.57	15.77
	Formant	8.52	11.43	6.55
	Total (P-F)	21.1	21.0	22.3
	Total (F-P)	23.7	23.6	23.9
THVF8	Pitch	14.80	10.76	16.91
	Formant	8.31	11.39	6.64
	Total (P-F)	23.1	22.2	23.6
	Total (F-P)	24.8	24.7	24.8

Table 5.2 Complete Experimental Results

is retrieved in the formant gain such that the total gain is almost equal to that obtained originally. The lattice predictor reveals a higher pitch gain than the other methods because of the z^{-1} and z^{-2} terms in its transfer function. Due to some removal of the near-sampled based redundancies, the corresponding formant gain is lower than that obtained by the other methods.

An examination of the final residual signals provides additional insight. Two examples consisting of seven plots are shown. The first plot shows the original speech waveform. Plots (2), (3) and (4) depict the residual waveforms when the covariance, stabilization and lattice methods are used in a F-P sequence while (5), (6) and (7) illustrate their counterparts in a P-F arrangement. Figure 5.2 consists of the same segment as Fig. 4.4 but magnifies the residuals by a factor of two. Pulses of much higher amplitudes are present when a P-F arrangement is employed suggesting that a F-P sequence is more effective in removing pitch pulses. This phenomenon consistently occurs during many portions of different residuals formed by filtering various speech waveforms.

Another observation that explains the loss in pitch gain associated with stabilization is illustrated by another example (Fig. 5.3). This figure shows frames 103-123 of the signal OAKF8. In this segment, only frames 115 and 116 have stable pitch filters in a P-F arrangement. By comparing plots (5) and (6), it is observed that stabilization sometimes magnifies an already existing pulse (frame 120) and introduces pulses of significant amplitudes (frames 104 and 108). This is one main reason behind the substantial loss in pitch gain due to stabilization. Obviously, the corresponding increase in formant gain has no effect in removing pitch pulses.

Plot (7) in Fig. 5.3 depicts a residual in which pitch pulses of high amplitudes are absent. This characteristic obtained from the lattice method occurs in some portions of different residuals suggesting that if a P-F sequence is employed, the Burg algorithm should be chosen in preference to the other two. The speech segment shown in Fig. 5.3 was chosen in order to simultaneously explain the effects

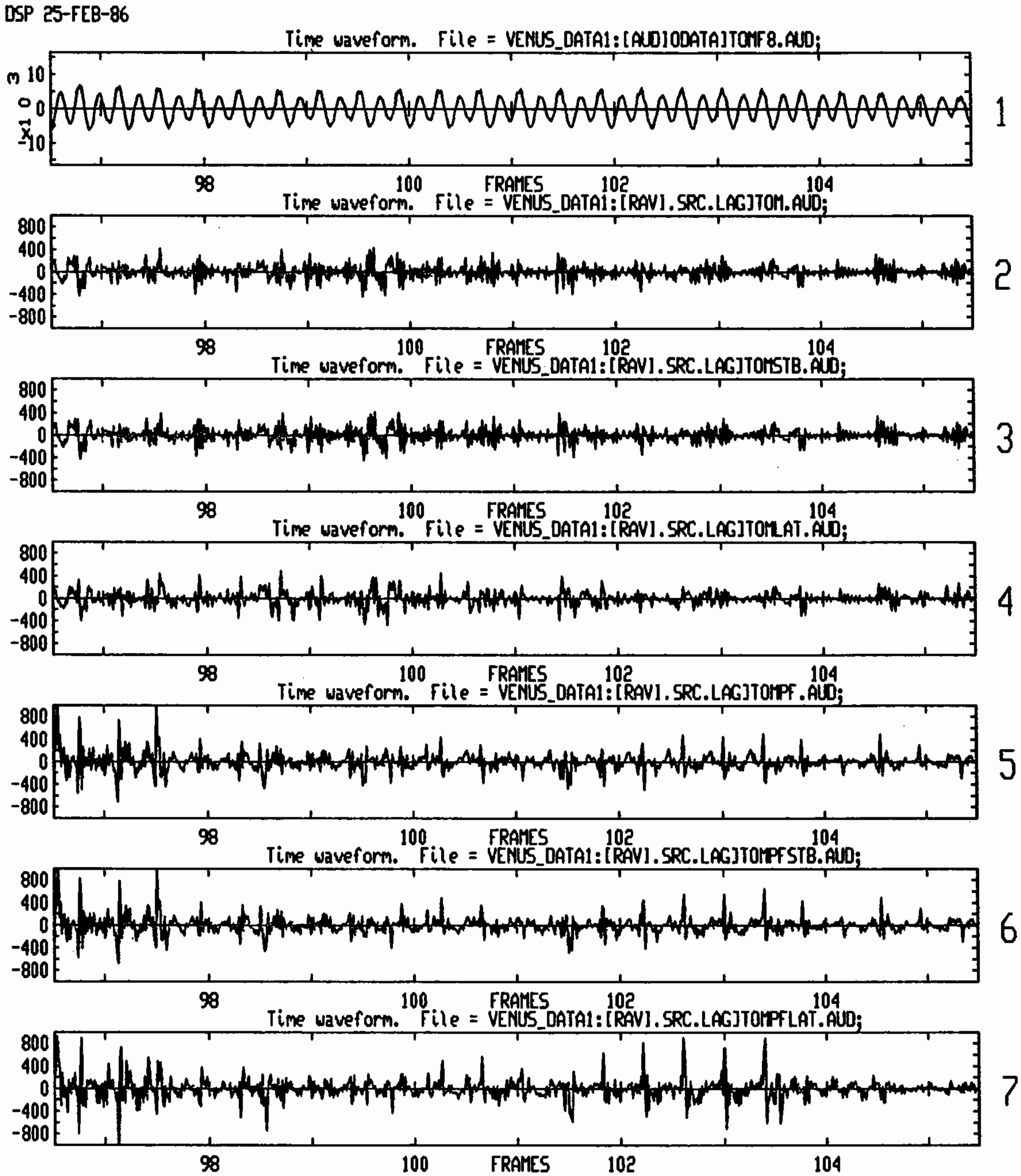


Fig. 5.2 Residual Waveforms (Example 1) (1) Speech Data, (2) Pitch Predicted Residual for Covariance Formulation (F-P), (3) Pitch Predicted Residual for Stabilization Technique (F-P), (4) Pitch Predicted Residual for Lattice Method (F-P), (5) Pitch Predicted Residual for Covariance Formulation (P-F), (6) Pitch Predicted Residual for Stabilization Technique (P-F) and (7) Pitch Predicted Residual for Lattice Method (P-F)

of stabilization and illustrate a certain characteristic associated with the lattice method.

As an overall conclusion, it is highly recommended that a F-P sequence be used in a CELP coder. This is partly due to the better performance it achieves and its effectiveness in removing redundancies in the speech signal. In a F-P sequence, stabilization of the pitch filter is very effective and the Burg method is extremely useful. In a P-F configuration, the pitch filter becomes unstable more often and makes stabilization virtually ineffective.

5.3 Coding Considerations

Coding the formant and pitch predictor coefficients for transmission is an important issue. The formant predictor coefficients are usually first transformed into reflection coefficients or line spectral frequencies (LSFs). When dealing with reflection coefficients, the use of log-area ratios has been shown to have excellent quantization properties [22] in view of their spectral sensitivity. The use of LSFs is also very promising since they show a lower average distortion than log-area ratios and achieve a 30% reduction in bit rate [23,24]. A computationally efficient method of computing LSFs that requires no evaluation or storage of trigonometric functions involves the use of a series expansion in Chebyshev polynomials [25]. Quantizing log-area ratios or LSFs preserves the stability of formant synthesis filters. The use of vector quantization combined with tree searching is also feasible. Vector quantization subsumes any scalar technique and hence, stability can be preserved.

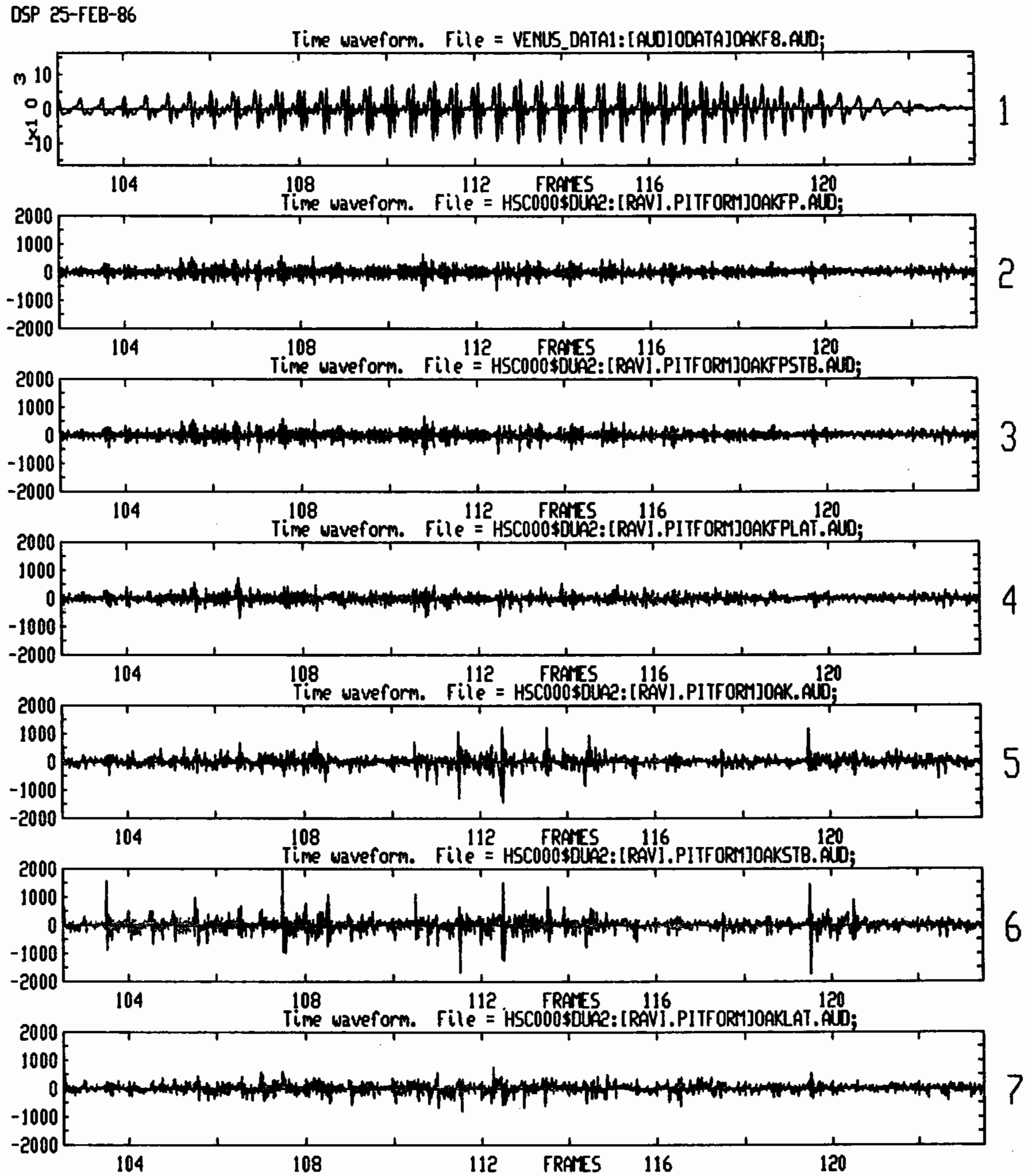


Fig. 5.3 Residual Waveforms (Example 2) (1) Speech Data, (2) Pitch Predicted Residual for Covariance Formulation (F-P), (3) Pitch Predicted Residual for Stabilization Technique (F-P), (4) Pitch Predicted Residual for Lattice Method (F-P), (5) Pitch Predicted Residual for Covariance Formulation (P-F), (6) Pitch Predicted Residual for Stabilization Technique (P-F) and (7) Pitch Predicted Residual for Lattice Method (P-F)

The ideas for coding pitch predictor coefficients form a set of recommendations for future research. Since this thesis studies both the stability and performance of pitch filters, coding the coefficients is the next logical step. Any method of quantization should preserve the stability of the pitch synthesis filter. Therefore, the use of vector quantization is very promising. The codebook used would not contain any entry whose coefficients do not guarantee stability. This is equivalent to the stabilization algorithm introduced in Chapter 3. The optimality criterion involves the minimization of the mean-square error $\varepsilon^2 = \varepsilon_{\min}^2 + \boldsymbol{\delta}^T \boldsymbol{\Phi} \boldsymbol{\delta}$. Therefore, $\boldsymbol{\delta}^T \boldsymbol{\Phi} \boldsymbol{\delta}$ must be minimized. Since formant prediction has been accomplished, (valid assumption due to superiority of F-P sequence), the off-diagonal terms in the matrix $\boldsymbol{\Phi}$ can be neglected. Therefore, the entry in the codebook that minimizes a weighted mean-square distance is chosen. If the diagonal terms in $\boldsymbol{\Phi}$ are treated as being approximately equal, the expression $\boldsymbol{\delta}^T \boldsymbol{\delta}$ or simply the Euclidean distance between the entry in the codebook and the vector of predictor coefficients is minimized. In the case of formant predictors, more complicated distortion measures are employed. Tree searching can be used to diminish the computational burden.

Scalar quantization of the coefficients has been accomplished by Atal [6]. It involves a transformation of the coefficients as shown in Table 5.3. This approach provides no guarantee of stability. Any set of coefficients can be stabilized using the algorithm presented in Chapter 4 and then quantized to preserve stability. The quantization scheme must be designed very carefully.

Other new methods of scalar quantization are now suggested and start by collapsing the polynomial $A_{M+1}(z) = 1 - \beta_1 z^{-(M-1)} - \beta_2 z^{-M} - \beta_3 z^{-(M+1)}$ to a

Parameter	Minimum	Maximum	Bits
$\log(\beta_1 + \beta_2 + \beta_3)$	-1.2	1.2	5
$\beta_1 + \beta_3$	-1.0	1.0	4
$\beta_1 - \beta_3$	-1.0	1.0	4

Table 5.3 One Method of Scalar Quantization

third degree polynomial $A_3(z) = 1 - \beta_1 z^{-1} - \beta_2 z^{-2} - \beta_3 z^{-3}$. Assuming that one commences with a set of coefficients that ensure the stability of the pitch synthesis filter (verified by sufficient test in Chapter 3), a set of reflection coefficients or LSFs corresponding to $A_3(z)$ can be computed. After quantization, the new parameters ensure that the roots of $A_3(z)$ are within the unit circle. The same cannot be said of $A_{M+1}(z)$. The stability test formulated in Chapter 3 is independent of the order of the filter and should be utilized in order to design a quantization scheme that keeps the roots of $A_{M+1}(z)$ within the unit circle. These methods of quantizing reflection coefficients (or log-area ratios) and LSFs should be compared with Atal's approach especially to see if the bit rate is reduced.

Quantizing the log-area ratios generated from the Burg technique is very useful since after quantization, the magnitude of the reflection coefficients are still bounded by 1. Therefore, if the Burg method is used to realize a lattice structured predictor, preserving stability after quantization is no problem. Also, Makhoul has shown that quantization can be accomplished as an integral part of the covariance-lattice algorithm since minimization of the output energy is done stage by stage [21].

Chapter 6 Conclusion

The use of pitch filtering in a speech coder is highly beneficial. Redundancies that are not removed by a formant predictor are removed by a pitch predictor.

The direct form coefficients of a pitch filter are calculated by a system of linear equations derived by minimizing the mean-square error. This does not ensure the stability of the pitch synthesis filter placed at the receiver. The stability of this filter can be checked by the Schur-Cohn test described in Appendix A. A new test for 1, 2 and 3 tap filters is proposed. The test is based on a set of coefficient conditions, independent of the order of the filter and sufficient to ensure a stable pitch filter. The test imposes a negligible computational load on the coder and does not involve any evaluation of transcendental functions. It is also shown that as the order of the filter keeps increasing, this sufficient condition becomes asymptotically necessary and sufficient. Since the order of a pitch filter is very high, the sufficient condition is very tight.

Before obtaining the direct form predictor coefficients, the pitch period must be known. First, a F-P (formant-pitch) sequence is examined. A new method of estimating the pitch period is proposed. This method is formulated by minimizing an approximation to the mean-square prediction residual. The prediction gain is consistently higher than when using Atal's method of pitch estimation.

When a pitch filter is found to be unstable, a convenient stabilization technique based on scaled coefficients which minimizes the loss in prediction gain is developed. Each predictor coefficient is multiplied by a factor c in order to reduce the magnitude of the poles of the filter such that they lie within the unit circle. The loss in prediction gain resulting from stabilization is negligible.

The Burg algorithm is applied in realizing a lattice structured pitch predictor. Here, the stability of the pitch synthesis filter is guaranteed at the outset. For 2 and 3 tap filters, an efficient algorithm that estimates the pitch period for this structure is derived empirically. This avoids a huge computational burden resulting from a theoretically optimal approach. Excellent prediction gains are achieved.

The effect of unstable pitch filters on decoded speech is perceived as the enhancement of quantization noise. A gradual growth of noise is audible when a series of frames in a voiced segment have unstable filters. An impulse-type distortion perceived as a pop or click occurs when one or two frames have unstable filters whose value of $\sum_i |\beta_i|$ is very high. This type of distortion can occur even in a relatively low energy portion of the speech signal.

A CELP coder having a P-F (pitch-formant) arrangement is compared to that possessing a F-P configuration. Estimating the pitch period is done in a different manner in a P-F sequence. Therefore, techniques of estimating the pitch period depend on the filter structure (transversal or lattice) and on the configuration used at the analysis stage (F-P or P-F). A F-P sequence is clearly superior because of the higher overall gain it achieves and its effectiveness in removing pitch pulses. Also, the pitch filter in a P-F sequence is more susceptible to instability.

As an overall conclusion, the coder should use the F-P sequence and stable synthesis filters should be available at the receiver. This results in decoded speech of high quality.

Appendix A. Schur-Cohn Test

The Schur-Cohn test [12,13] can be used to determine whether or not the roots of a polynomial $D(z) = a_0 + a_1 z + \cdots + a_n z^n$ are within the unit circle.

A sequence of polynomials $D_0(z) = D(z)$, $D_1(z)$, $\cdots$, $D_{n-1}(z)$ are defined such that:

$$D_{j+1}(z) = a_0^{(j)} D_j(z) - a_{n-j}^{(j)} z^{n-j} D_j(z^{-1}) \quad \text{for } j = 0 \text{ to } n-1 \tag{A.1}$$

In this recursion, $a_i^{(0)} = a_i$, the original coefficients of $D(z)$.

From the above equation, it is deduced that since $D_j(z)$ is of degree $n-j$, $D_{j+1}(z)$ is at most of degree $n-j-1$. Also, the coefficients of $D_{j+1}(z)$ are derived from those of $D_j(z)$ by the relationship:

$$a_k^{(j+1)} = a_0^{(j)} a_k^{(j)} - a_{n-j}^{(j)} a_{n-j-k}^{(j)} \quad \text{for } k = 0 \text{ to } n-j-1 \tag{A.2}$$

At any stage j, the constant coefficient of $D_j(z)$ is $a_0^{(j)} = \delta_j$ such that at the next stage:

$$\delta_{j+1} = a_0^{(j+1)} = [a_0^{(j)}]^2 - [a_{n-j}^{(j)}]^2 \tag{A.3}$$

A set of necessary and sufficient conditions for the roots of $D(z)$ to lie within the unit circle is $\delta_1 < 0$, $\delta_2 > 0$, $\cdots$, $\delta_n > 0$. This is equivalent to the set of conditions:

$$\begin{aligned} |a_0^{(0)}| &< |a_n^{(0)}| \\ |a_0^{(1)}| &> |a_{n-1}^{(1)}| \\ &\vdots \\ |a_0^{(n-1)}| &> |a_1^{(n-1)}| \end{aligned} \tag{A.4}$$

A.1 Application to Pitch Filters

A 2 tap pitch synthesis filter has a denominator polynomial:

$$\begin{aligned} D(z) &= z^{M+1} - \beta_1 z - \beta_2 \\ &= a_{M+1}^{(0)} z^{M+1} + a_1^{(0)} z + a_0^{(0)} \end{aligned} \tag{A.5}$$

When using the Schur-Cohn test, the first stage in the recursion ($j = 0$) produces only three non-zero coefficients, namely, $a_0^{(1)}$, $a_1^{(1)}$ and $a_M^{(1)}$. At stage j of the recursion, only the coefficients $a_0^{(j+1)}$, $a_1^{(j+1)}$ and $a_{M-j}^{(j+1)}$ need be computed. When $j = M - 1$, only two coefficients $a_0^{(M)}$ and $a_1^{(M)}$ are derived and the condition $|a_0^{(M)}| > |a_1^{(M)}|$ is tested.

The denominator polynomial of a 3 tap pitch synthesis filter is:

$$\begin{aligned} D(z) &= z^{M+1} - \beta_1 z^2 - \beta_2 z - \beta_3 \\ &= a_{M+1}^{(0)} z^{M+1} + a_2^{(0)} z^2 + a_1^{(0)} z + a_0^{(0)} \end{aligned} \tag{A.6}$$

This test is again very convenient since at each stage j, only five non-zero coefficients exist. These coefficients are $a_0^{(j+1)}$, $a_1^{(j+1)}$, $a_2^{(j+1)}$, $a_{M-j-1}^{(j+1)}$ and $a_{M-j}^{(j+1)}$. When $j = M - 4$, the computed coefficients are naturally ordered from $a_0^{(M-3)}$ to $a_4^{(M-3)}$. Then, the number of coefficients is reduced by one at each successive stage and the test terminates when the last condition $|a_0^{(M)}| > |a_1^{(M)}|$ is examined.

For pitch synthesis filters, the Schur-Cohn test is extremely simple to implement. The implementation can be generalized to an l tap filter whose denominator polynomial is:

$$D(z) = a_{M+1}^{(0)} z^{M+1} + a_{l-1}^{(0)} z^{l-1} + \cdots + a_1^{(0)} z + a_0^{(0)} \tag{A.7}$$

In this case, a maximum of $2l - 1$ coefficients must be calculated at each stage in the recursion.

Appendix B. Properties of Quadratic Stability Function

To derive the properties of $f(b^2)$, again assume that β_1 and β_3 have opposite signs. The substitutions $a = |\beta_1 + \beta_3|$ and $b = |\beta_1 - \beta_3|$ are used and $b > a$. The angle γ is the angle at which a point of tangency occurs as is shown in Fig. 3.4(b). Also, assume that the necessary conditions, $a + \beta_2 < 1$ and $b^2 + \beta_2^2 < 1$ are satisfied. The quadratic function $f(b^2) = b^4 + b^2(\beta_2^2 - a^2 - 1) + a^2$ is examined. Now, $f(0) = a^2 > 0$, the parabola is convex up and the roots are:

$$b_{1,2}^2 = \frac{(a^2 + 1 - \beta_2^2) \pm \sqrt{(\beta_2^2 - a^2 - 1)^2 - 4a^2}}{2} \tag{B.1}$$

The roots b_1^2 and b_2^2 are real. This requires that the discriminant $(\beta_2^2 - a^2 - 1)^2 - 4a^2$ be positive or zero.

$$(\beta_2^2 - a^2 - 1)^2 - 4a^2 \geq 0 \Longleftrightarrow |\beta_2^2 - a^2 - 1| \geq 2a \tag{B.2}$$

Now, $|\beta_2^2 - a^2 - 1| = 1 + a^2 - \beta_2^2$ since $\beta_2^2 + b^2 < 1$ and $a < b$. Continuing the above derivation gives:

$$|\beta_2^2 - a^2 - 1| \geq 2a \Longleftrightarrow |a - 1| \geq \beta_2 \tag{B.3}$$

Since $a + \beta_2 < 1$, $|a - 1| = 1 - a$. Now:

$$|a - 1| \geq \beta_2 \Longleftrightarrow 1 \geq a + \beta_2 \tag{B.4}$$

From the necessary condition $a + \beta_2 < 1$; the proof is complete.

If the smaller of the two roots $(a^2 + 1 - \beta_2^2) - \sqrt{(\beta_2^2 - a^2 - 1)^2 - 4a^2}$ is positive (ignoring the division by 2), both roots are positive. It has been shown that $a^2 +$

$1-\beta_2^2$ and $(\beta_2^2-a^2-1)^2-4a^2$ are by themselves positive. Therefore:

$$\begin{aligned} & (a^2+1-\beta_2^2)-\sqrt{(\beta_2^2-a^2-1)^2-4a^2}>0 \\ \iff \quad & (a^2+1-\beta_2^2)^2 > (\beta_2^2-a^2-1)^2-4a^2 \\ \iff \quad & 4a^2>0 \end{aligned} \tag{B.5}$$

The proof is complete since the last inequality is true.

The function defines a convex parabola with two positive roots b_1^2 and b_2^2. Furthermore, $f(b^2)<0$ for $b_1^2<b^2<b_2^2$. Since $0<|\gamma|<\frac{\pi}{2}$ to achieve tangency, therefore $0<\cos\gamma<1$. This requires that

$$0<\frac{a\beta_2}{b_{\max}^2-a^2}<1 \tag{B.6}$$

or

$$b_{\max}^2>a\beta_2+a^2. \tag{B.7}$$

Evaluating $f(a\beta_2+a^2)$ gives:

$$\begin{aligned} f(a\beta_2+a^2) &= (a\beta_2+a^2)^2+(a\beta_2+a^2)(\beta_2^2-a^2-1)+a^2 \\ &= a\beta_2((a+\beta_2)^2-1) \\ &< 0 \end{aligned} \tag{B.8}$$

The last inequality follows from $0<a+\beta_2<1$. Since, $b_{\max}^2>a\beta_2+a^2$ and $f(a\beta_2+a^2)<0$, the only permissible root is b_2^2. The quantity $b_{\max}=b_2$ is the value of b for which the ellipse is tangent to the circle. For the ellipse to lie entirely within the circle, $b<b_{\max}$.

The vertex of the parabola is found by letting the first derivative of $f(b^2)$ with respect to b^2 be equal to 0. The vertex occurs at $b^2=(a^2+1-\beta_2^2)/2$. The point

$a\beta_2 + a^2$ comes to the left of the vertex on the horizontal axis as shown below:

$$a\beta_2 + a^2 < \frac{a^2 + 1 - \beta_2^2}{2} \Longleftrightarrow a + \beta_2 < 1 \tag{B.9}$$

The proof is complete since $a + \beta_2 < 1$ is true by the assumed necessary condition.

Finally, a plot of $f(b^2)$ is shown.

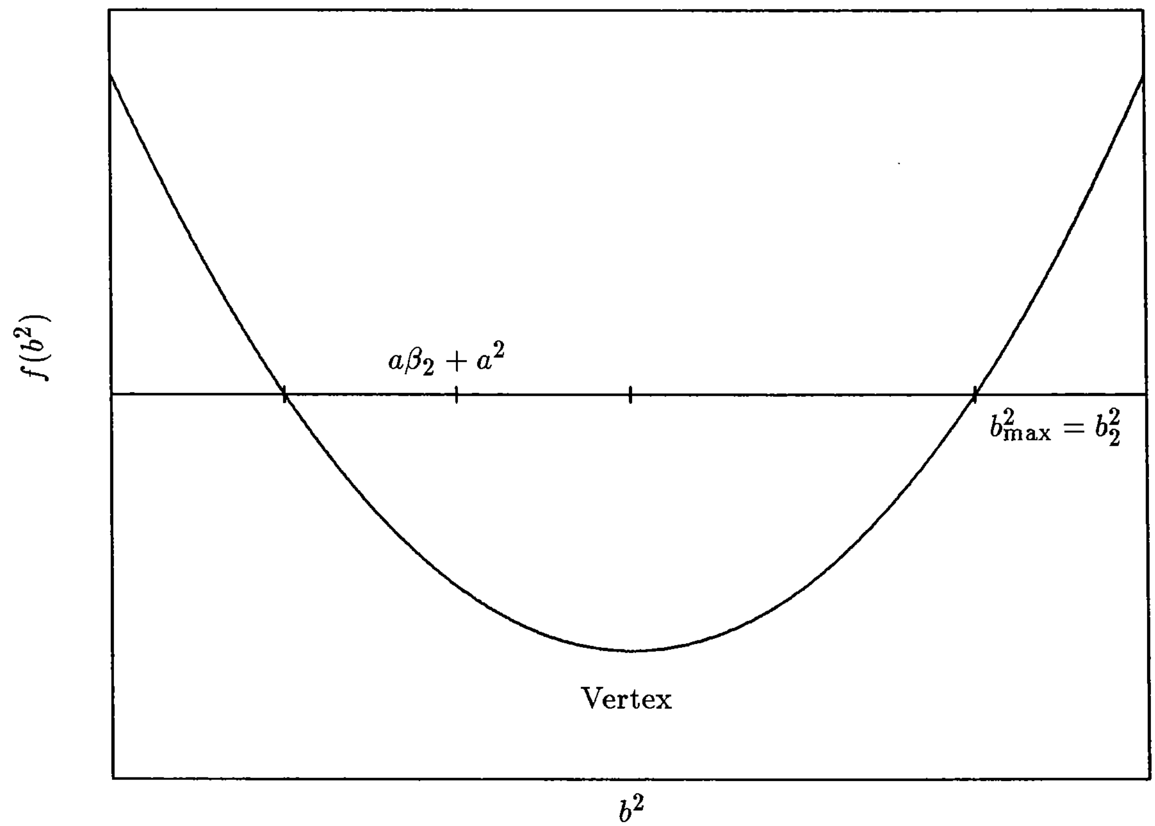


Fig. B.1 Plot of $f(b^2)$ versus b^2

Appendix C. Contents of the Speech Data Files

The speech data files used in the experiments conducted contain different sentences as given below. The first three sentences are spoken by male speakers and the last three are spoken by females.

CATM8: Cats and dogs each hate the other.

ADDM8: Add the sum to the product of these two.

PIPM8: The pipe began to rust while new.

TOMF8: Tom's birthday is in June.

OAKF8: Oak is strong and also gives shade.

THVF8: Thieves who rob friends deserve jail.

References

1. B.S. Atal and M.R. Schroeder, "Predictive coding of speech signals", *Bell System Tech. J.*, pp. 1973–1986, Oct. 1970.
2. B.S. Atal and S.L. Hanauer, "Speech analysis and synthesis by linear prediction of the speech wave", *J. Acoust. Soc. Amer.*, pp. 637–655, Aug. 1971.
3. N.S. Jayant and P. Noll, *Digital Coding of Waveforms*, Prentice-Hall, 1984.
4. J.L. Flanagan, M.R. Schroeder, B.S. Atal, R.E. Crochiere, N.S. Jayant and J.M. Tribolet, "Speech coding", *IEEE Trans. Commun.*, vol. COM-27, pp. 710–736, April 1979.
5. M.R. Schroeder and B.S. Atal, "Code-Excited Linear Prediction (CELP): High-quality speech at very low bit rates", *Proc. Int. Conf. Acoust., Speech, Signal Processing*, Tampa, Florida, pp. 25.1.1–25.1.4, March 1985.
6. B.S. Atal, "Predictive coding of speech at low bit rates", *IEEE Trans. Commun.*, vol. COM-30, pp. 600–614, April 1982.
7. L.R. Rabiner and R.W. Schafer, *Digital Processing of Speech Signals*, Prentice-Hall, 1978.
8. B.S. Atal and M.R. Schroeder, "Predictive coding of speech and subjective error criteria", *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-27, pp. 247–254, June 1979.
9. J. Makhoul and M. Berouti, "Adaptive noise spectral shaping and entropy coding of speech", *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-27, pp. 63–73, Feb. 1979.
10. P. Noll, "On predictive quantizing schemes", *Bell System Tech. J.*, pp. 1499–1532, May-June 1978.
11. A.V. Oppenheim and R.W. Schafer, *Digital Signal Processing*, Prentice-Hall, 1975.
12. E.I. Jury, *Theory and Application of the z-Transform Method*, John Wiley and Sons, 1964.
13. S. Haykin, *Adaptive Filter Theory*, Prentice-Hall Inc., 1986.
14. H.W. Schussler, "A stability theorem for discrete systems", *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-24, pp. 87–89, Feb. 1976.
15. R. Gnanasekaran, "A note on the new 1-D and 2-D stability theorems for discrete systems", *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-29, pp.1211–1212, Dec. 1981.
16. Y. Bistriz, "Zero location with respect to the unit circle of discrete-time linear system polynomials", *Proc. IEEE*, vol. 72, pp. 1131–1142, Sept. 1984.
17. R.A. Silverman, *Introductory Complex Analysis*, Dover Publications, 1967.
18. J. Makhoul, "Linear prediction: A tutorial review", *Proc. IEEE*, vol. 63, pp. 561–580, April 1975.

19. D.T.L. Lee and M. Morf, "A novel innovations based time-domain pitch detector", *Proc. Int. Conf. Acoust., Speech, Signal Processing*, Denver Col., pp. 40–44, April 1980.

20. M.L. Honig and D.G. Messerschmitt, "Comparison of adaptive linear prediction algorithms in ADPCM", *IEEE Trans. Commun.*, vol. COM-30, pp. 1775–1785, July 1982.

21. J. Makhoul, "Stable and efficient lattice methods for linear prediction", *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-25, pp. 423–428, Oct. 1977.

22. R. Viswanathan and J. Makhoul, "Quantization properties of transmission parameters in linear predictive systems", *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-23, pp. 309–321, June 1975.

23. F. K. Soong and B. W. Juang, "Line spectrum pair (LSP) and speech data compression", *Proc. Int. Conf. Acoust., Speech, Signal Processing*, San Diego Cal., pp. 1.10.1–1.10.4, March 1984.

24. G. S. Kang and L. J. Fransen, "Application of line spectral pairs to low bit rate speech encoders", *Proc. Int. Conf. Acoust., Speech, Signal Processing*, Tampa Fl., pp. 7.3.1–7.3.4, April 1985.

25. P. Kabal and R. P. Ramachandran, "The computation of line spectral frequencies using chebyshev polynomials", (accepted for publication in *IEEE Trans. Acoust., Speech, Signal Processing*).