The Stability of Pitch Synthesis Filters in Speech Coding

by

Victor T.M. Lam
B.Eng.

Department of Electrical Engineering
McGill University
Montréal, Canada
June, 1985

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

ABSTRACT

This thesis studies the problem of instability in pitch synthesis filters found in Adaptive Predictive Coding of speech. The performance of such coders is often improved by adding, in the analysis stage, a pitch predictor which removes the redundancy due to the pitch periodicity in the speech signal. The pitch synthesis filter used to restore this periodicity is known to be quite susceptible to instability, causing distortion in the decoded speech. The system function of the synthesis filter has a denominator polynomial of relatively high degree, ranging from 20 to 120 for a signal sampled at 8 kHz. Testing the stability of the filter by solving for the roots of the polynomial is time consuming and impractical for real time applications.

This study establishes a simple criterion to check the filter stability for a given frame of speech, it also proposes several stabilization schemes, and examines the effects of stabilizing the filter on the decoded speech. One criterion determines the filter stability by checking the sum of the magnitudes of the predictor coefficients against unity. It introduces a negligible delay and is shown to be a sufficient condition for the stability of the pitch synthesis filter.

SOMMAIRE

Ce mémoire examine les problèmes d'instabilité dans les filtres de synthèse de périodicité qui font partie des systèmes de codage prédictif adaptif pour le traitement de la parole. Les performances de ces codeurs sont souvent améliorés en incorporant un prédicteur de périodicité qui exploite la redondance de la périodicité dans le signal de parole. Le filtre de synthèse utilisé dans la restoration de la périodicité possède de fortes tendances d'instabilité ce qui cause des distorsions dans la parole décodée. La fonction de transfert du filtre de synthèse est caractérisée par un polynôme de degré élevé variant entre 20 et 120 pour un signal échantillonné à 8 kHz. Vérifier la stabilité du filtre en essayant de trouver les racines du polynôme est une tâche qui demanderait trop de temps pour un ordinateur fontionnant un temps réel.

Ce travail établie un critère simple pour déterminer la stabilité du filtre dans une fenêtre d'analyse donnée du signal. Plusieurs méthodes de stabilisation y sont également proposées ainsi que l'examen des effets de la stabilisation sur le signal reconstruit. Parmi les critères examinés, l'un de ceux-ci vérifie la stabilité en comparant la somme des valeurs absolues des coefficients du prédicteur par rapport à l'unité. Cette méthode simple demande un temps de calcul négligeable et il est démontré que ceci constitue une condition suffisante pour la stabilité du filtre de synthèse de périodicité.

ACKNOWLEDGEMENT

I would like to express my gratitute to Dr. Peter Kabal for initiating this research project and for his constant interest and guidance throughout. My thanks also go to Dr. Douglas O'shaughnessy, Dr. Michael Beyrouti and Dr. Michel Fortier for their willingness to offer advice and discuss related topics.

The experimental work of this thesis was carried at BNR/INRS laboratory at Nun's Island Montréal. The help from many of its staff members, including Ms. Ghislaine Gauthier and Ms. Mary Baribeau from the Information Resource Centre, Jeff Abbott from the Software System and Jean-Luc Moncet for occasional technical discussion of the subject matter is much appreciated.

Last but not least, I like to take this opportunity to thank my parents for their unfailing moral and financial support during the entire course of my studies.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1 Introduction

1.1 Adaptive Predictive Coding of Speech

An efficient speech coding scheme is one that maximizes the utilization of a given channel capacity, or equivalently one that minimizes the bit rate requirement to transmit the speech signal. As far back as in the late 1930's, speech signal was known to have contained some redundant components which could be predicted from the recent history of the signal. The remaining signal after prediction is known as the residual, and it is the difference between the input signal and the predicted signal.

Predictive coding [ELIA(55)] takes advantage of this predictability of a signal in reducing the transmission load of the channel. If the current sample can be predicted from the past several samples in the transmitter, the approximate version of the same sample can similarly be reproduced from the past several reconstructed samples in the receiver. The residual however, which is unpredictable, must be transmitted to fill the missing information in the output. By nature, the residual is relatively low in amplitude; therefore it requires fewer bits/sample to code the residual than it does to code the original input signal itself. The obligation to transmit only the low-amplitude residual signal is one

major feature that makes predictive coding an efficient coding scheme.

The application of the predictive coding to speech signal was pioneered by Atal in the late 1960's [ATAL(70)]. Two sources of redundancies in speech signal are (i)- the lack of flatness in the short-time spectral envelope [SCHR(66)], reflecting its correlative nature, and (ii)- its quasi-periodicity during voiced segments. These two redundancies can be described as the near-sample-based redundancy caused by the slowly varying vocal tract shape, and the distant-sample-based redundancy as a result of the rhythmic glottal excitation or vibration [JAYA(84)].

To effectively remove these redundancies requires different predictors, each tailored to match the characteristics of one specific form of redundancy. They are appropriately called the formant predictor — for predicting the first type of redundancy, and the pitch predictor — for predicting the second redundancy described above. Since the speech signal changes its characteristics from time to time, it is necessary to have an adaptive or time-varying predictor to keep track of this change. In other words, each set of the predictor coefficients used by a specific predictor must be constantly updated to match the changing speech characteristics; therefore the term Adaptive Predictive Coding (APC). In the course of the development of APC system, many of its algorithms have been modified and improved, and new features are being added to better its performance.

The digital channel in APC is used to transmit two quantities: the quantized prediction residual, and the side information which includes primarily the predictor coefficients and the step size of the quantizer. The transmission of the prediction residual normally occupies a significantly larger proportion of the total number of bits, thus an efficient quantization of the residual is essential in obtaining the lowest possible bit rate for a given speech quality. Studies [ATAL(80)] indicate that the high-amplitude portion of the residual is more significant than

the low-amplitude counterpart, and that an accurate quantization of the former reduces the perceptual distortion in the decoded speech. The studies also discover that even when only the higher-amplitude part of the signal is retained by center-clipping process, very little or no distortion is observed. The above finding provides an effective alternative in reducing the number of bits required to code the residual in APC, as we then need to transmit only the high amplitude part instead of the complete residual signal.

Another new feature later introduced to the original system, and which significantly improves the perceptual quality of the output speech, is the noise spectral shaping filter $N(z)$. Quantization noise — the coding error of APC[†], can be treated as a white noise with a flat spectrum [ATAL(79)]. When the noise spectrum is compared to a typical short-time voiced speech spectrum, we see that the SNR[‡] is high in the formant regions where the noise is effectively masked by the speech signal, but rather low and even negative (in dB) in between the formants (or in the valleys) due to the relatively low signal energy in these regions. Hence, the large part of the perceived noise originates from these 'valley' regions. The noise spectral shaping filter is designed to distribute the noise power from one frequency to another, so that a more uniform SNR across the spectrum is achieved.

In the earlier version of APC system described above, a relatively high bit rate $\geq$ 9.6 kbps is usually required. With this capacity of bit rate, instantaneous quantization and sample by sample coding of the quantized residual is effective, and it is possible to generate quantization noise with an approximately flat spec-

---

† To be verified in Chapter 2.

‡ Signal to quantization noise ratio.

trum. Recently, there is a quest for reducing the bit rate to below 5 kbps, in hope of transmitting speech through the existing analog voice channel.

In response to the need for low bit rate APC, a new coding scheme based on the principle of APC has been investigated since 1982 [ATAL(82.B),(82.C),(85)]. It is called the Code-Excited Linear Prediction(CELP) coder [ATAL(85)]. This new coder is a derivative of APC; it modifies certain aspects of the original APC system. For instance, the prediction in CELP is based on the past input of the predictor instead of on the past reconstructed output, and a vector quantization strategy is used to code the residual, which consequently requires only 2 kbps to code the residual as compared to the 8 kbps requirement in APC system for 1-bit/sample accuracy.

Presently, CELP still demands a heavy computation due to the need for an exhaustive search during the residual model selection process. But preliminary results have shown promises that with further simplification of the algorithm and a more appropriate design of the dictionary, this new derivative of APC should be able to produce high quality speech at the targeted bit rate of under 5 kbps. More details of this new coder along with that of the APC system will be described in Chapter 2.

1.2 Problem in Pitch Synthesis

In the analysis phase of either APC or CELP system, the stability of the two FIR filters involved is generally guaranteed. Instability occurs usually in the synthesis stage, in particular — the pitch synthesis, where the filters are autoregressive (IIR). In formant synthesis, if the predictor coefficients (which are identical to the ordinary LPC coefficients) are computed by using auto-correlation method [OPPE(75)] or modified covariance method [ATAL(79)], the stability of

the synthesis filter can be guaranteed. Section 3.2 provides a simple proof of the general stability condition of the formant synthesis filter. In pitch synthesis however, for reasons to be studied and justified later, the synthesis filter is quite vulnerable to instability.

The pitch predictor in the early APC system contains only a single tap; subsequent research indicates that to better ensure higher synchronism between the sampling rate and the actual speaker pitch (peak), a multiple tap ($m > 1$)[†] pitch predictor is recommended. More recent APC analyzer using 3-tap pitch predictor confirms that the 3-tap (or in general the multi-tap) predictor does indeed improve the prediction gain substantially. But at the same time, it also increases the number of unstable frames for a given speech file.

The effect of unstable synthesis filter, particularly the unstable pitch synthesis filter, has been found to cause an annoying effect in the perception of the output speech. The degradation due to pitch synthesis filter is characterized by the presence of 'beeps' or click sounds in the output speech. Theoretically, a system is unstable because some of its poles lie outside the unit circle. To solve the stability problem requires (i)-testing the system stability, to see if all the roots of the system polynomial are inside the unit circle, and (ii)-stabilizing the system, which involves modifying the coefficient values of the characteristic polynomial to make all the roots to be inside the unit circle. In the pitch synthesis, the problem is further compounded by the difficulty in determining the stability status of the pitch synthesis filter, as its characteristic equation is in the form of high degree (20 to 120) polynomial. Using conventional methods to test the stability status demands a heavy computation. It is this instability and the associated problems in the pitch synthesis filter that initiate and shape the development of this thesis.

[†] Where m is the order, or the number of taps, of the pitch predictor.

1.3 Organization of the Report

In Chapter 2 which follows, we describe the principles of APC system and its new derivative CELP. The two categories of coders share many common features as well as a common problem, i.e., the instability during the pitch synthesis process. This chapter compares different ways of operations and the individual characteristics of the two coders.

Chapters 3, 4 and 5 contain the findings and results based on both theoretical and experimental observations on the problem of instability in pitch synthesis filter; they form the main trunk of this thesis. Chapter 3 analyzes the instability problem in detail, establishes the true stability region of the pitch synthesis filter, derives as well as verifies the sufficient condition for stability. The reliability of using the sufficient condition as opposed to the necessary condition which is difficult to generalize is also estimated in this chapter. In Chapter 4, we propose two basic methods to stabilize a single-tap filter, and two different approaches to stabilize multi-tap filters when using unity-replacement method; these methods are tested separately to show their relative efficiencies in terms of the consequent loss in prediction gain. Chapter 5 examines the effect of the stabilization process on the output speech, both objectively and subjectively. In particular, the effect on CELP is examined where the results and observations are shown to be applicable as well to APC system under certain valid assumption.

Chapter 6 summarizes some of the important observations and findings of the experimental tests and simulation work carried out during the course of this research project, from which a conclusion is drawn about the instability of the pitch synthesis filter, its negative effect, as well as the effect of its stabilization on the the output speech.

Chapter 2 Principles of APC and CELP

This chapter describes the basic principles of APC and its derivative CELP, which have been touched upon briefly in the 'Introduction'. Both of these speech coders use pitch prediction as part of their algorithm in the encoding process. APC removes the speech redundancies, transmits the residual, and attempts to reconstruct the original speech by combining the transmitted residual with the predicted signal at the receiver. CELP operates basically on the same principle of APC, only it does not transmit the residual. Instead, a model of the residual is created and used at the receiver to drive the synthesis system.

2.1 Adaptive Predictive Coding (APC)

Although the principle of APC is rather simple, its operation — especially when two predictors are used — is quite involved. The analysis stage in APC is the more complicated part of the system. But once it is understood, the corresponding synthesis is simply the inverse operation. A good approach to make the operation of APC analysis more comprehensible is to assume the use of a single predictor, and with the aid of block diagram shown in Fig. 2.1.

The basic APC system in Fig. 2.1 only has a formant predictor. The function of the predictor is to predict the input signal $s(n)$ by making its estimate $\tilde{s}(n)$

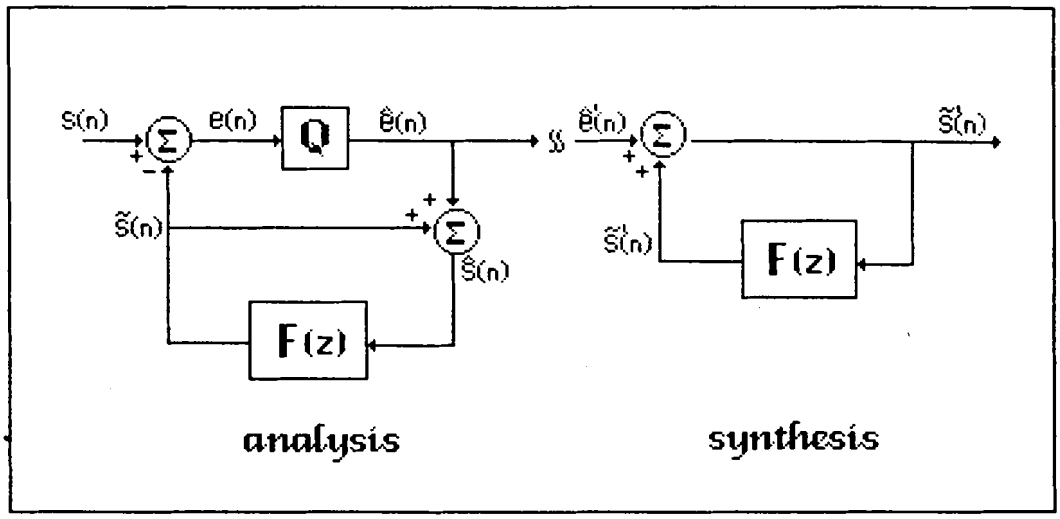


Fig. 2.1 Basic APC system without pitch predictor

based on the previously reconstructed samples $\hat{s}(n-k)$, $k = 1, 2, ..., p$. The currently reconstructed sample $\hat{s}(n)$ is made up of two components, namely the quantized residual $\hat{e}(n)$ and the current estimate or the predicted sample $\tilde{s}(n)$. The unquantized residual $e(n)$ is formed by taking the difference between the input signal and the predicted signal, i.e., $e(n) = s(n) - \tilde{s}(n)$. This intricate relationship between the various quantities above becomes obvious when the following relations are clear, i.e.,

$$\hat{s}(n) = \tilde{s}(n) + \hat{e}(n) \tag{2.1}$$

where:

$$\tilde{s}(n) = \sum_{k=1}^{p} \alpha_k \hat{s}(n-k) \tag{2.2}$$

The key is to recognize that the predicted sample $\tilde{s}(n)$ is formed from the previously constructed samples $\hat{s}(n-k)$ as defined in Eq. (2.1). Having understood the basic APC system involving only the formant predictor, let us examine the typical APC which includes a pitch predictor.

2.1.1 APC Analysis

In APC system, the two predictions denoted as F (formant prediction) and P (pitch prediction) can be implemented in either sequence. This sequence issue still remains controversial at the present time. Many maintain that so long the reverse sequence is used in the synthesis, the sequence of predictions is irrelevant to the performance of the system. The earlier APC adopts the formant prediction followed by the pitch prediction (F-P) sequence; nevertheless, a few prefer to use the (P-F) prediction sequence. If a predictor is first used in the prediction process so that the input to the predictor is the original speech signal, the resulting prediction gain is usually higher than if it were used as a second predictor. But regardless of the sequence used, there is always a ceiling to which the total prediction gain can reach.

From our point of view and experience, (F-P) is more desirable to use for two reasons; (1)-it gives a slightly higher total prediction gain, and (2)-as the input to the pitch prediction is the first residual with small amplitude instead of the original speech, the generated pitch predictor coefficients tends to produce a more stable pitch synthesis filter at the receiver.

Assuming (F-P) prediction sequence, Fig. 2.2 is the block diagram of the complete APC system. The basic operation in the analysis is similar to that described previously, except now the prediction is carried out in two stages — the formant prediction followed by the pitch prediction.

Similar to the operation of the basic APC system, the formant predictor $F(z)$ in Fig. 2.2 attempts to predict the current sample of the input speech $s(n)$ by making an estimate $\tilde{s}(n)$, and their difference called the first residual

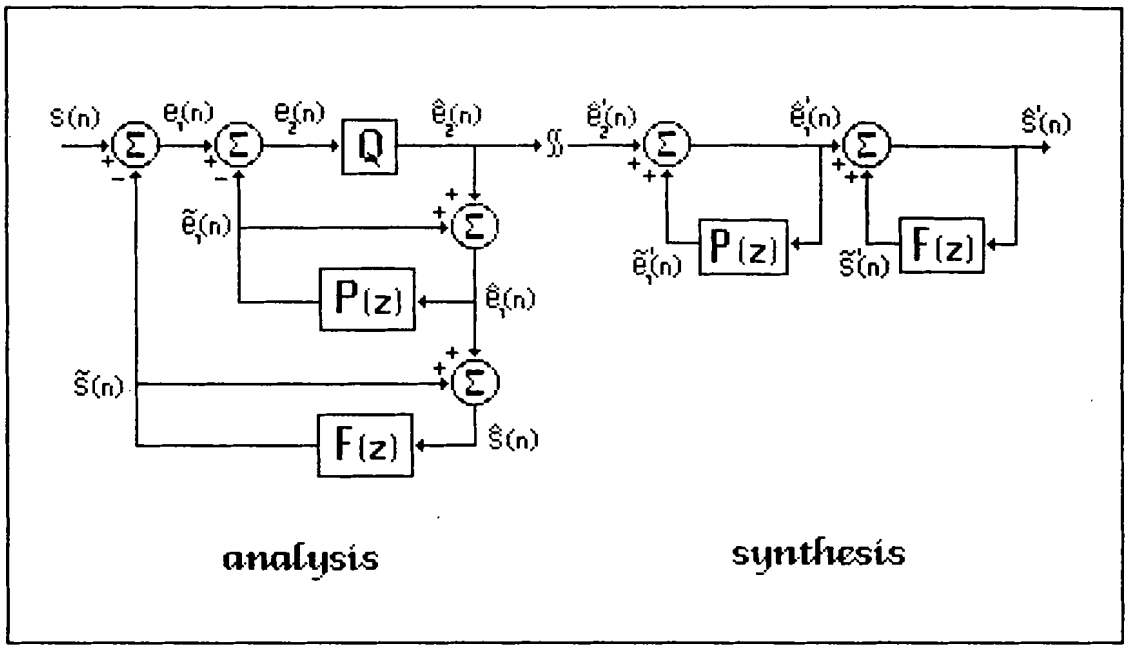


Fig. 2.2 APC system including pitch predictor

$e_1(n)$ is formed. Based on the previously reconstructed residual $\hat{e}_1(n-M)$[†], the pitch predictor $P(z)$ predicts its current value $\tilde{e}_1(n)$, and forms another difference called the second residual $e_2(n)$ by subtracting the currently predicted $\tilde{e}_1(n)$ from $e_1(n)$. The second residual is then quantized, and transmitted to the receiver. The same cycle repeats for the next sample.

The above analysis assumes the prediction sequence (F-P), which is the scheme we will use in our study. The other sequence (P-F) is formed when the two predictors $P(z)$ and $F(z)$, as well as the two residuals $e_1(n)$ and $e_2(n)$ in Fig. 2.2 are interchanged. In general, the inclusion of the pitch predictor generates extra prediction gain, but the formant prediction contributes the larger proportion of the total prediction gain.

---

† Assuming 1-tap pitch predictor.

In APC system with noise shaping (APC-NS) [MAKH(79.B)][ATAL(79)], the quantization noise is fed into a filter $N(z)$, whose output is then subtracted from the residual. It can be shown that the difference between the output and the input of APC-NS (in frequency domain notation) is equivalent to the quantization noise itself filtered by $(\frac{1-N(z)}{1-P(z)})$†. The noise spectral shaping filter $N(z)$ is designed such that $N(z) = P(\rho z^{-1})$, where ρ is the parameter which controls the bandwidth of the zeros of $(1-N(z))$. As ρ ranges from 1 to 0, $N(z)$ varies from $P(z)$ to 0, thus causing the noise spectrum to be shaped by a factor varying from 1(no effect) to $(\frac{1}{1-P(z)})$. The idea behind the noise spectral shaping can be rationalized from the following point of view: the filter $(1 - P(z))$ tends to flatten a typical voiced speech spectrum which has strong formant resonances, thus its inverse filter $(\frac{1}{1-P(z)})$ must have an inverse effect on the presumably white quantization noise, i.e., it tends to mold the noise spectrum to follow the shape of the speech spectrum. When $0 < \rho < 1$, the factor $(1 - P(\rho z^{-1}))$ controls the weights to be used for redistributing the energy across the spectrum. Note however that with noise spectral shaping, the absolute SNR is increased as a result of the net shift of noise energy from the high-frequency region to the low-frequency region, or more accurately from the corresponding 'valley' regions to the formant regions. Nevertheless, the more evenly distributed SNR across the spectrum produces a better perceptual quality speech [ATAL(79)].

2.1.2 APC Synthesis

The synthesis part of APC is simply the inverse operation of the analysis. In Fig.2.2, $\hat{e}_2(n)$ denotes the transmitted second residual (quantized) where the prime ($'$) indicates that it is corrupted by channel noise. When it is combined with

---

† Where $P(z)$ may represent the combined predictor of formant/pitch predictors.

the predicted sample of the first residual $\tilde{e}'_1(n)$, the first residual is reconstructed $(\hat{e}'_1(n))$. Finally, it is added to the predicted sample of the input $\tilde{s}(n)$ to form the output speech $\hat{s}'(n)$.

Observe that the synthesis structure of the APC is actually included in the analysis structure. Naturally, the input to this synthesis structure in the analysis is the quantized second residual, whereas the input to the synthesis system in the receiver is the quantized residual which has been corrupted by channel noise.

In the absence of channel error, the decoded output can be expressed (without the prime) as:

$$\hat{s}(n) = \tilde{s}(n) + \hat{e}_1(n). \tag{2.3}$$

From the analysis structure in Fig. 2.2, we have

$$\tilde{s}(n) = s(n) - e_1(n) \tag{2.4}$$

and

$$\begin{aligned} \hat{e}_1(n) &= \hat{e}_2(n) + \tilde{e}_1(n) \\ &= \hat{e}_2(n) + [e_1(n) - e_2(n)] \end{aligned} \tag{2.5}$$

Substituting Eq. (2.4) and Eq. (2.5) into Eq. (2.3), we have

$$\begin{aligned} \hat{s}(n) &= [s(n) - e_1(n)] + [e_{2q}(n) + e_1(n) - e_2(n)] \\ &= s(n) + [e_{2q}(n) - e_2(n)] \\ &= s(n) + q(n) \end{aligned} \tag{2.6}$$

Eq. (2.6) shows that the coding noise in APC is simply equal to the quantization noise.

2.2 Code-Excited Linear Prediction (CELP)

In an attempt to reduce the bit rate of the existing APC system, a new coder CELP has been suggested recently [ATAL(85)]. Just as in APC, CELP uses two predictors (formant and pitch) to generate a residual signal. However, the prediction processes in CELP are more straightforward than that in APC, where the two residuals are respectively $e_1(n)$ — the difference between the input signal and the formant predicted output, and $e_2(n)$ — the difference between the input to the pitch predictor and its predicted output. In other words, all the predictions are based on the past inputs to the predictors. Furthermore, the second residual $e_2(n)$ is not transmitted, but is used only as a reference in the residual model selection process.

A code-book or dictionary containing 1024 ($= 2^{10}$) waveforms provides the data base for modeling the residual signal. These waveforms, each 40-sample long, are generated by a Gaussian noise generator. Each entry in the dictionary is tested for its resemblance to the actual residual segment by passing it through the synthesis system†, and the output is subtracted from the corresponding original input segment (40 samples). The resulting difference is then spectrally weighted, as in noise spectral shaping, to emphasize/de-emphasize the importance of the error according to the human auditory perception, and the entry with the minimum weighted mean-square error is selected to represent the residual. The number of bits required to code 40 samples of residual using the above vector quantization scheme requires only 10 bits/40 samples. In other words, 2 kbps bit rate is required to code the residual signal.

The analysis part of CELP is much less involved than the one in APC. Math-

† The process is identical to the one carried out in the APC system.

ematically, the system functions of the two inverse filters and the corresponding synthesis filters are:

Formant : $(p^{th} - order)$

Predictor :

$$F(z) = \sum_{i=1}^{p} \alpha z^{-i} \tag{2.7}$$

Inverse Filter :

$$A_f(z) = 1 - F(z) \tag{2.8}$$

Synthesis Filter :

$$\begin{aligned} H_f(z) &= \frac{1}{A_f(z)} \\ &= \frac{1}{1 - \sum_{i=1}^{p} \alpha z^{-i}} \\ &= \frac{z^p}{z^p - \alpha_1 z^{p-1} - \alpha_2 z^{p-2} - ... - \alpha_p} \end{aligned} \tag{2.9}$$

Pitch : $(3^{rd} - order)$

Predictor :

$$P(z) = \beta_1 z^{-(M-1)} + \beta_2 z^{-M} + \beta_3 z^{-(M+1)} \tag{2.10}$$

Inverse Filter :

$$A_p(z) = 1 - P(z) \tag{2.11}$$

Synthesis Filter :

$$\begin{aligned} H_p(z) &= \frac{1}{A_p(z)} \\ &= \frac{1}{1 - \beta_1 z^{-(M-1)} - \beta_2 z^{-M} - \beta_3 z^{-(M+1)}} \\ &= \frac{z^{M+1}}{z^{M+1} - \beta_1 z^2 - \beta_2 z - \beta_3} \end{aligned} \tag{2.12}$$

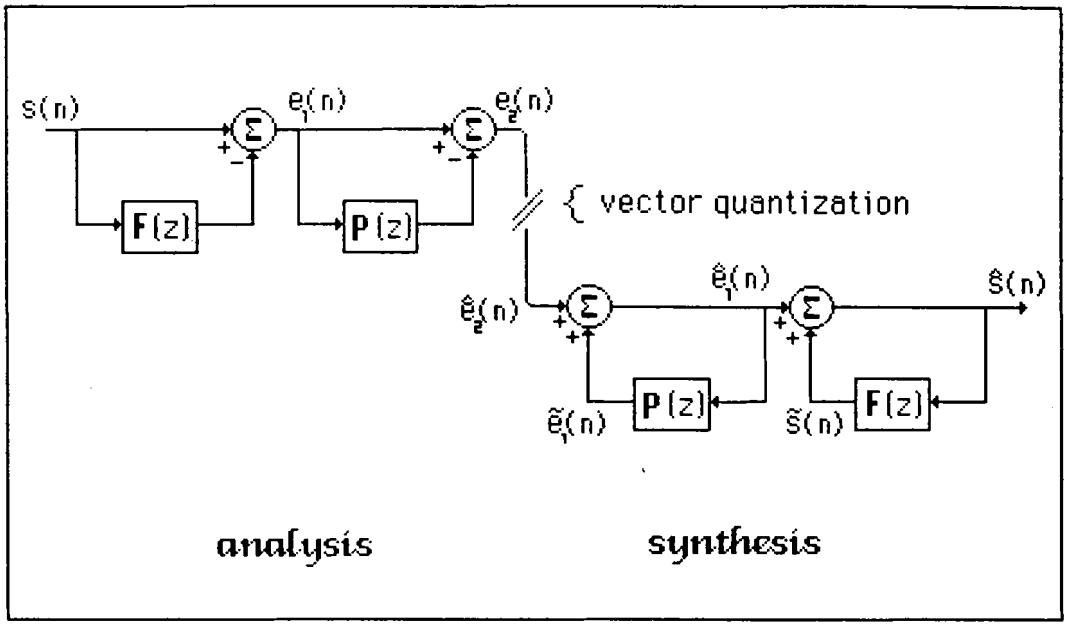


Fig. 2.3 Analysis/Synthesis processes in CELP.

In short, the analysis/synthesis processes of CELP are simply speech redundancy removal and restoring operations. The simple block diagram in Fig. 2.3 illustrates this clear-cut relationship.

2.3 Comparison between APC and CELP

It is evident from the above description that the essential difference between APC and CELP is the way in which the residual is generated, quantized and transmitted. In both systems, the residual is defined as the difference between the input signal and its predicted signal. But in APC system, the prediction is based on the past reconstructed signal; whereas in CELP, it is based directly on the past input signal. In APC, the residual is quantized sample by sample and is transmitted to drive the synthesizer; while in CELP, a residual model instead is generated and used as excitation to the synthesizer. Effectively, vector quantization is used in CELP to code the residual. As a result, CELP needs a

much lower bit rate to code the speech than the APC, which spends most of its bits on directly transmitting the residual.

In Section 2.1, we showed that the noise in APC system is equivalent to the quantization noise. In CELP, the residual model is equivalent to the clean (unquantized) residual plus the noise introduced by the model. Assuming linearity in the synthesis system, it can be shown that the output of the CELP system consists of two independent components, one due to the unquantized residual (which according to Fig. 2.3 gives rise to the original signal), and the other due to the vector quantization noise. In other words, the noise of the CELP is the vector quantization noise filtered by the synthesis system.

Chapter 3 Pitch Synthesis Filter

This chapter concerns with the stability or the problem of instability of the pitch synthesis filter $H_p(z)$. Because of the difficulty in checking the stability of $H_p(z)$ by conventional methods, we investigate the possibility of finding a simple yet reliable algorithm which can determine the stability status of $H_p(z)$ for a given frame of input. The cause of instability is discussed in terms of the filter coefficients, and in terms of the characteristic of the input data upon which the generation of the coefficients are based. Our approach to arriving at the optimal stability criterion is presented, and two criteria are compared in terms of their reliability in predicting the stable status of a speech file. Based on the actual stability region derived numerically and later verified by Jury's constraints, we establish the necessary and the sufficient conditions for the stability of the pitch synthesis filter. At the end, two approximate models of the true stability regions for the 2-tap and the 3-tap filters are constructed, and are used to determine the reliability of using the simple algorithm in testing the stable status of the pitch synthesis filter.

3.1 Definition of Stability

By definition, a stable system is one for which a bounded input produces

a bounded output. A linear shift invariant system is stable if and only if the impulse response of the system is such that

$$\sum_{k=-\infty}^{\infty} |h(k)| < \infty. \tag{3.1}$$

It can be shown that if Eq. (3.1) is true, then any bounded input, say, $|x(n)| < M$ for all n produces a bounded output

$$|y(n)| = |\sum_{k=-\infty}^{\infty} h(k)x(n-k)| \leq M \sum_{k=-\infty}^{\infty} |h(k)| < \infty. \tag{3.2}$$

The APC system is not shift invariant. However, if we consider only one frame of speech input at a time, and ensure that the impulse response for each frame is stable, the stability of the system can be guaranteed. But because of the adaptive nature of the system, an instability in one frame may or may not lead to a distortion. The issue of distortion due to instability is to be dealt with in a later chapter. At the present time, we are concerned only with the general criterion for the stability of the pitch synthesis filter.

The $z-$transform of $h(n)$, or the system function of the pitch synthesis filter in general can be expressed as

$$H_p(z) = \frac{N(z)}{D(z)}, \tag{3.3}$$

where $N(z)$ has multiple roots at the origin, and $D(z)$ is a special polynomial of very high degree. There are only $(m+1)$ non-zero coefficients, where m is the number of taps used by the filter. In Eq. (2.12), we have the system function $H_p(z)$ for $m = 3$, where the lag M is aligned with the middle tap.

The basic rule to determine the stability of $H_p(z)$ is to check if the poles of the filter or equivalently the roots of $D(z)$ are all inside the unit circle on the z-plane. However, this method of directly solving for the roots is efficient

only if $D(z)$ is a low degree polynomial such as in the formant synthesis filter $H_f(z)$ in Eq. (2.9), where the degree of polynomial normally does not exceed 12. In $H_p(z)$, the degree of the polynomial equals to $(M+m-1)$[†]. Thus when the sampling rate is 8 kHz, M in general represents the average pitch of the speech in an analysis frame, which may range from about 60 to 100 samples in male voices and usually higher in female voices. It is obvious that the amount of computations required to solve for the M roots of the characteristic equation is tremendous, and a long delay is always accompanied by this process. This presents a problem particularly when the processing is intended for real time applications. In order to minimize the delay, or the time required to check the stability[‡] for each frame of data, we must have a simple algorithm by which the stability condition of the pitch synthesis filter can be conveniently detected.

3.2 Instability in Pitch Synthesis Filter

Theoretically, when the auto-correlation method is used for the formant prediction, the stability of the corresponding synthesis filter is guaranteed. The general stability condition in the formant synthesis filter can be verified theoretically by a simple counter proof [LANG(79)].

Assuming the formant inverse filter $A_f(z)$ in Eq. (2.8) has only a single root outside the unit circle, $A_f(z)$ can be written as

$$A_f(z) = B(z)(1-\gamma_o z^{-1}) \tag{3.4}$$

where $\gamma_o = r_o e^{j\theta}$, and $|\gamma_o| = r_o > 1$. The output of the inverse filter is then equivalent to the input signal filtered by a cascade of the minimum phase filter

[†] When the lag M is positioned with the first tap.

[‡] By conventional method, this is equivalent to solving for the $(M+m-1)$ roots of the system and then examine their distribution with respect to the unit circle.

$B_f(z)$ (i.e., all the roots are inside the unit circle) producing an intermediate output c_n, and then by the unstable first order filter $(1-\gamma_o z^{-1})$. Expressing the residual in terms of this intermediate output c_n,

$$e_n = c_n - \gamma_o c_{n-1} \tag{3.5}$$

Lang and McClellan showed that if the residual energy is minimized using auto-correlation method, r_o (the magnitude of the largest root) has to be less than or equal to unity, which shows that $H_f(z) = \frac{1}{A_f(z)}$ is stable. The above proof of the general stability condition in the formant synthesis filter is detailed in Appendix A.

Unlike the formant synthesis filter, the pitch synthesis filter is rather fragile in that its stability can not always be guaranteed. The system function of the pitch predictor in Eq. (2.10) suggests the mechanisms whereby the pitch component of the speech is extracted or predicted from the input signals. Given a set of optimal pitch predictor coefficients β_i, $i = 1,2,...,m$, the m^{th}–order pitch synthesizer generates the current sample from a linear sum of the previously constructed m samples, each scaled by the pitch predictor coefficients β_i, $i = 1,2,3,...,m$, and all delayed roughly by M samples. As to be explained shortly, this big separation between the two quantities — the predicted sample and the series of samples upon which the prediction is based — constitutes a major cause, although indirect, of the frequent instability encountered in the pitch synthesis filter.

Using a one-tap pitch filter for analysis, the current sample s_n[†] in terms of the past sample can be related by

$$s_n = \beta s_{n-M} \tag{3.6.a}$$

or

$$\beta = s_n / s_{n-M} \tag{3.6.b}$$

[†] Of necessity in later discussion, subscripted notation of this type is used.

Thus, the coefficient β can be treated as a ratio between the current sample and the M^{th}-delayed sample. If a situation arises where the input (first residual or speech signal depending on the sequence used) to the pitch analysis filter is perfectly periodic, the current sample s_n can be treated as a replica of the previous sample s_{n-M}. This situation implies that the coefficient β in Eq. (3.6.b) equals one. In reality however, this situation rarely exists; thus β can assume almost any value depending on the characteristic of the current segment being analyzed.

Consider the following illustration where we have one frame of speech signal to be processed using a one-tap pitch predictor. Assuming P-F[†] sequence so that the input to the pitch predictor is the speech signal itself, the optimal coefficient (to be derived in detail later) is defined as:

$$\beta = \frac{\sum_k s_k s_{k-M}}{\sum_k s_{k-M}^2} \tag{3.7}$$

The value of the coefficient β, according to Eq. (3.7), depends on the relative magnitudes of the different correlation terms with lag M[‡]. Imagine, for argument, that there is only one sample per frame so that β can simply be treated as a ratio of the current sample to the M^{th}-delayed sample as in Eq. (3.6.b). In this situation, if these two samples are at the opposite sides of the junction between low- and high-energy as in the data segment sketched in Fig. 3.1, this ratio or the coefficient β would exceed unity, causing the recursive pitch synthesizer to be unstable. For higher order filters, the optimal coefficients can not be analyzed in the same manner, but the basic principle still remains the same. Experimental

---

[†] Pitch analysis followed by spectral (formant) analysis.

[‡] Where M is assumed optimal.

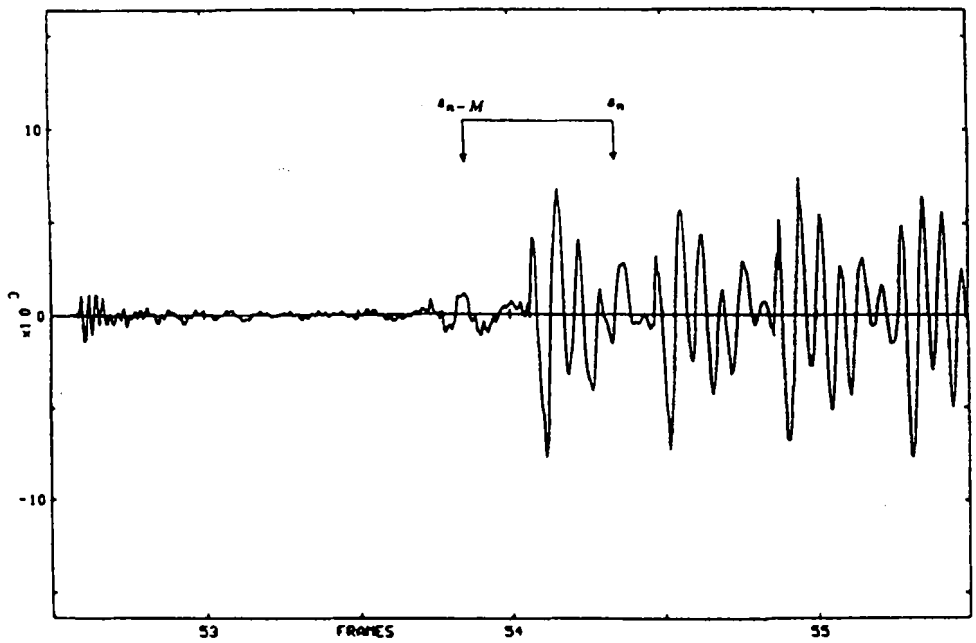


Fig. 3.1 Speech segment prone to instability.

results in later sections serve as verification of this statement.

The key point suggested here is that the cause of the frequent unstable frames in the pitch synthesis filter can be attributed partly to the inherent long lag (M) in the filter structure. When the prediction of a sample is based on past samples with such a long delay, the correlation between the current sample and those samples upon which the prediction is based is usually low. The data segment in Fig. 3.1 is a typical example in which there is a big energy differential within a frame. Statistics shows that instability tends to occur in segments of speech where there is a sudden increase of energy level, e.g., at the onset of voicing.

3.3 Single Tap System

Equivalent characteristic polynomial: $D(z) = z^M - \beta = 0$

The single tap filter is an exceptional case since the poles of the system[†] can be easily determined from the predictor coefficients β. The equivalent characteristic polynomial reveals a simple direct relationship between the only coefficient β and the common magnitude shared by the poles. To evaluate the roots of the system, we solve the characteristic equation for z and obtain

$$z = |\beta|^{\frac{1}{M}} \exp(j2\pi k/M), \qquad k = 1, 2, 3, ..., M. \tag{3.8}$$

Eq. (3.8) shows that the system consists of M poles distributed symmetrically about the origin with common magnitude $|\beta|^{\frac{1}{M}}$, and separated evenly from one another in angular frequency by $\frac{2\pi}{M}$ radians. The system stability is absolutely guaranteed if all the poles are inside the unit circle. This requires $|\beta|^{\frac{1}{M}}$ (or simply $|\beta|$) to be less than unity. Taking the poles that lie on the unit circle as the marginal stability condition, the criterion for absolute stability in one-tap filter is simply

$$|\beta| < 1. \tag{3.9}$$

Therefore, to detect the stability of 1-tap pitch synthesis filter, we just have to check whether or not the magnitude of the coefficient β is less than unity.

3.4 Multiple Tap System

In a multiple tap filter, i.e., filter with more than one tap, the poles no longer have a common magnitude. This leads to a lack of direct relationship between the predictor coefficients β_i and the pole magnitudes. Hence the pole locations, and consequently the stability status, can not simply be determined from the given set of optimal coefficients β_i.

[†] Pitch synthesis filter.

3.4.1 Stability Criteria

In the following, we attempt to find a possible relation between β_i and the magnitudes of the poles. The procedure is equivalent to finding the best combination of the $i^{th}-$dimensional vector $\vec{\beta}_i$, $(\beta_1, \beta_2, .., \beta_i)$, where $\vec{\beta}_i$ represents the set of predictor coefficients for an $i^{th}-$order filter, that corresponds to marginal stability; or equivalently that would yield poles with maximum magnitude equal to unity. In the discussion, we use the 2-tap and the 3-tap filters to represent the multiple-tap system. A graphical representation of $\vec{\beta}_i$ for $i > 3$ is impossible and the analysis becomes too complicated. But if the testing algorithm can be generalized mathematically in terms of $\vec{\beta}_i$, the results of the analysis should also be applicable to filter with any number of taps.

Consider a 3-tap filter with

Characteristic polynomial: $D(z) = z^{M+1} - \beta_1 z^2 - \beta_2 z - \beta_3 = 0.$

Unlike in the previous case of single tap filter, the roots of this polynomial can not in general be determined directly from the coefficients. As M is a relatively large number, using the direct method to solve for the roots requires a large amount of computations. Our initial approach to searching for the sought after relationship proceeds by using the inverse z-transform method to convert the system function $H_p(z) = 1/A_p(z)$ to its corresponding impulse response function h_n.

By long division, and where a_{ij} represent the coefficients, the impulse response of this filter can be shown to be in the form:

$$
\begin{aligned}
h_n = {} & \delta_n \\
& + a_{11}(\delta_{n-M+1}) + a_{12}(\delta_{n-M}) + a_{13}(\delta_{n-M-1}) \\
& + a_{21}(\delta_{n-2M+2}) + a_{22}(\delta_{n-2M+1}) + a_{23}(\delta_{n-2M}) + \\
& \quad a_{24}(\delta_{n-2M-1}) + a_{25}(\delta_{n-2M-2}) \\
& + a_{31}(\delta_{n-3M+3}) + a_{32}(\delta_{n-3M+2}) + a_{33}(\delta_{n-3M+1}) + a_{34}(\delta_{n-3M}) + \\
& \quad a_{35}(\delta_{n-3M-1}) + a_{36}(\delta_{n-3M-2}) + a_{37}(\delta_{n-3M-3}) + \ldots\ldots
\end{aligned}
\tag{3.10}
$$

According to Eq. (3.10), the impulse response consists of a series of clusters in which the envelope of the response depends on the values of the coefficients β_i. In Fig. 3.2, we show the typical characteristics of the impulse response h_n in reaction to (a)-high and (b)-low coefficient values using M=20. In Figs. 3.2(a) where $\beta_1 = \beta_2 = \beta_3 = 0.5$, the impulse response tends to grow rapidly, leading to unstable filter; but in Fig. 3.2(b) where $\beta_1 = \beta_2 = \beta_3 = 0.3$, h_n stays bounded even as $n \rightarrow \infty$.

We note that each coefficient a_{ij} in Eq. (3.8) can be expressed in terms of the predictor coefficients β_i as:

$a_{11} = \beta_1$	$a_{21} = \beta_1^2$	$a_{31} = \beta_1^3$	etc.
$a_{12} = \beta_2$	$a_{22} = 2\beta_1\beta_2$	$a_{32} = 3\beta_1^2\beta_2$	
$a_{13} = \beta_3$	$a_{23} = 2\beta_1\beta_3 + \beta_2^2$	$a_{33} = 3(\beta_1^2\beta_3 + \beta_1\beta_2^2)$	
	$a_{24} = 2\beta_2\beta_3$	$a_{34} = 6\beta_1\beta_2\beta_3 + \beta_2^3$	
	$a_{25} = \beta_3^2$	$a_{35} = 3(\beta_1\beta_3^2 + \beta_2^2\beta_3)$	
		$a_{36} = 3\beta_2\beta_3^2$	
		$a_{37} = \beta_3^3$	

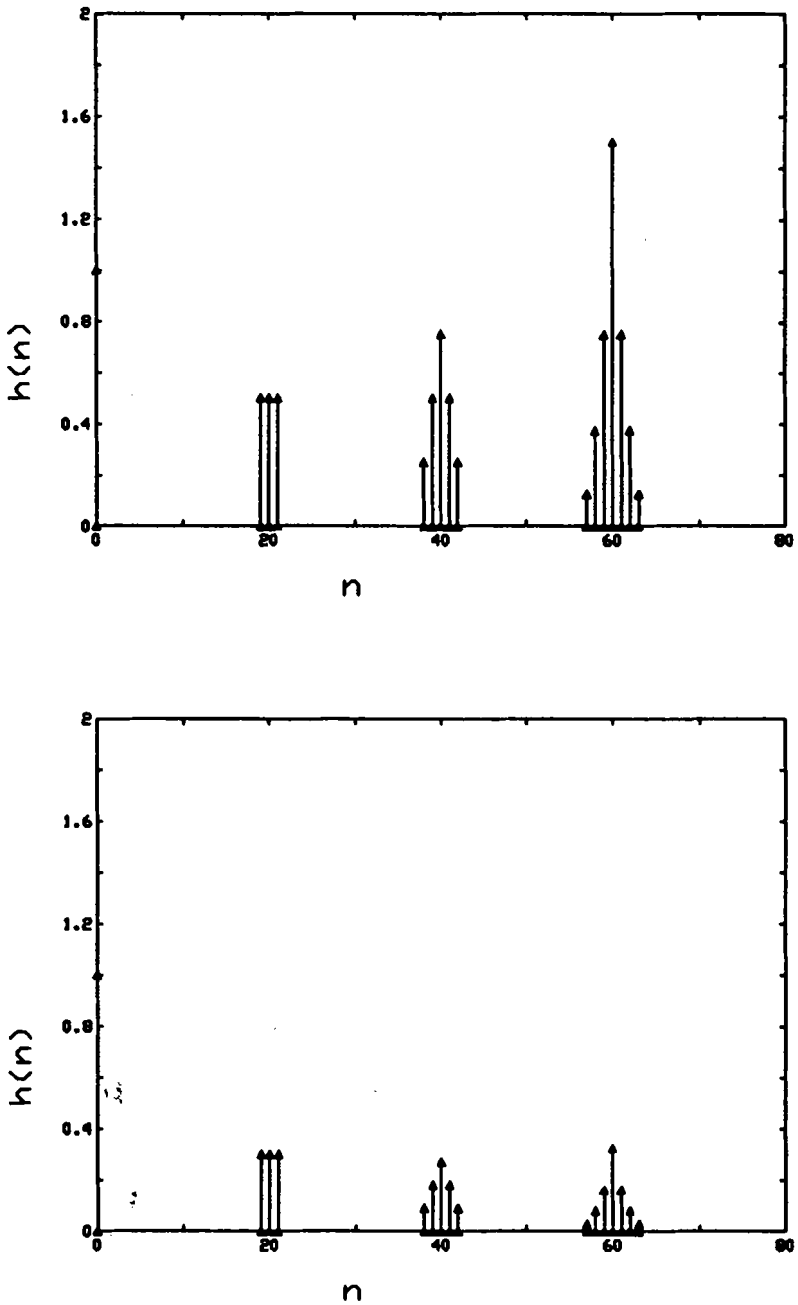


Fig. 3.2 **(a)** Growing impulse response due to large coefficients (upper figure); **(b)** Well-behaved impulse response due to small coefficients.

From the illustration in Figs. 3.2, the aggregate sum of each cluster $\sum_j a_{ij}$ can be expressed in terms of the predictor coefficients β_i so that

$$
\begin{aligned}
G_1 &= \sum_{j=1}^{3} a_{1j} = \beta_1 + \beta_2 + \beta_3 \\
G_2 &= \sum_{j=1}^{5} a_{2j} = (\beta_1 + \beta_2 + \beta_3)^2 \\
G_3 &= \sum_{j=1}^{7} a_{3j} = (\beta_1 + \beta_2 + \beta_3)^3 \\
&\cdot \\
&\cdot \\
&\cdot \\
G_i &= \sum_{j=1}^{2i+1} a_{ij} = (\beta_1 + \beta_2 + \beta_3)^i
\end{aligned}
\tag{3.11}
$$

Eq. (3.11) suggests a correlation between the characteristic of the impulse response and a quantity involving the sum of the coefficients.

The the arithmetic sum of the coefficients

$$\sum_i \beta_i < 1 \tag{3.12}$$

have been suggested in the past as a convenient check on the stability of the filter. But our study shows that this is much too lenient criterion for testing the stability, and is not a sufficient condition for stability.

Based on the following analysis and empirical observations, we establish a somewhat similar but fundamentally different criterion to determine the stability. For simplicity, a 2-tap filter is selected for illustration. The characteristic equation for the 2-tap system is

$$A_p(z) = 1 - \beta_1 z^{-M} - \beta_2 z^{-(M+1)} = 0, \tag{3.13}$$

which implies

$$z^{M+1} - \beta_1 z - \beta_2 = 0 \tag{3.14}$$

or

$$z^{M+1} = \beta_1 z + \beta_2. \tag{3.15}$$

For the system function $H_p(z) = \frac{1}{A_p(z)}$ to be stable, the roots of $A_p(z)$ must be inside a unit circle; i.e., $z \leq e^{j\theta}$.

According to the maximum modulus theorem, we can replace the variable z in Eq. (3.15) by $e^{j\theta}$, so that the quantities on both sides of the equation (now functions of θ) become

$$F_1(\theta) = F_2(\theta) \tag{3.16.a}$$

$$e^{j\theta(M+1)} = \beta_1 e^{j\theta} + \beta_2 \tag{3.16.b}$$

$F_1(\theta)$ traces a unit circle, while $F_2(\theta)$ consists of a vector of magnitude β_1 rotating about a scalar β_2. If we assume that $|\beta_2| > |\beta_1|$†, and that β_1, β_2 are of the *same* signs, $F_2(\theta) = \beta_1 e^{j\theta} + \beta_2$ can be represented graphically as in Fig.A.

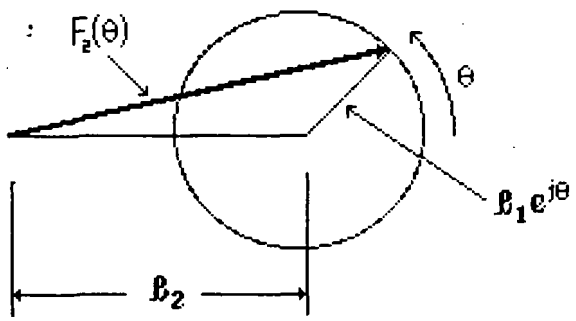


Fig. A Graphical representation of $F_2(\theta)$.

---

† Could be the other way around.

At any angle θ, the magnitude of $F_1(\theta)$ is equal to unity and it represents a unit circle. It can be shown that in this case were β_1 and β_2 have the same signs, $F_2(\theta)$ traces an ellipse shown in Fig. 3.3(a), where $|\beta_1 + \beta_2| > |\beta_2 - \beta_1|$. On the other hand, when β_1 and β_2 have *opposite* signs, β_1 and β_2 are facing each other at $\theta = 0$, which makes $F_2(\theta)$ minimum at that angle. The resulting ellipse in this case with respect to $F_1(\theta)$ is shown in Fig. 3.3(b), where $|\beta_2 - \beta_1| > |\beta_1 + \beta_2|$.

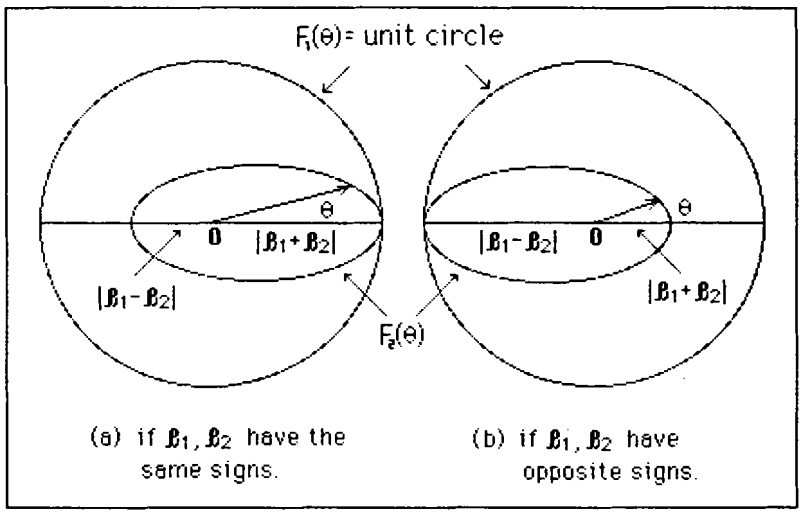


Fig. 3.3 Graphical representation of Eqs. (3.16) when β_1, β_2 are of **(a)** the same signs; **(b)** opposite signs.

Thus, Fig. 3.3 suggests that the following two conditions must be satisfied

$$|\beta_1 + \beta_2| < 1 \qquad (3.17.a)$$

$$|\beta_1 - \beta_2| < 1, \qquad (3.17.b)$$

which can be shown to be equivalent to

$$|\beta_1| + |\beta_2| < 1. \qquad (3.18)$$

Appendix B provides a slightly different approach to the derivation of Eq. (3.18). When similar approaches are applied to the 3-tap filter, the constraint to be sat-

isfied for stability is

$$|\beta_1| + |\beta_2| + |\beta_3| < 1 \tag{3.19}$$

or in general,

$$\sum_i |\beta_i| < 1 \tag{3.20}$$

We have thus come to the conclusion that it is the sum of the magnitude, rather than the arithmetic sum of the coefficients as in Eq. (3.12) that forms the critical quantity which should be confined to unity value for filter stability.

3.4.2 Stability Regions

In this section, we derive numerically the actual stability regions of $H(z)$ for $m = 2, 3$ with M as a parameter. The purpose is to use this derived stability region to establish the necessary condition for the stability of the pitch synthesis filter, and also to compare the tightness of the regions defined by the two criteria [Eq. (3.12) and Eq. (3.20)] against the actual stability region.

Consider a second order pitch synthesis filter with system function

$$H_p(z) = \frac{1}{1 - \beta_1 z^{-M} - \beta_2 z^{-(M+1)}}. \tag{3.21}$$

For a given lag M and (β_1, β_2), the system has $(M + 1)$ multiple zeros at the origin and $(M + 1)$ poles distributed around the origin, some of which may be outside the unit circle. To locate the boundary of the stability region, the characteristic equation is expressed as a function of β_1 and β_2

$$F(\beta_1, \beta_2) = z^{M+1} - \beta_1 z - \beta_2 = 0. \tag{3.22}$$

$F(\beta_1, \beta_2)$ has $(M + 1)$ roots whose distribution depends on the values of both β_1 and β_2. We are interested in locating all $\{\beta_1, \beta_2\}$ which give rise to

$(M + 1)$ poles and whose maximum magnitudes are equal to unity. In other words, for a given β_1, we search for a value β_2 which will give poles with maximum magnitudes of unit length. Confining the coefficient values to the square region $|\beta_i| \leq 1,\ i = 1,2$, we search for all $\{\beta_1, \beta_2\}$ which satisfy the above stated condition.

An interesting phenomenon is observed on the true stability region of a 2-tap pitch synthesis filter. We noticed that for small M values, the upper bound of $\{\beta_1, \beta_2\}$ corresponding to a marginal stability for 2-tap filter are as shown in Fig. 3.4. As the lag M is incremented gradually from a small value of 2, the axis of symmetry of the stability region alternates between (1)-the β_1-axis when M is even, and (2)-the β_2-axis when M is odd. As M increases, the curvature part of the region quickly fades away, and seems to have disappeared as soon as M reaches 10.

The range of M considered in our present analysis is between 20 and 120. With such large M values and from the converging tendency illustrated in Fig. 3.4 as M increases, it is save to assume that the $\{\beta_1, \beta_2\}$ combinations which correspond to stability, are bounded by a diamond, mathematically described by

$$|\beta_1| + |\beta_2| < 1. \tag{3.23}$$

For purpose of convenience in the later analysis, let us name the criterion defined in Eq. (3.12) as crit(SUM) and the criterion in Eq. (3.20) as crit(SOM). Crit(SUM) checks the arithmetic SUM of the coefficients, whereas crit(SOM) checks the Sum Of the Magnitude of the coefficients against unity. In a 2-tap filter, the boundary of crit(SUM) is simply a straight line passing through points (β_1, β_2)=(0,1) and (β_1, β_2)=(1,0) but crit(SOM) defines a diamond with vertices at (β_1, β_2)=(0,$\pm$1), (β_1, β_2)=($\pm$1, 0). It is obvious that the constraint imposed

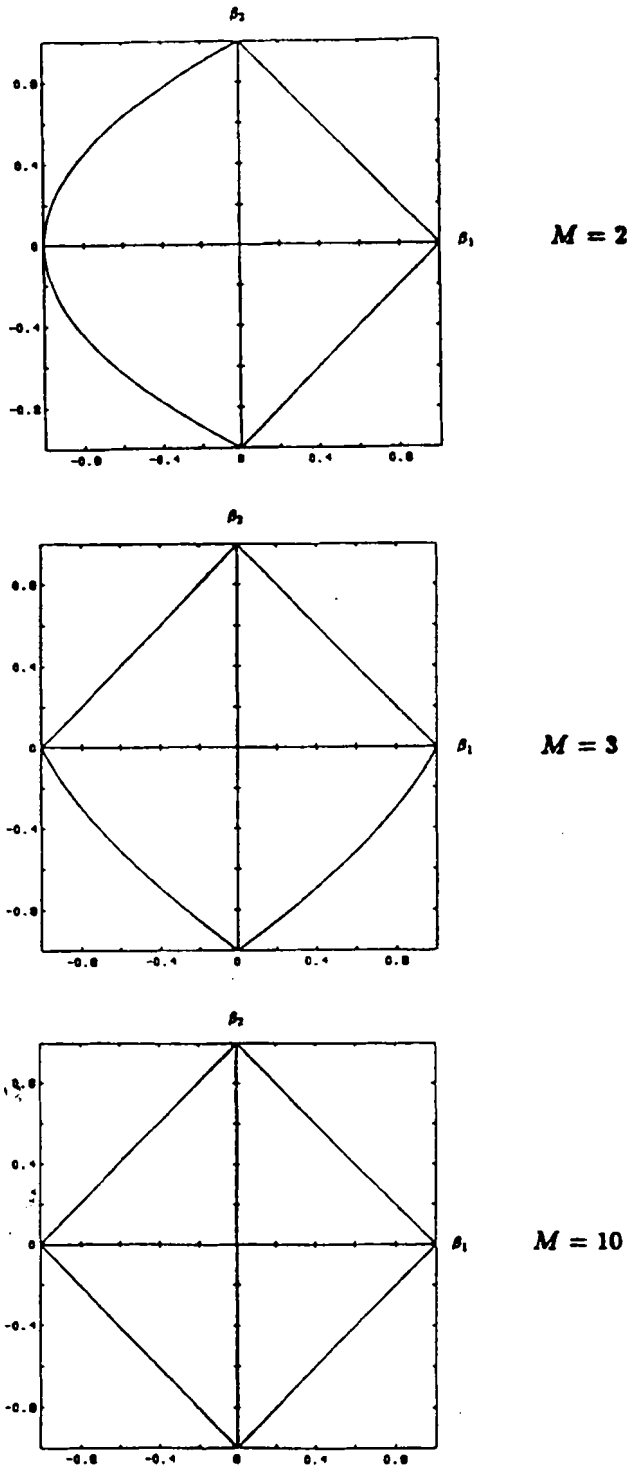


Fig. 3.4 The upper bound of 2-tap coefficients corresponding to marginal stability for M=2,3,10.

by crit(SUM) is much more lenient than the one defined by crit(SOM). The latter requires the fitting of the sum of magnitude (SOM) of all coefficients into a unit length, which implies that as the order increases, a larger number of coefficients must be jammed into a unit length, thus putting a tighter upper limit on each coefficient value. On the other hand, crit(SUM) literally does not have any limit on the coefficient values when cancellation between coefficients with opposite signs occurs.

In determining the stability region for the 3-tap pitch synthesis filter, a similar procedure as described above is used. In the present case, the characteristic equation is a function of $\{\beta_1, \beta_2, \beta_3\}$

$$F(\beta_1, \beta_2, \beta_3) = z^{M+1} - \beta_1 z^2 - \beta_2 z - \beta_3 = 0 \tag{3.24}$$

Again, restricting the coefficients to values within the cube $|\beta_i| \leq 1$, $i = 1, 2, 3$, we locate all $\{\beta_1, \beta_2, \beta_3\}$ which satisfy Eq. (3.24) and which also give maximum poles on the unit circle.

The pictorial stability region for 2-tap filter in Fig. 3.4 suggests that the corresponding 3-tap version for large M value is likely to take the general shape of the figure indicated in Fig. 3.5, which incidently is the crit(SOM)-defined region, made up of an upper pyramid ($\beta_3 > 0$) and a lower inverted pyramid ($\beta_3 < 0$). The common base ($\beta_3 = 0$) of the two pyramids is essentially the 2-tap stability (diamond) region at high M, when the region has achieved a steady status. For low M values however, there is a one-order mismatch between the 2-tap region and the 3-tap region on the $\beta_1 - \beta_2$ plane where $\beta_3 = 0$. To see this, compare Eq. (3.22) with Eq. (3.24) setting $\beta_3 = 0$.

When numerically locating the actual stability region (upper), and as the parameter M is gradually incremented, we observe a dramatic transformation

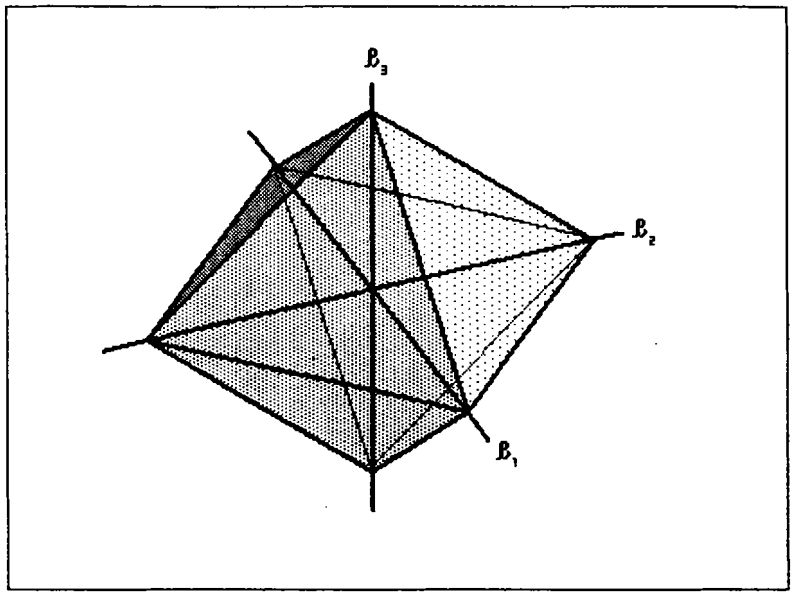
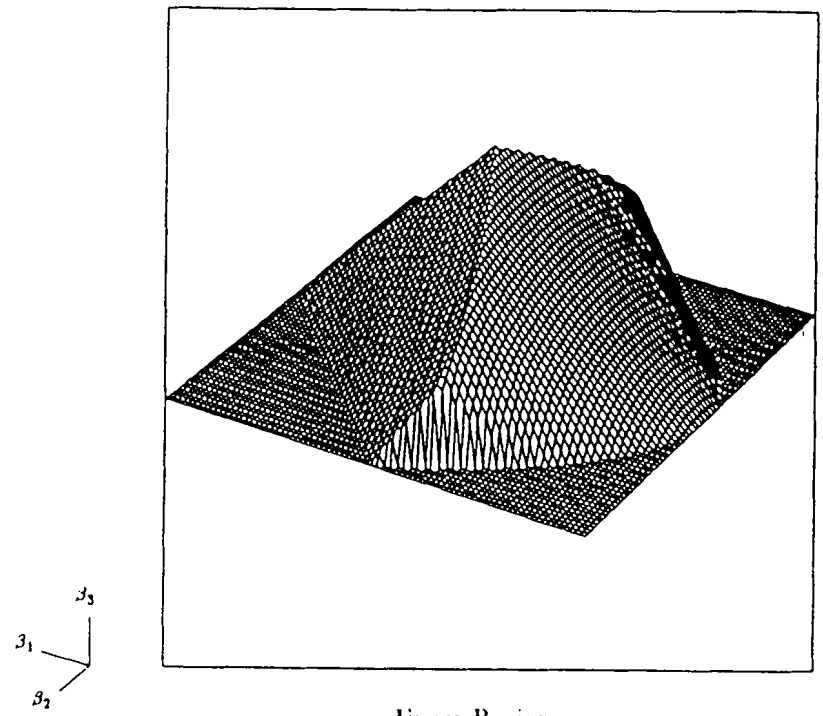


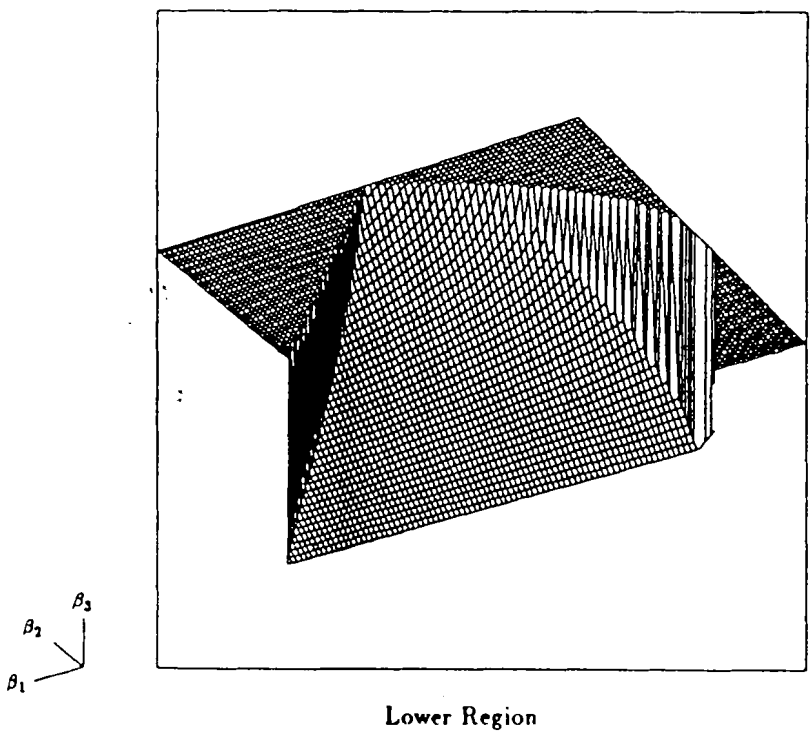
Fig. 3.5 Stability region for 3-tap pitch synthesis filter according to crit(SOM).

of the boundary of the stability region. At the lowest possible lag M=3, one half of the region where $\beta_1 > 0$[†] is very close to the speculated region shown in Fig. 3.5, but the other half of the region where $\beta_1 < 0$ has a bulging shape as shown in Fig. 3.6. As M is slowly incremented, the section which resembles the perfect pyramid remains more or less unchanged while the bulging part of the region undergoes an erratic transformation. Appendix C contains several plots of the stability regions for M=4, 5, 6, 7. Just as in the 2-tap case, in the process of transformation as M increases, the stability region displays a symmetrical property when M is odd, but lacks a symmetry when M is even. But as soon as M exceeds 7, the shape transformation process decelerates quickly and suddenly freezes. Further analysis using increasing values of M indicates that the total change in the shape of the stability region accumulated from M=7 to M=23 is insignificant. In Fig. 3.7, we compare the stability regions for M=7, 23. As

[†] Note that the positive axis of β_1 in Fig. 3.6 points to the west.



Upper Region



Lower Region

Fig. 3.6 Numerically derived stability region at M=3, upper and lower region

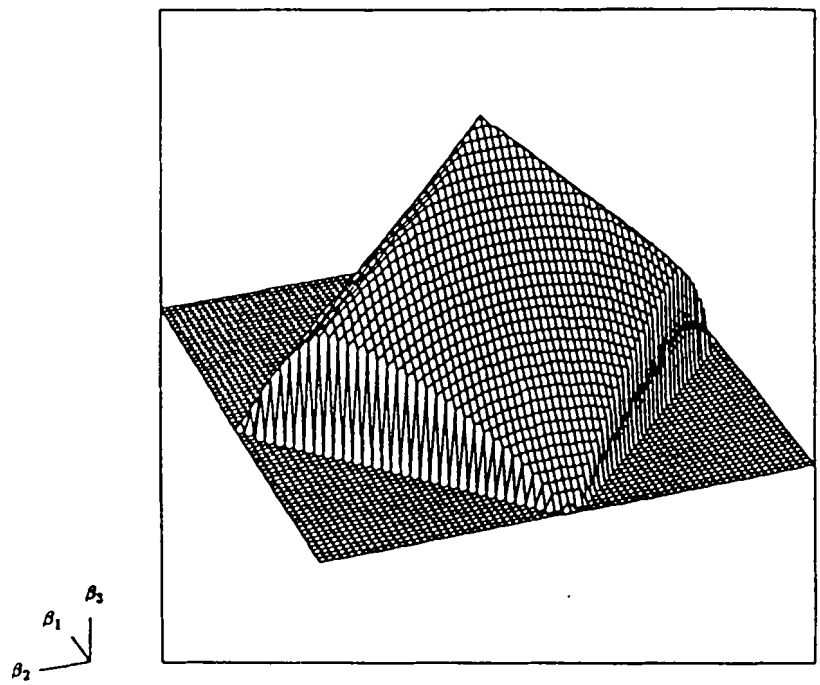
illustrated in Appendix C, the upper region where $\beta_1 > 0$ even when M=7 is already very close to a perfect pyramid, or to those depicted in Fig.3.5.

Owing to the limitation of the subroutines we used in solving for the roots, we experienced some difficulties when attempting to use values of M over 23. Only when using smaller selective range of $(\beta_1, \beta_2)^\dagger$ which reduced the load of computation that we managed to continue the test up to M=40. But even at this magnitude of M, we hardly detected any localized change from the increased M values. These test results (on the localized stability regions) for M=23 up to M=40 further confirm that the stability region has indeed frozen at a value of M as low as 23. From the observation that there is no further change in the shape of the stability region for higher value of M, we assume that the stability region has reached a steady state at M=23. Hence for M within the range of interest (20 : 120), the stability region in Fig. 3.7 can be assumed to be the upper bound for β_i in a 3-tap filter (upper region).

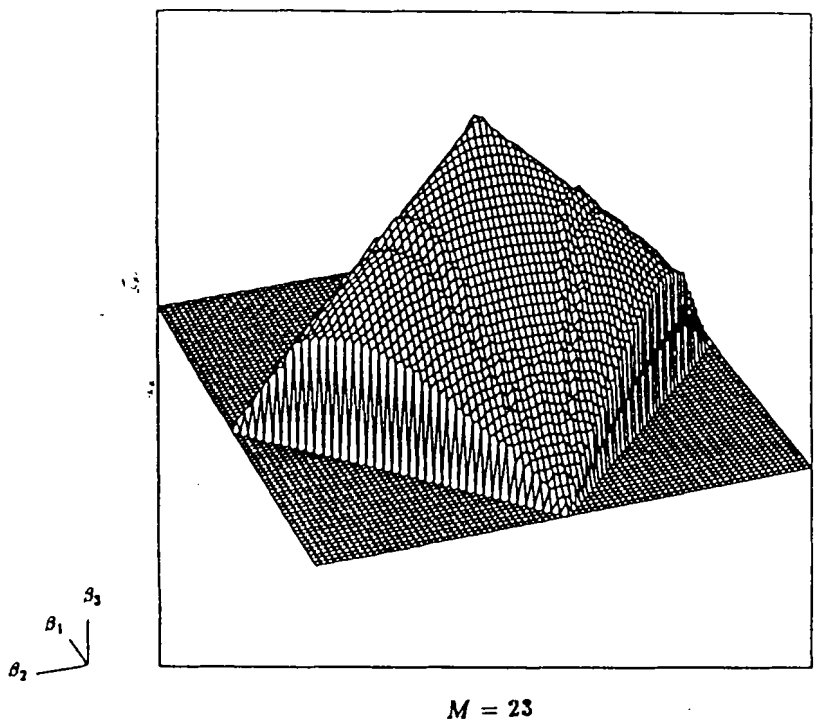
When the same procedure is applied to the lower stability region ($\beta_3 < 0$), the results show an interesting symmetry between the upper and the lower stability regions: the lower stability region is observed to be the inverted image of the upper region followed by a 180°– rotation. The upper and lower regions are depicted in parallel in Fig. 3.8 to show their symmetry. Note that this symmetry is strictly true only for high values of M. At low M, there is no clear symmetry between the upper and the lower regions, as illustrated for example in Fig. 3.6 for M=3.

---

† restricting β_1 to region near -1 where the bulging occurs in region $\beta_3 > 0$.



$M = 7$



$M = 23$

Fig. 3.7 Comparison of stability region when (M=7 and M=23)

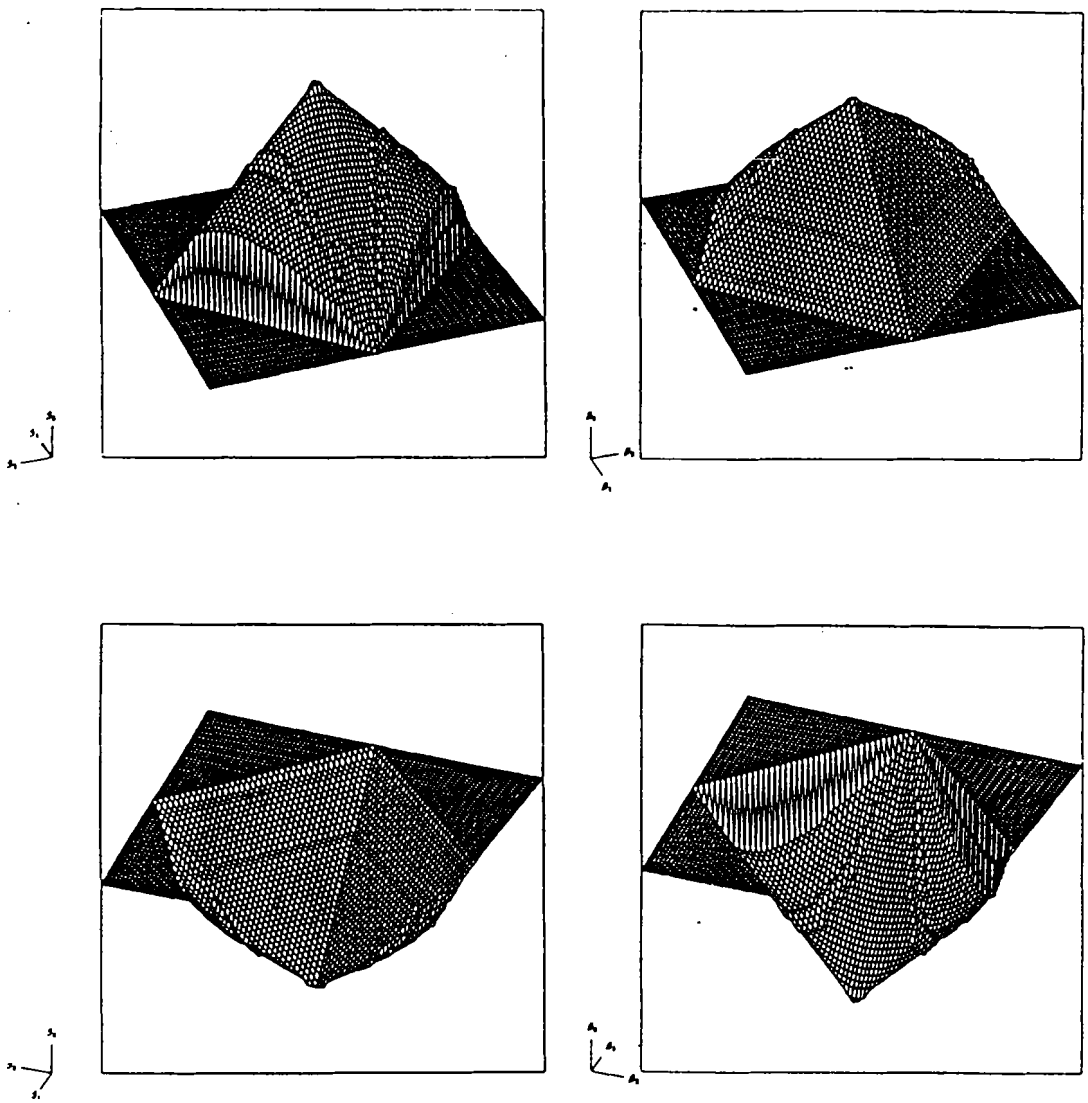


Fig. 3.8 Complete view of the true stability regions for 3-tap pitch synthesizer at M=23: (a) upper region, (b) lower region

3.4.3 Space Division

From the above analysis, we note that the 3-tap filter stability region for large M is made up partly of 'perfect' pyramids where $\beta_1\beta_3 > 0$, and partly of distorted region where $\beta_1\beta_3 < 0$. In many of the later discussions concerning the 3-tap stability region, we expect to encounter many occasions in which the 8 divisions in space need to be referenced. To facilitate the discussion, the 3-dimensional space is divided into 8 different regions as shown in Fig. 3.9, with $\beta_1, \beta_2, \beta_3$ forming the 3-axes. In anti-clockwise direction, the upper four compartments starting from regions where all $\beta_i > 0$ are named regions I, II, III, IV, while the corresponding lower four compartments form regions V, VI, VII, VIII.

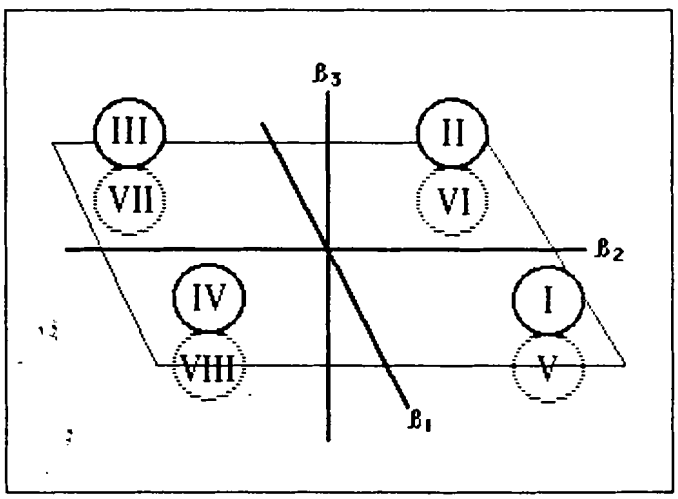


Fig. 3.9 Division of 3-dimensional (3-tap) stability region.

Matching Fig. 3.8 to the convention in Fig. 3.9, the stability regions that resemble the perfect pyramids now correspond to divisions I, IV, VI, and VII, while divisions II, III, V, VIII correspond to the regions with the bulging shapes.

3.5 Necessary and Sufficient Conditions

The established crit(SOM) in Eq. (3.20) states that a stability exists if the SOM of all coefficients is less than unity. We know from previous discussion that for multi-tap filters, some of the true stability regions lie outside the regions defined by crit(SOM). Accordingly, those stable coefficients that lie between the crit(SOM)-defined region and the true stability region are considered unstable. As shown earlier and will be further verified later, the 2-tap filter has an actual stability region very close to a diamond for $M > 7$. In this case, crit(SOM) represents the necessary condition for stability in 2-tap filter without losing much accuracy. For the 3-tap filter however, the actual stability region is not well represented by crit(SOM), particularly in regions II, III, V and VIII, where there is a mismatch between the crit(SOM)-defined region and the true stability region. The degree of mismatch between the two regions seems to be increasingly larger as the order increases, implying that the representation of the true stability region by crit(SOM) becomes less accurate as the order of the filter increases.

Due to the irregularity of the shape of the actual stability region, it is difficult to express it with a generalized mathematical function. We therefore consider using crit(SOM) as a convenient model for the sufficient condition of stability, as it can be easily generalized to any order. The minimum requirement for using crit(SOM) as sufficient condition is to ensure that the crit(SOM)-defined regions be enclosed by the actual stability regions. In the single-tap filter, the direct relationship between β and the pole magnitudes of the system implies that crit(SOM)-defined region exactly overlaps the actual stability region. In 2-tap filter, it is also obvious, at least visually from Fig. 3.4, that the square stability region defined by crit(SOM) is always equal to or smaller than the true stability region. In order to determine the validity in the case of 3-tap filter, we creat a

difference function $\Delta\beta_3$ by taking the difference between the true stability region and the region defined by crit(SOM), i.e.,

$$\Delta\beta_3 = |actual\ region| - |(1 - |\beta_1| - |\beta_2| - |\beta_3|)| \tag{3.25}$$

According to Eq. (3.25), crit(SOM) is guaranteed to be inside the actual region if only if $\Delta\beta_3 \geq 0$. Again by the symmetry exhibited in the stability region at high M, it is sufficient to study $\Delta\beta_3$ for the upper region alone, and the the result is assumed to be applicable to the lower region.

Using the data generated by the two quantities on the right side of Eq. (3.25), the difference function $\Delta\beta_3$ is generated and plotted in Fig. 3.10. The figure clearly indicates that $\Delta\beta_3$ is positive in divisions I, IV† and zero elsewhere. The positive $\Delta\beta_3$ marks the extent to which crit(SOM)-region lies below the actual region, while the zero difference indicates a perfect match between the two regions.

The above observation of the relationship between the true stability regions and those defined by crit(SOM) for 1-, 2- and 3-tap filters establishes the stability conditions: while the actual stability regions (Fig. 3.4 for 2-tap and Fig. 3.8 for 3-tap) define the necessary condition for stability, crit(SOM) stated in Eq. (3.18) — which have been shown to be subregions of the true regions — can serve as the sufficient conditions for stability. The deviation of the sufficient condition from the necessary condition, which leads to an over-estimation of the instability of the pitch synthesis filter, is to be determined in the following sections.

---

† Also divisions VI, VII when the lower region is considered.

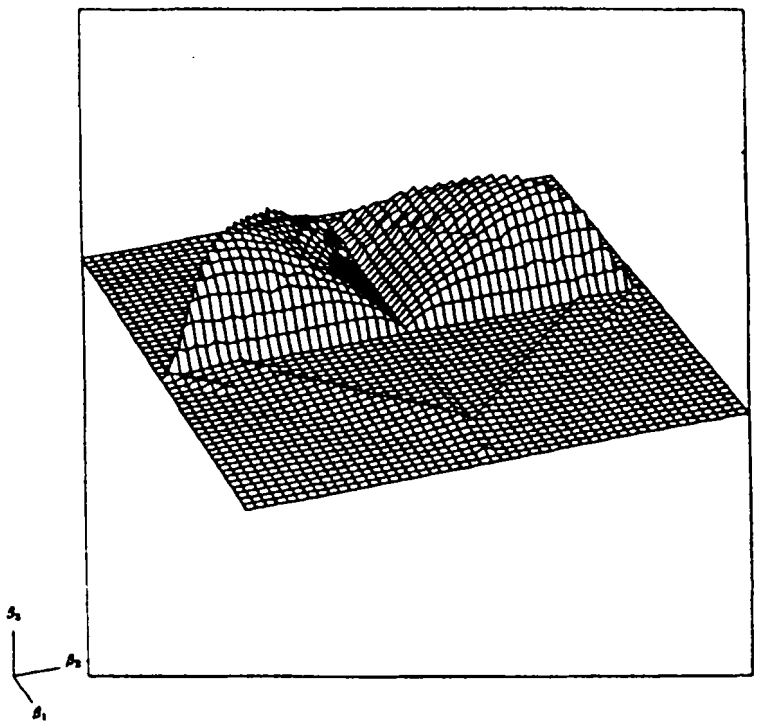


Fig. 3.10 Difference function $\Delta\beta_3$.

3.6 Application of Jury's Critical Stability Criterion

Several algorithms exist for testing the stability of linear discrete-time system. One of them, which we apply in this section, originates from Schur-Cohn criterion. Its original form is applicable to a system with the following characteristic function polynomial

$$F(z) = a_0 + a_1 z + a_2 z^2 + ... + a_{n-1} z^{n-1} + a_n z^n, \tag{3.26}$$

where the coefficients a_i can be complex.

The test for stability by the original criterion is quite laborious. For an n^{th}-order system, the criterion requires n evaluations of $2k$-order determinant for

$k = 1, 2, ..., n$ [JURY(64)], where the determinant is shown in Appendix D.

Based on the Schur-Cohn criterion, Jury develops a new critical stability constraints which is somewhat similar to the Hurwitz-Routh or Lièнard-Chipart method for the continuous case. The simplified form, however, is applicable only when the coefficients a_i in Eq. (3.26) are real, and it is summarized in the following.

For a system with characteristic function in Eq. (3.26) to be stable, the following three constraints must be satisfied.

(1) $F(1) > 0$
(2) $(-1)^n F(-1) < 0$
(3) $(-1)^{\frac{(k+1)k}{2}}(A_k - B_k) > 0$, where $A_k - B_k$ [†]$= |X_k - Y_k|$.

The matrices X_k and Y_k are in Appendix E. For n-odd, $k = 1, 3, 5, ..., n-1$: for n-even, $k = 2, 4, 6, ..., n-1$. The system must be tested successfully in sequence for all values of k for the n^{th}−order system to be stable.

In this section, we shall make use of the above three constraints to achieve two goals. One is to verify the validity of the true stability region, which was numerically derived in Section 3.4.2 for the 2-tap and 3-tap filters; another is to build a mathematical model which will approximate the actual stability region. These models will then be used to judge the effectiveness and the reliability of using the SOM-criterion as stability testing criterion.

3.6.1 Two Tap Filter

The 2-tap pitch synthesis filter in terms of lag M has the following charac-

[†] Refer to [JURY(64)] pp.89–90 for detail.

teristic equation

$$F(z) = z^{M+1} - \beta_1 z - \beta_2 \tag{3.27}$$

Comparing Eq. (3.27) with Eq. (3.26), we obtain

$$\begin{aligned} n &= M+1 \\ a_0 &= -\beta_2 \\ a_1 &= -\beta_1 \\ a_n &= 1 \\ a_i &= 0 \quad for \quad i \neq 0, 1, n. \end{aligned} \tag{3.28}$$

Substituting the above values and absorbing the minus signs of the entries in the determinants in constraint (3), the three constraints to be satisfied by the 2-tap filter for stability are:

$\underline{(for : M - odd, n - even)}$

(1) $1 - \beta_1 - \beta_2 > 0$

(2) $1 + \beta_1 - \beta_2 > 0$ and

(3) (M x M) determinant =

$$(-1)^{\frac{M(M+1)}{2}+1} \begin{vmatrix} \beta_2 & \beta_1 & 0 & 0 & \dots & \dots & 0 & 0 & 1 \\ 0 & \beta_2 & \beta_1 & 0 & \dots & \dots & 0 & 1 & 0 \\ 0 & 0 & \beta_2 & \beta_1 & \dots & \dots & 1 & 0 & 0 \\ . & . & . & . & . & . & . & . & . \\ . & . & . & \beta_2 & \beta_1 & 1 & . & . & . \\ 0 & 0 & 0 & 0 & \beta_2+1 & \beta_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & \beta_2 & \beta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & \beta_2 & \beta_1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & \beta_2 & \beta_1 \\ 1 & 0 & 0 & 0 & \dots & \dots & 0 & 0 & \beta_2 \end{vmatrix} > 0$$

Or

$\underline{(for : M - even, n - odd)}$

(1) $1-\beta_1-\beta_2>0$

(2) $1-\beta_1+\beta_2>0$ and

(3) (M x M) determinant =

$$(-1)^{\frac{M(M+1)}{2}+1}\begin{vmatrix} \beta_2 & \beta_1 & 0 & 0 & \dots & \dots & 0 & 0 & 0 & 1 \\ 0 & \beta_2 & \beta_1 & 0 & \dots & \dots & 0 & 0 & 1 & 0 \\ . & . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . & . \\ . & . & . & \beta_2 & \beta_1 & . & 1 & . & . & . \\ 0 & 0 & 0 & 0 & \beta_2 & \beta_1+1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & \beta_2 & \beta_1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & \beta_2 & \beta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & \beta_2 & \beta_1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & \beta_2 & \beta_1 \\ 1 & 0 & 0 & 0 & \dots & \dots & 0 & 0 & 0 & \beta_2 \end{vmatrix} < 0$$

Constraints (1) and (2) define the two straight boundaries of the stability region shown in Fig. 3.4, where they intersect at the point $(\beta_1,\beta_2)=(0,1)$ for M-odd, and at the point $(\beta_1,\beta_2)=(1,0)$ for M-even. The third constraint, when evaluated for a given M and expressed as a function of β_2, defines M different curves. One of the curves with the smallest magnitude completes the boundary of the stability region. However, even with Jury's simplified criterion, a direct evaluation of the determinants for high M values can be quite involved. In the following, we expand the third constraint only for M =1 to 7. Using constraint (3), the expanded matrices are:

$(M=1):\beta_2+1>0$

$(M=2):\beta_2^2-\beta_1-1<0$

$(M=3):-[\beta_2^3+\beta_2^2+\beta_2+(\beta_1-1)]>0$

$(M=4):-\beta_2^4+\beta_2^2(\beta_1+2)+\beta_1^3+\beta_1^2-\beta_1-1<0$

$(M=5):\beta_2^5+\beta_2^4-2\beta_2^3+\beta_2^2(\beta_1^2-2)+\beta_2(1-\beta_1^2)+(1-2\beta_1^2+\beta_1^4)>0$

$(M=6):\beta_2^6-\beta_2^4(\beta_1-3)-\beta_2^2(\beta_1^3-2\beta_1^2+3)-\beta_1^5-\beta_1^4+2\beta_1^3+\beta_1^2-\beta_1-1<0$

$(M=7):-[\beta_2^7+\beta_2^6-3\beta_2^5+(\beta_1^2-3)\beta_2^4+(3-2\beta_1^2)\beta_2^3+(3-4\beta_1^2+\beta_1^4)\beta_2^2$

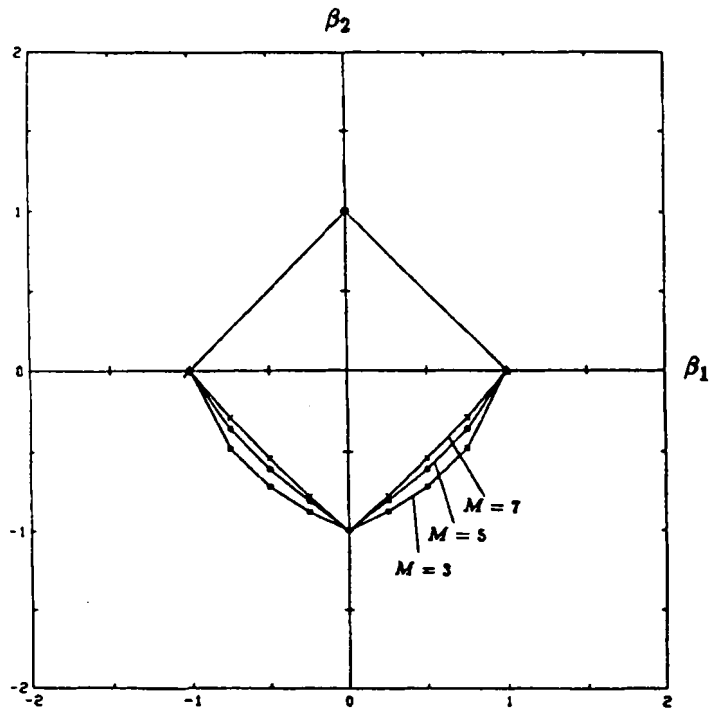


Fig. 3.11 Converging tendency in the curvature of 2-tap stability region as M steps through 3,5 and 7.

$$-(1 - 2\beta_1^2 + \beta_1^4)\beta_2 + \beta^6 - 3\beta_1^4 + 3\beta_1^2 - 1] > 0$$

The plots corresponding to the above relations† between β_1 and β_2 along with the first two constraints are provided in Appendix F. They show, as observed earlier when numerically locating the boundary, that the points of intersection alternate from the β_1-axis (at even M) to β_2-axis (at odd M). More importantly, they reveal that the degree of curvature on the boundary diminishes as the order of the polynomial or M increases. The shrinking tendency of the curvature as M steps through 3, 5 and 7 is depicted in Fig. 3.11. At M=7, the stability region boundary is approaching a straight line. Hence we conclude that as $M \rightarrow \infty$, the sufficient condition approaches the necessary condition for stability in 2-tap filter.

† Except for M=6.

3.6.2 Three Tap Filter

In a similar fashion, the Jury's critical constraints are applied to a 3-tap filter with characteristic equation:

$$F(z) = z^{M+1} - \beta_1 z^2 - \beta_2 z - \beta_3 \tag{3.29}$$

In this case, when Eq. (3.29) is compared to Eq. (3.26), the polynomial coefficients in terms of the predictor coefficients are:

$$\begin{aligned} n &= M+1 \\ a_0 &= -\beta_3 \\ a_1 &= -\beta_2 \\ a_2 &= -\beta_1 \\ a_n &= 1 \\ a_i &= 0 \quad for \quad i \neq 0,1,2,n \end{aligned}$$

Using the above values, the three critical constraints to test the stability of the 3-tap filter now become:

$\underline{(for: M-odd, n-even)}$

(1) $1 - \beta_1 - \beta_2 - \beta_3 > 0$

(2) $1 - \beta_1 + \beta_2 - \beta_3 > 0$ and

(3) (M x M) determinant =

$$(-1)^{\frac{M(M+1)}{2}+1} \begin{vmatrix} \beta_3-\beta_1 & \beta_2 & \beta_1 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 1 \\ 0 & \beta_3 & \beta_2 & \beta_1 & 0 & 0 & \dots & 0 & 0 & 1 & 0 \\ 0 & 0 & \beta_3 & \beta_2 & \beta_1 & 0 & \dots & 0 & 1 & 0 & 0 \\ . & . & . & . & . & . & . & . & . & . & . \\ . & . & . & . & \beta_3 & \beta_2 & \beta_1 & . & . & . & . \\ . & . & . & . & . & \beta_3+1 & \beta_2 & \beta_1 & . & . & . \\ 0 & 0 & 0 & 0 & 1 & 0 & \beta_3 & \beta_2 & \beta_1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & \beta_3 & \beta_2 & \beta_1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & \beta_3 & \beta_2 & \beta_1 \\ 0 & 1 & 0 & 0 & 0 & 0 & \dots & \dots & 0 & \beta_3 & \beta_2 \\ 1 & 0 & 0 & 0 & 0 & 0 & \dots & \dots & 0 & 0 & \beta_3 \end{vmatrix} > 0$$

Or

$\underline{(for : M - even, n - odd)}$

(1) $1 - \beta_1 - \beta_2 - \beta_3 > 0$

(2) $1 + \beta_1 - \beta_2 + \beta_3 < 0$ and

(3) (M x M) determinant =

$$(-1)^{\frac{M(M+1)}{2}+1} \begin{vmatrix} \beta_3 - \beta_1 & \beta_2 & \beta_1 & 0 & 0 & 0 & \dots & 0 & 0 & 0 & 0 & 1 & \\ 0 & \beta_3 & \beta_2 & \beta_1 & 0 & 0 & \dots & 0 & 0 & 0 & 1 & 0 & \\ 0 & 0 & \beta_3 & \beta_2 & \beta_1 & 0 & \dots & 0 & 0 & 1 & 0 & 0 & \\ . & . & . & . & . & . & . & . & . & . & . & . & \\ . & . & . & . & \beta_3 & \beta_2 & \beta_1 & . & . & . & . & . & \\ . & . & . & . & . & \beta_3 & \beta_2 + 1 & \beta_1 & . & . & . & . & \\ 0 & 0 & 0 & 0 & 0 & 1 & \beta_3 & \beta_2 & \beta_1 & 0 & 0 & 0 & \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & \beta_3 & \beta_2 & \beta_1 & 0 & 0 & \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & \beta_3 & \beta_2 & \beta_1 & 0 & \\ 0 & 0 & 1 & 0 & 0 & 0 & \dots & \dots & 0 & \beta_3 & \beta_2 & \beta_1 & \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & \dots & \dots & 0 & 0 & \beta_3 & \beta_2 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & \dots & \dots & 0 & 0 & 0 & \beta_3 \end{vmatrix} < 0$$

In the case of 3-tap filter, constraints (1) and (2) define two planes in space. The first constraint, which is independent of M , always define a plane in region I crossing the points $(\beta_1 = \beta_2 = \beta_3 = 1)$. The second constraint also defines a plane but it depends on the M value. If M is odd, the plane lies in region IV and crosses the points $(\beta_1 = -\beta_2 = \beta_3 = 1)$; if M is even, it lies in region VI and crosses the points $(-\beta_1 = \beta_2 = -\beta_3 = 1)$.

From Fig.3.8, we know that the true stability region has four flat surface boundaries in regions I, IV, VI and VII. Yet according to Jury's criterion, only two flat surfaces are defined by Jury's first two constraints. Therefore, we theorize that the two other planes which constitute the true stability region must have evolved from that part of the surface defined by the third constraint when $M \rightarrow \infty$, which also models the irregular bulging boundaries of the true stability region.

To overcome the complexity in evaluating the matrix in Jury's third con-

straint for 3-tap filter, we use only M=5. The (5x5) matrix

$$\begin{vmatrix} \beta_3-\beta_1 & \beta_2 & \beta_1 & 0 & 1 \\ 0 & \beta_3 & \beta_2 & \beta_1+1 & 0 \\ 0 & 0 & \beta_3+1 & \beta_2 & \beta_1 \\ 0 & 1 & 0 & \beta_3 & \beta_2 \\ 1 & 0 & 0 & 0 & \beta_3 \end{vmatrix}$$

gives the following inequality

$$\beta_3^5+\beta_3^4(1-\beta_1)-2\beta_3^3(1+\beta_1)+\beta_3^2(\beta_2^2+2\beta_1^2-2)+\beta_3(\beta_1^2+2\beta_1-\beta_2^2-4\beta_1\beta_2^2+1)$$

$$+(\beta_2^4-2\beta_2^2-\beta_1\beta_2^2-\beta_1^3-\beta_1^2+\beta_1+1)>0 \qquad (3.30)$$

Taking the equality sign and treating β_1, β_2 as independent variables, i.e., $\beta_3(\beta_1, \beta_2) = 0$ for all values of β_1, β_2 in a region defined by

$$|\beta_1| + |\beta_2| < 1,^{\dagger}$$

we obtain 5 sets of real roots, each forming a surface in space. It can be shown that the 5 sets of roots correspond to five surfaces. The part of the surface that is closest to the origin defines the 3-tap filter stability region boundary in regions II, III, V, VI, VII and VIII. We have shown earlier that the true stability region for M=7 is very close to the steady state (Fig. 3.7). Even when using M=5, Appendix C shows that the resulting stability region is still reasonably close to the steady state. Thus, along with constraints(1) and (2) which model the flat portion of the boundary in regions I and IV, Jury's simplified criterion at M=5 should serve as a fair model for the 3-tap stability region (necessary condition).

3.7 Reliability of Criteria

The two variations of criteria established in Section 3.4.1 can be used to

† solving $F(\beta_3) = 0$ outside this region will yield complex roots.

determine the stability of the pitch synthesis filter. However, neither of them is necessarily reliable. Even crit(SOM), which is more accurate than crit(SUM) in judging the stability status, is not a perfect model for the actual stability regions. As shown earlier in Fig. 3.4 and Fig. 3.8, instead of the straight and flat boundaries defined by crit(SOM), part of the *true* stability region boundaries are made up of curves (in 2-tap filter) and bulging surface (in 3-tap filter). It appears that the discrepancy between the actual and the crit(SOM) defined stability regions increases with the number of taps used by the filter.

In this section, we are to determine the reliability of using these criteria, particularly crit(SOM), in testing the actual stability condition of the pitch synthesis filter. The term reliability indicates the closeness between the stability region defined by a criterion and the actual stability region, or the tightness between the sufficient and the necessary conditions. Using the pitch analysis filter of different orders, the eight speech files in Appendix G are processed, Based on the optimal coefficients $\{\beta_i\}$ for each frame generated in the analysis, the stability status of the corresponding pitch synthesis filter is judged according to each of the two criteria. If $\{\beta_i\}$ fall within the boundary set by the specific criterion, the filter is considered stable. Subsequently, Jury's models developed in Section 3.6 are used to predict the curves which approximate the actual instability for m =2, 3.

The average percentage of unstable frames in the speech files, defined as the ratio of unstable frames over the total number of frames in each file in percentage, is used to indicate the level of instability by the criteria. The unstable frame levels as reflected by crit(SUM) and crit(SOM) are shown in Figs. 3.12. The curves of instability indicated in the figure by no means represents the true level of instability. In Fig. 3.12(a), we indicate the average of eight speech files according to the two criteria; the upper curve is produced by using crit(SOM).

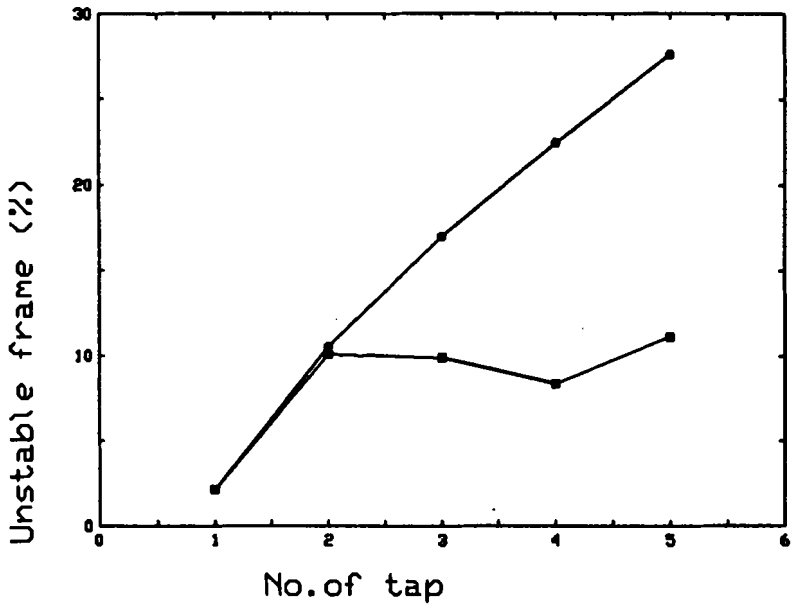
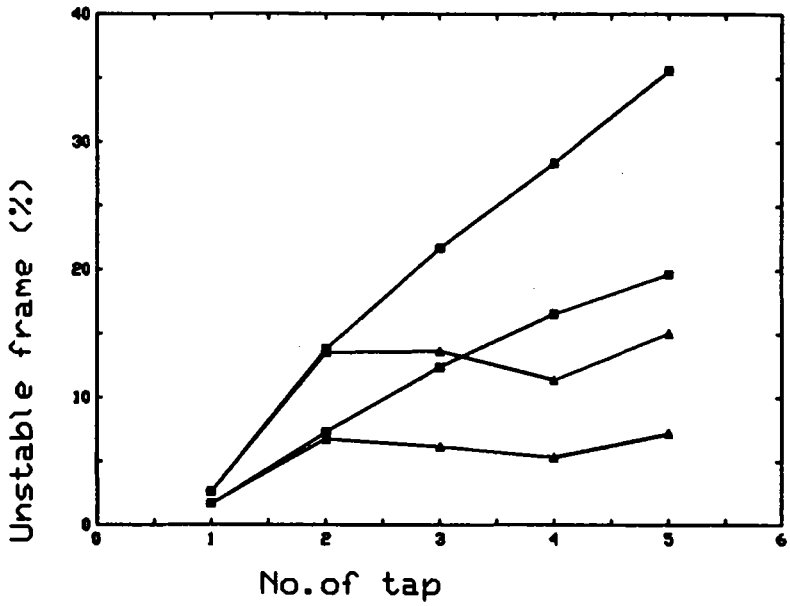


Fig. 3.12 **(a)** Level of instability reflected by crit(SOM) and crit(SUM) (lower curve).

The instability estimated by crit(SUM) (except for order one) stays more or less on the same level, while the instability level according to crit(SOM) is observed to be directly proportional to the number of taps the filter uses. Figs. 3.12(b) and (c) respectively represent the level of instability according to (b)-male and female speech files and (c)-the French and English files. These curves show that the female files tend to be more susceptible to instability than the male files; and that the English files are also slightly more liable to instability than the French files. Based on long term experimental observation, the statement concerning the different levels of sensitivity of male/female files to instability is well justifiable. But the one which suggests that a higher sensitivity to instability in the English files than in the French files should only be taken as an empirical observation rather than a conclusion; inaccuracy may arise in this case due to the different contents of the files used in each category, and the insufficient tests done in this manner.



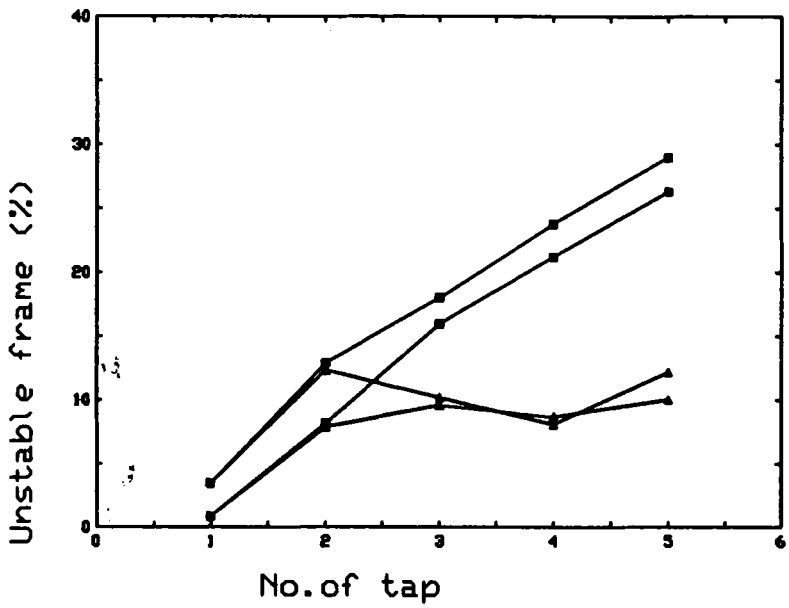


Fig. 3.12 Levels of instability reflected by crit(SUM) and crit(SOM) separated into (b) female (upper pair) and male file categories; (c) English (upper pair) and French file categories.

The inaccuracy of the two criteria is due to the following reasons. Crit(SUM) tolerates a much wider range of β_i than necessary, thus unduly allowing some β_i which contribute poles outside the unit circle to be considered stable; as a result, it tends to indicate a lower level of instability than the actual level, especially for higher-order filter. In contrast, crit(SOM) restricts the range of β_i more than necessary, excluding the β_i which lie between (1)-$\sum_i |\beta_i| = 1$ and (2)-the actual stability boundaries; this results in the tendency of over estimating the true level of instability.

In view of these, the true curve of instability should lie somewhere between the two curves depicted in Fig. 3.12(a). Based on the fact that the physical region defined by crit(SOM) deviates only slightly from the true region, we expect the true unstable curve to lie closer to the upper curve as determined by crit(SOM).

3.7.1 Jury's Models for the Necessary Condition

The models developed in the Section 3.6 help in determining the true level of instability. Based on the earlier observation that the shape of the actual stability region become stabilized even at low values of M. We use M=6,7 and M=5 to model the actual stability regions for 2-tap and 3-tap systems respectively. These two models may not be sufficient to accurately determine the actual instability curve, but will help in narrowing the range within which the true curve lies, or in other words in determining the reliability of using either of the criteria for the stability test.

For 2-tap filter, the stability region alternates its axis of symmetry as M steps up; in which the bulging boundary is located at one region when M is odd, and at another region when M is even (see Fig. 3.4). Thus it is necessary to have two sets of constraints to model the necessary condition. They are:

(for : M − even)

(1) $1 - \beta_1 - \beta_2 > 0$

(2) $1 + \beta_1 - \beta_2 > 0$

(3) $\beta_2^6 - \beta_2^4(\beta_1 - 3) - \beta_2^2(\beta_1^3 - 2\beta_1^2 + 3) - \beta_1^5 - \beta_1^4 + 2\beta_1^3 + \beta_1^2 - \beta_1 - 1 < 0$

and

(for : M − odd)

(1) $1 - \beta_1 - \beta_2 > 0$

(2) $1 + \beta_1 - \beta_2 > 0$

(3) $-[\beta_2^7 + \beta_2^6 - 3\beta_2^5 + (\beta_1^2 - 3)\beta_2^4 + (3 - 2\beta_1^2)\beta_2^3 + (3 - 4\beta_1^2 + \beta_1^4)\beta_2^2$
$-(1 - 2\beta_1^2 + \beta_1^4)\beta_2 + \beta^6 - 3\beta_1^4 + 3\beta_1^2 - 1] > 0$

Because of the rapid convergence of the 2-tap stability region, the perfect square region defined by $\sum_{i=1}^{2} |\beta_i| = 1$ or crit(SOM) is a good model. Consequently, the above model for the necessary condition is expected to give a very accurate level of stability/instability.

For 3-tap filter, the modeling of the stability region is far more complex than in the 2-tap case. Strictly speaking, there are 8 boundaries which make up the 3-tap filter stability region. Two of the boundaries are defined by Jury's first two constraints, while the remaining ones are theoretically furnished by Jury's third constraint. Using the knowledge from the last section, we know that this constraint models the bulging part of the region as well as the two of the four flat surfaces at high M values. But our model uses only $M = 5$; the constraint may model the bulging surface of the region properly, yet the convergence of the two flat surfaces may not have taken place at this value of M. To simulate the performance of the model at high M, we generate two extra constraints [(3) and (4) in the following] to artificially model the two flat surfaces. In other words, the third constraint of Jury's criterion is used here solely for modeling the bulging

part of the stability region, and the following equations (inequalities) form the set of constraints which model the actual stability region in 3-tap filter.

$$(1) 1 - \beta_1 - \beta_2 - \beta_3 > 0$$

$$(2) 1 - \beta_1 + \beta_2 - \beta_3 > 0$$

$$(3) 1 + \beta_1 + \beta_2 + \beta_3 > 0$$

$$(4) 1 + \beta_1 - \beta_2 + \beta_3 > 0$$

$$(5) = \beta_3^5 + \beta_3^4(1 - \beta_1) - 2\beta_3^3(1 + \beta_1) + \beta_3^2(\beta_1^2 + 2\beta_1\beta_2^2 - 4\beta_1\beta_2^2 + 1)$$

$$+(\beta_2^4 - 2\beta_2^2 - \beta_1\beta_2^2 - \beta_1^3 - \beta_1^2 + \beta_1 + 1) < 0$$

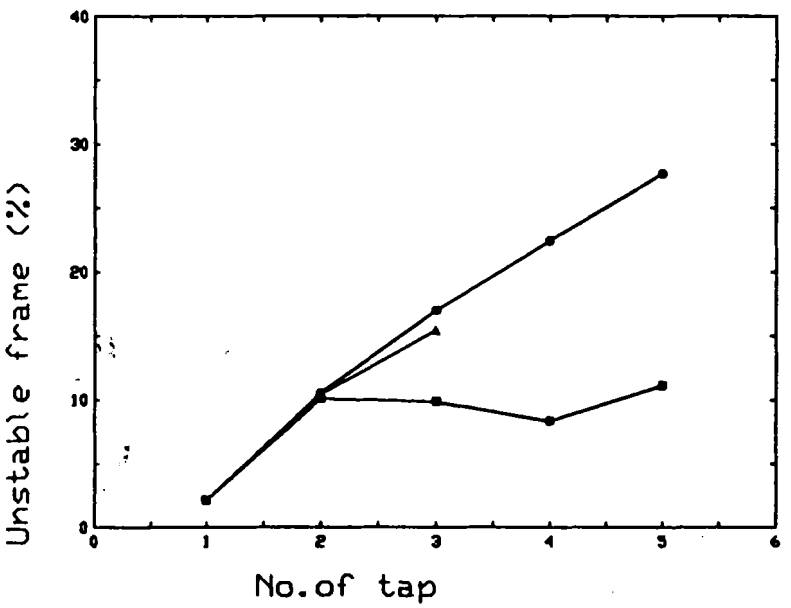


Fig. 3.13 True level of instability reflected by Jury's models for 1,2,3-tap filters.

Using the models described above for the necessary conditions in 2- and 3-tap filters, we repeat the computation of the percentage of the unstable frame. The

results, along with the previous results using the two criteria, are tabulated in Table.3.2. The middle curve in Fig. 3.13 represents the true[†] level of instability. Although the models for the necessary conditions for 4-tap and 5-tap filters are not available as they are too complicated to construct, we expect that the middle curve in Fig. 3.13 should follow the same trend as the curve for $m = 1, 2, 3$. The curves corresponding to those determined by the two criteria are also included for comparison. The figure shows that the reliability in using either criterion to determine the stability status is quite accurate for 1- and 2-tap filters; but it over-estimates the number of unstable frames by about 3.6% for $m = 3$, and we expect increasing deviations for $m = 4, 5$.

Table 3.2 compares the average number of unstable frames per speech file in Appendix H as determined by the two criteria with those determined from Jury's models. Since crit(SOM) has been shown to be the sufficient condition for stability and that it deviates only slightly from the necessary condition, it can serve as a fairly reliable criterion to test the stability of the pitch synthesis filter.

[†] According to Jury's models for the necessary condition.

Unstable Frames (%) Predicted By Different Criteria			
1-TAP			
Speech File	crit(SUM)	crit(SOM)	Model(necessary condition)
Male	1.691	1.691	1.691
Female	2.622	2.622	2.622
Average:	2.157	2.157	2.157
2-TAP			
Speech File	crit(SUM)	crit(SOM)	Model(necessary condition)
Male	6.711	7.275	6.966
Female	13.475	13.770	13.770
Average:	10.093	10.523	10.368
3-TAP			
Speech File	crit(SUM)	crit(SOM)	Model(necessary condition)
Male	6.133	12.331	9.136
Female	13.580	21.642	17.677
Average:	9.857	16.989	13.407
4-TAP			
Speech File	crit(SUM)	crit(SOM)	Model(necessary condition)
Male	5.320	16.554	not available
Female	11.385	28.327	not available
Average:	8.353	22.441	not available
5-TAP			
Speech File	crit(SUM)	crit(SOM)	Model(necessary condition)
Male	7.204	19.707	not available
Female	15.013	19.707	not available
Average:	11.109	27.642	not available

Table 3.2 Comparison of the level of instability predicted by crit(SUM) and crit(SOM) against the expected true level.

Chapter 4 Stabilization Process

Being able to detect the instability in $H_p(z)$ is only part of the battle; the more essential task is the correction of this undesirable unstable status. This chapter presents several alternative ways to correct the instability problems, and compares the efficiency in using them to stabilize the pitch synthesis filter. At this point, the criterion used to gauge the efficiency of each method is by measuring the consequent loss in prediction gain accompanied by the stabilization process. In Chapter 5, we evaluate the efficiency from a perceptual point of view.

4.1 Methods of Stabiization

We propose two basic methods to stabilize the pitch synthesis filter; they are described separately in this section. The fundamental requirement is to satisfy the sufficient condition dictated by the crit(SOM)

$$\sum_i |\beta_i| < 1 \quad i = 1, 2, ..., m. \tag{4.1}$$

Hence, the objective of stabilization process is to modify the predictor coefficients, in the most efficient manner, to values so that their new SOM is less than unity. The term 'the most efficient' implies a way of stabilization process that incurs the least expense in terms of losing the prediction gain. During unstable

frames, some (or all in case of single-tap filter) of the poles of the pitch synthesis filter lie outside the unit circle. Thus from the point of view of the system function, the stabilization process implies a relocation of the pole positions in such a way that all of them are repositioned to inside the unit circle.

4.1.1 Method(1): Unity Replacement

The first method of stabilization is a direct implementation of the marginal constraint in crit(SOM). In other words, it is to modify the predictor coefficient values so that $\sum_i |\beta_i| = 1$. In one-tap filter, the implementation of this algorithm is simply replacing any $\beta > 1$ by unity. In a multi-tap filter, the implementation can be done in two different manners; one is to scale each of the coefficients β_i by a certain common factor, the other is to scale them with different factors. The details of these two approaches are described in Section 4.4.

4.1.2 Method(2): Reciprocal Replacement

Another method of stabilization proposed in this study is called the reciprocal replacement method. It is equivalent to moving every pole of the system with magnitude $|p_i| > 1$ radially inward to a new location $\frac{1}{|p_i|}$ away from the origin. When applied to 1-tap filter, this method requires the substitution of the single predictor coefficient $|\beta| > 1$ by $\frac{1}{|\beta|}$. Since the poles in 1-tap filter share a common magnitude, the effect of this substitution results in shrinking the poles radially from outside to inside the unit circle by an equal amount. Because there is no direct relation between the pole magnitudes and the predictor coefficients in a multi-tap system, this method is not effective when applied to the multi-tap system.

4.2 The Rationale of the Proposed Methods

In this section, we make use of the single-tap filter to provide an insight into the rationale of using the two methods of stabilization proposed in the previous section. These analytical studies serve to justify the potential effectiveness of using the two methods as stabilization tools.

The one-tap pitch synthesis filter has characteristic system function

$$A_p(z) = 1 - \beta z^{-M} \tag{4.2}$$

The two parameters associated with it are: M(*the lag*) and β(*the only coefficient*). Note that unless otherwise specified and for notational simplicity, when these parameters refer to the pitch synthesis filter, M and β should always be taken to represent the optimal parameters before any stabilization. In the discussion of the analysis where confusion may arise and emphasis is required, the optimized parameters will be specifically denoted as M_{opt} and β_{opt} respectively.

In the analysis, a normalized correlation coefficient $\alpha(\tau)$ of the input signal is defined as:

$$\alpha(\tau) = \sum_k \frac{s_k s_{k-\tau}}{\left[\sum_k s_k^2 \sum_k s_{k-\tau}^2\right]^{\frac{1}{2}}}, \tag{4.3}$$

where the limit of the summation $\sum_k$ runs from k=(1 : N (frame size[†])) is computed over a certain range to search for the local optimal values for β and M in that range. The optimality of these two parameters are based on the minimizing the prediction error, which is the output of the inverse filter $A_p(z)$.

The basic procedure for searching for these optimal parameters M_{opt} and β_{opt} is to evaluate $\alpha(\tau)$ over the range (τ=minlag : maxlag) of the past input, where

[†] Typically 200 samples.

the minimum and maximum lags are set at 20 and 120 respectively. This range is so chosen to cover the complete range of human pitch periods (in samples). The minimum lag also serves to prevent the pitch prediction process from interfering with the formant prediction, which removes the near-sample-based redundancy of the speech. Once $\alpha(\tau)$ is evaluated, the optimal lag M_{opt} is set to assume the value of τ that corresponds to the first maximum of the correlation function.

The segmental (on a per frame basis) mean square error is defined as:

$$\begin{aligned}\overline{e^2} &= \sum_k e_k^2 \\ &= \sum_k (s_k - \beta s_{k-M})^2 \\ &= \sum_k s_k^2 - 2\beta \sum_k s_k s_{k-M} + \beta^2 \sum_k s_{k-M}^2\end{aligned} \tag{4.4}$$

Taking the derivative of the error energy function $\overline{e^2}$ with respect to β and setting the result to zero yields the single optimal coefficient

$$\beta_{opt} = \frac{\sum_k s_k s_{k-M_{opt}}}{\sum_k s_{k-M_{opt}}^2}. \tag{4.5}$$

The residual energy in Eq.(4.4) is minimized when the coefficient β is replaced with β_{opt}, giving a minimum residual energy below

$$\overline{e^2}(min) = \sum_k s_k^2 - \frac{(\sum_k s_k s_{k-M_{opt}})^2}{\sum_k s_{k-M_{opt}}^2}. \tag{4.6}$$

4.2.1 Unity Replacement Method

If one rewrites the minimum residual energy function in Eq.(4.6) in the following manner:

$$\overline{e^2}(min) = \sum_k s_k^2 \Big[1 - \frac{(\sum_k s_k s_{k-M})^2}{\sum_k s_k^2 \sum_k s_{k-M}^2}\Big], \tag{4.7}$$

the second term in the bracket is easily recognized to be the square of the normalized correlation coefficient $\alpha(\tau = M)$ as defined in Eq. (4.3). $\alpha(M)$ can be a potential candidate as a substitute for the unstable β as its value is always less than unity. In terms of formulation, there is also a high degree of similarity between β in Eq.(4.5) and the correlation coefficient at $\tau = M$ below

$$\alpha(M) = \frac{\sum_n s_k s_{k-M}}{[\sum_n s_k^2 \sum_n s_{k-M}^2]^{\frac{1}{2}}}. \tag{4.8}$$

In an ideal voiced speech segment where the current sample is the copy of the previous sample delayed by one period such that $s_k = s_{k-M}$, the expressions in these two equations are essentially the same. This suggests the possibility of instantaneously replacing the unstable coefficient β by $\alpha(M)$ to generate a stable system. For comparison, Fig.4.1 shows the contours of the segmental β and $\alpha(M)$. Judging from the close values of the two quantities across the utterance†, it is quite feasible to replace β by $\alpha(M)$ unconditionally at a certain loss in prediction gain. The advantage of using this proposed scheme is that the quantity $\alpha(M)$ is readily available for use, and there is no need to test for stability. But to minimize the loss, the coefficient β should be replaced selectively, i.e., only when it exceeds unity. In this manner, β always stays optimal except when it is substituted with $\alpha(M)$ occasionally during the unstable frames. The less disturbance to the optimal coefficient values in this selective replacement scheme leads to a higher value of prediction gain than that obtained from the unconditional replacement scheme. Experimental results in Section (4.3.2) later compare these two schemes and their consequences.

Theoretically, one must strive to minimize the deviation of the new, modified

† Except during unstable frames.

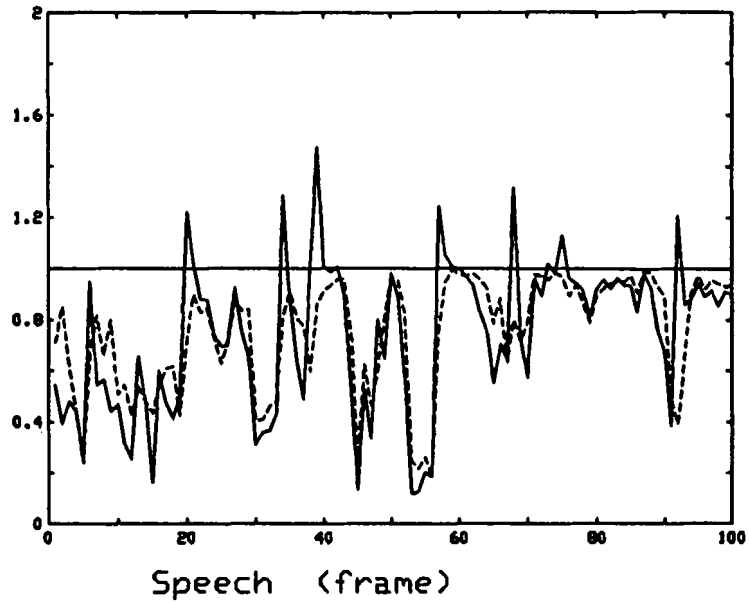


Fig. 4.1 Contours of β (solid line) and $\alpha(\tau = M)$.

stable coefficient denoted as, say, $\bar{\beta}_1$ from the unstable β. The further $\bar{\beta}_1$ deviates from β, the larger the residual becomes and the lower is the prediction gain. According to the stability criterion, the maximum value that can be assumed by the coefficient for stability is limited by unity. Therefore, for stability and for minimum distortion, the unstable coefficients should be suppressed only to unity. Indeed, if the correlation coefficient were defined in the following manner

$$\alpha'(\tau) = \bar{\beta}_1 = \frac{\sum_k s_k s_{k-\tau}}{\sum_k s_k s_{k-M}}, \tag{4.9}$$

it remains a normalized function and its value at M is unity. This, in a way, demonstrates the validity of the unity replacement for stabilizing the pitch synthesis filter. A more elaborate justification of the unity replacement proposal is provided at a later section.

4.2.2 Reciprocal Replacement Method

This algorithm is inspired by an all-pass system in which the poles and the

zeros are located on opposite sides of the unit circle. The constant magnitude response of the all-pass system is the result of the reciprocal relation of their pole/zero pairs, which allows the magnitudes of the zeros to vary in direct proportion to those of the poles, giving a constant magnitude response at all frequencies. It is easier to see the cause of this constant magnitude response in its analog counterpart, where the pole/zero of the system are located side by side at equal length along the imaginary axis. Under analog-digital (bilinear) transformation, the left half plane of the s-plane maps to a region on the z-plane confined by a unit circle, the right half plane to the outside region, and the imaginary axis onto the unit circle. As a result, the pole/zero pair at s_p[†] and $-s_p$ maps to a pair at z_p and its reciprocal $\frac{1}{z_p}$.

When the filter for a frame is unstable, the magnitudes of all the poles $|r| = |\beta|^{\frac{1}{M}}$ are greater than one. Assuming that all the poles outside the unit circle are moved to new locations, say at $\bar{\beta}_2 = \frac{1}{|r|} = \frac{1}{|\beta|^{\frac{1}{M}}}$ inside the unit circle, then not only the stability condition is satisfied, also because of the reciprocal relation between the original and new locations of the poles, the resulting magnitude response function in dB also remains unaltered. In other words, the relocation of the poles from $|r|$ to $\frac{1}{|r|}$ preserves the shape of the original spectrum.

The above two methods may have been derived with different immediate objectives in mind; where the first method strives to minimize the distortion done to the coefficient value, and the second method attempts to preserve the spectral property of the filter, both algorithms achieve the essential goal in fulfilling the basic stability requirement, i.e., to reduce the unstable coefficient to either less than or equal to unity.

---

† real

The two proposed methods in Sections 4.2.2 and 4.2.3 are both applicable to a single-tap filter. As mentioned earlier, the lack of any direct relationship between the predictor coefficients and the pole distribution makes the reciprocal replacement method ineffective when applied to the multi-tap system. Considering the different approaches required to stabilize the single-tap and the multi-tap system, we separate the discussion of the experimental results of these two systems.

4.3 Single-Tap Filter

In this section, we like to estimate the price tags of the two methods of stabilization when applied to a single-tap system in terms of prediction gain. The segmental prediction gain, which is defined as the ratio in dB between the average energy of the input signal and that of the residual signal, is computed for each frame; from which the average of a complete speech file can be evaluated. The same procedure is then repeated after the stabilization using each method, and the loss in prediction gain is taken to indicate the cost of the stabilization process associated with that particular method.

4.3.1 Testing Method(1)

At this stage, we are interested mainly in the side effect of modifying the pitch predictor coefficients on the residual signal. Thus in order to see more clearly the effect on the residual from the stabilization process, we use only the pitch analysis filter in the analyzer so that a larger amplitude residual signal is produced at the analyzer output.

The bottom line proposed in method(1) is to replace the unstable coefficient ($\beta > 1$) by unity. In the development of this method, we suggested the direct use of the correlation function $\alpha(M)$ as a convenient substitute regardless of the

stability of the system. The two segmental quantities (β and α(M)) generated from processing an audio files have been shown in Fig. 4.1 to follow a very similar contour except at segments where $\beta > 1$; they show high degree of resemblance during the voiced segments. In 1-tap filter, we see a clear tendency for the unstable frames to occur at the junction between silence and voice. Using the ten speech files listed in Appendix H as data base, and neglecting the formant prediction in the analysis filter, we noted a slight drop in the average prediction gain per file of about 0.4dB as a result of replacing β by $\alpha(M)$ unconditionally; and only 0.06 dB when the replacement takes place selectively (i.e., only at $\beta > 1$). It is clear that the loss in prediction gain even by using the correlation coefficient to replace the unstable coefficient is quite reasonable. This, at least, provides an attractive alternative to the other algorithms explicitly proposed in this chapter.

Some other characteristics of the unstable coefficients observed during the experiments include the following: For the same utterance, the female file tends to have a higher rate of instability and higher coefficient values than the corresponding male counterpart. We also observe that the average value of the optimal lag for the unstable frames is comparable to the average pitch in samples. For the ten speech files, the average lag $M^{\dagger}$ for the five female files is 38.3 samples and that for the five male files is 70.1 samples; at a sampling rate of 8 kHz., these two figures are good representatives of the pitch periods of female and male speakers respectively.

When the unstable coefficient ($\beta > 1$) is replaced by unity instead, the prediction gain loses only 0.03 dB when compared to the original prediction gain. Table 4.1 lists the original prediction gain of the ten speech files used for this

† Exclusively for unstable frames.

test, along with the prediction gains obtained from stabilization processes using different methods. In the table, SNR is the prediction gain when no stabilization is involved; SNR(1a), SNR(1b) and SNR(1c) respectively represent the prediction gains after stabilization using $\alpha(M)$ unconditionally, using $\alpha(M)$ selectively and using the unity replacement method; whereas SNR(2) indicates the resulting prediction gain using the reciprocal replacement, which is discussed next.

Prediction Gains (1-Tap)					
Speech File	SNR	SNR(1a)	SNR(1b)	SNR(1c)	SNR(2)
CATM8	3.01	2.68	2.95	2.99	2.97
PB1M1	3.87	3.20	3.80	3.83	3.78
DOUG5	4.75	4.41	4.68	4.72	4.68
PIPM8	3.58	3.33	3.52	3.56	3.52
PB1M5	4.11	3.46	4.03	4.07	4.01
Male Average:	3.86	3.42	3.80	3.83	3.79
CATF8	6.14	5.81	6.08	6.11	6.06
PB1F1	8.14	7.73	8.11	8.13	8.11
VOICF5	8.30	8.14	8.24	8.27	8.21
PIPF8	8.55	8.18	8.49	8.53	8.49
PB1F5	7.73	7.23	7.65	7.70	7.63
Female Average:	7.77	7.42	7.71	7.75	7.70
Total Average:	5.82	5.42	5.76	5.79	5.75

Table 4.1 Prediction gain in one-tap filter

4.3.2 Testing Method(2)

In testing the efficiency of the second method, we examined the effect on the prediction gain as a result of modifying the unstable coefficient to its reciprocal position $\frac{1}{\beta}$. To verify that this is indeed the critical point, some other points near

$\frac{1}{\beta}$ were also tested to see the various effects of using these points as replacements for $\beta > 1$. In this test, a special matching criterion is used to judge the result: a closer spectral match between the original and the modified system functions is construed to indicate less distortion in the resulting residual.

Replacing the unstable coefficient $\beta > 1$ by various points above and below the critical point $\bar{\beta}_2 = \frac{1}{\beta}$, the resulting spectral peaks from the corresponding pitch synthesis filter were seen to be diminishing while the valleys remained intact on the same level. Fig.4.2.(a) shows the resulting spectra when $\beta = 1.05$ is replaced with points above, below and equal to $\bar{\beta}_2 = \frac{1}{\beta} \approx 0.95$. Comparing these spectra with the original spectrum of $\beta = 1.05$ shown in Fig.4.2.(b), we observe that only the spectrum corresponding to $\bar{\beta}_2 = \frac{1}{\beta}$ retains a similar shape to that of the original spectrum. Fig.4.3 shows the spectra of Fig.4.2 displayed in dB; it clearly indicates that the only the spectrum in Fig.4.2.(a) that overlaps the original spectrum in Fig.4.2.(b) is the one which has a replacement value of 0.95† (the middle spectrum). This evidence suggests that the original spectrum can be retained only when β is replaced by its reciprocal, and that any other replacements will only result in a spectral distortion in one way or another.

To implement the reciprocal replacement method, we can cascade the unstable $H_p(z)$ with an all-pass filter, say, $H_a(z)$ that has all its poles located at $|\beta|^{\frac{1}{M}} > 1$. The equivalent of this cascade system is

$$H_p(z)H_a(z) = \left[\frac{z^M}{z^M - \beta} \right] \left[\frac{z^M - \beta}{z^M - \frac{1}{\beta}} \right] = \frac{1}{1 - \beta z^{-M}}. \qquad (4.10)$$

† Note that $\frac{1}{\beta}$ where β =1.05 is actually close to 0.9524.

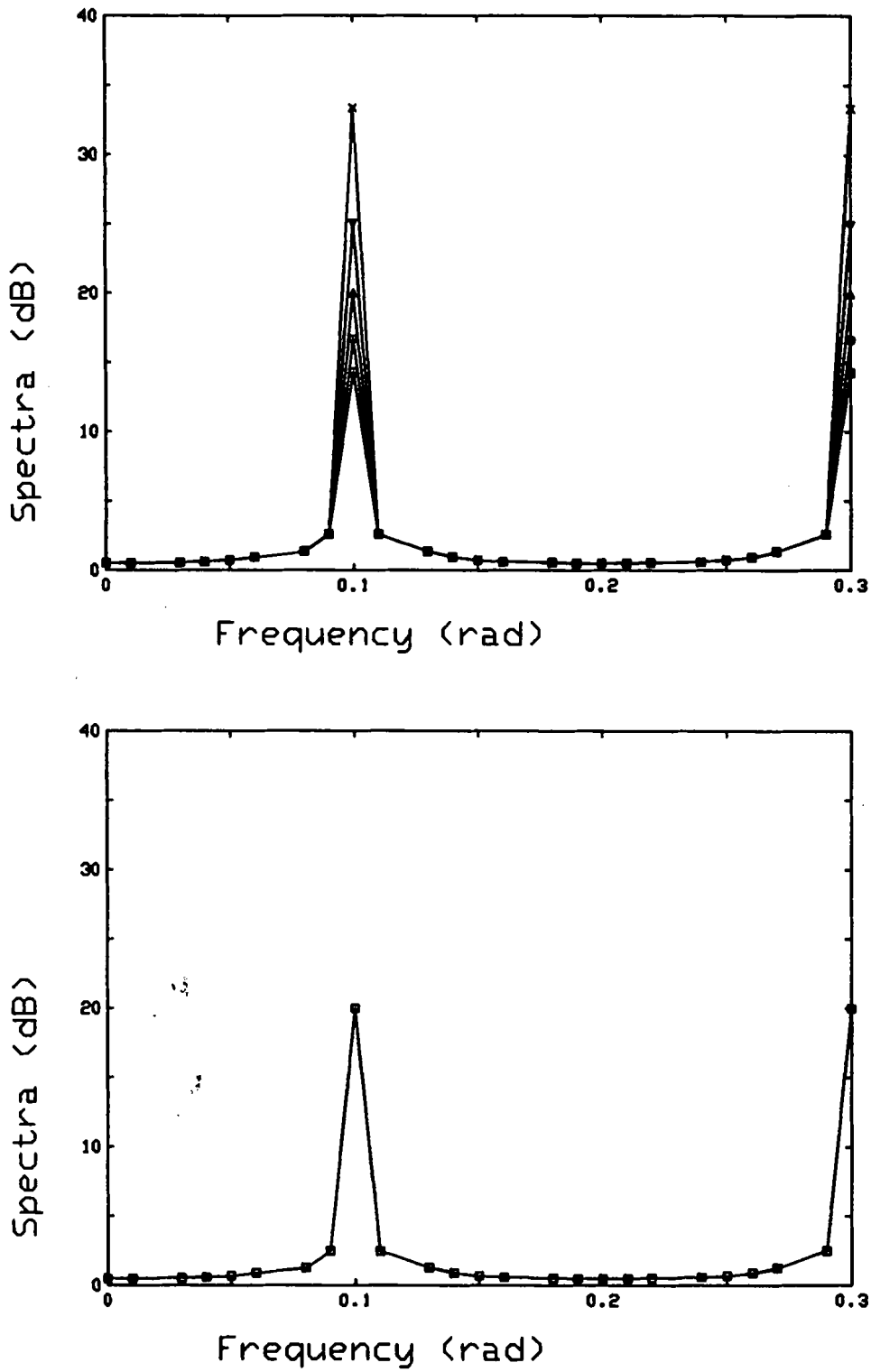


Fig. 4.2 **(a)** Linear spectra due to $\bar{\beta}_2 = 0.97, 0.96,$ $0.95 \approx \frac{1}{\beta}, 0.94, 0.93$ (upper figure); **(b)** Linear spectra due to $\bar{\beta}_2 = \beta = 1.05$.

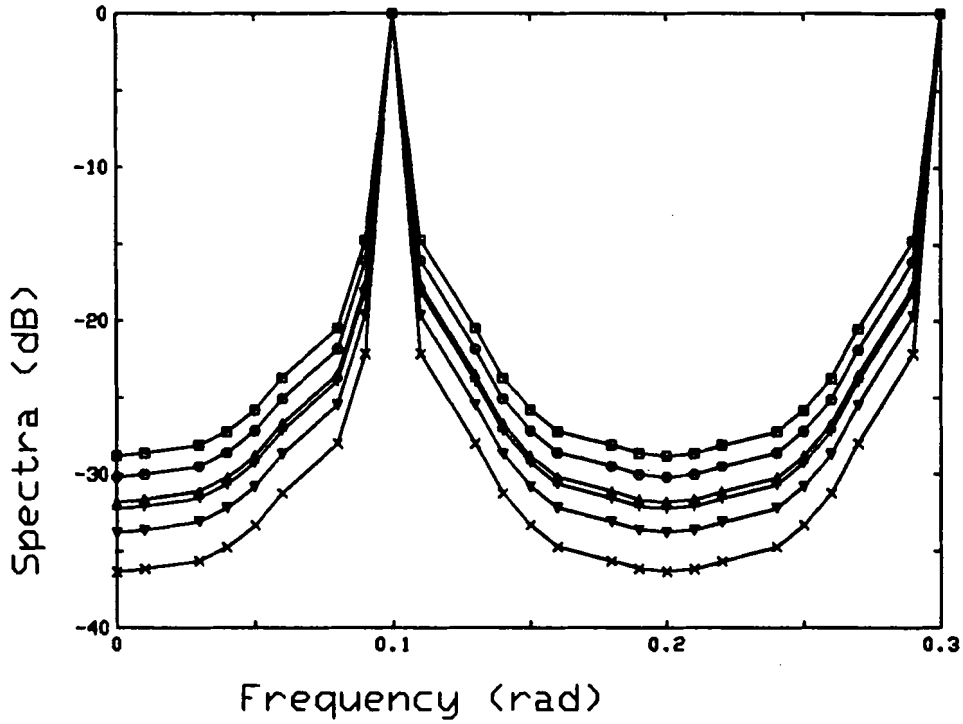


Fig. 4.3 Logarithmic spectra of Fig.4.2 where the almost overlapping spectra are due to $\beta = 1.05$ and $\bar{\beta}_2 = 1.09 \approx \frac{1}{\beta}$.

The M poles from the unstable filter will cancel with the M zeros from the all-pass system, leaving behind M poles inside the unit circle with common magnitude $\frac{1}{|\beta|^{1/M}}$. In terms of prediction gain, tests show that using the reciprocal replacement method for stabilization incurs a loss of 0.07 dB as shown in Table 4.1.

4.3.3 Sub-optimal parameters after stabilization

The previous two sections demonstrate that the stabilization of the one-tap filter is a simple process to reduce the unstable coefficient value β to a certain value $\bar{\beta}$ less than unity. The assumption of this stabilization process is that regardless of the new modified coefficient $\bar{\beta}$, we still retain the original lag as the

optimal parameter. The following provides some theoretical arguement to verify the above statement.

The expression for the residual function with optimal parameters β and M has been derived in Section 4.2 to be

$$\overline{e^2} = \sum_k s_k^2 - 2\beta \sum_k s_k s_{k-M} + \beta^2 \sum_k s_{k-M}^2. \tag{4.11}$$

This residual function will no longer be minimum if the optimal coefficient β is modified. Let $\bar{\beta} \leq 1$ denotes the new value† that replaces $\beta > 1$. It was argued earlier that when the speech segment is voiced, the two quantities $\sum_k s_k s_{k-M}$ and $\sum_k s_{k-M}^2$ (optimal M) in Eq.(4.11) were approximately equal as a result of the quasi-periodicity. Under this assumption, it can be shown that the residual function in Eq.(4.11) can be expressed (as a function of the suboptimal coefficient $\bar{\beta}$) as:

$$\bar{e}(\bar{\beta})^2 = \sum_k s_k^2 \Big[1 - \frac{(\sum_k s_k s_{k-M})}{\sum_k s_k^2} [F(\bar{\beta})]\Big] \tag{4.12}$$

where

$$F(\bar{\beta}) = 2\bar{\beta} - \bar{\beta}^2. \tag{4.13}$$

Minimizing the residual $\bar{e}(\bar{\beta})^2$ with $\bar{\beta}$ as parameter is equivalent to maximizing the function $F(\bar{\beta})$. By taking the derivative $\frac{dF(\bar{\beta})}{d\bar{\beta}}$ and setting it to zero, it is clear that $\bar{\beta} = 1$ is the value that minimizes the residual in Eq. (4.12). Fig. 4.4 plots the function $F(\bar{\beta})$. According to the relationship between $\bar{e}(\bar{\beta})^2$ and $F(\bar{\beta})$ in Eq. (4.12), the figure indicates that as the coefficient is slowly reduced from $\beta > 1$ to unity, the energy of the residual diminishes and becomes optimal at $\beta = 1$. But further drop of β below unity level corresponds to a decrease in $F(\bar{\beta})$, and leads to an increase in the residual energy again.

† Assumed to be variable here.

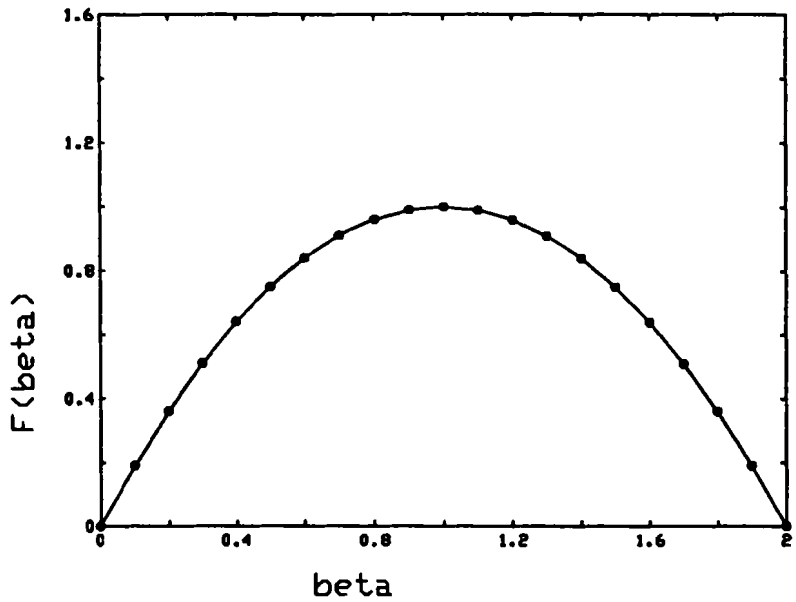


Fig. 4.4 Function $F(\beta)$.

To support the arguement that the original optimal lag M_{opt} remains optimal after stabilization process, let us express the residual energy in Eq.(4.11) as a function of general lag M[†]

$$\begin{aligned}\overline{e^2}(M) &= \sum_k s_k^2 - 2\sum_k s_k s_{k-M} + \sum_k s_{k-M}^2 \\ &= \sum_k s_k^2 \Big[1 - \frac{(2\sum_k s_k s_{k-M} - \sum_k s_{k-M}^2)}{\sum_k s_k^2}\Big] \end{aligned} \tag{4.14}$$

As M deviates slightly from M_{opt}, the term $\sum_k s_{k-M}^2$ does not vary much, and therefore can be regarded as constant. The term $\sum_k s_k s_{k-M}$ on the other hand peaks at M_{opt}, and gradually diminishes as M deviates from M_{opt}. As a result, the second term in the bracket of Eq. (4.14) peaks at $M = M_{opt}$, which means $\overline{e^2}(M_{opt}) < \overline{e^2}(M)$ for $M \neq M_{opt}$. Hence $M = M_{opt}$ is the optimal value.

Therefore, we conclude that if minimizing the residual energy is the main

[†] Also assumed to be variable in this discussion.

objective, we must suppress the predictor coefficient to unity and retain the originally derived optimal lag value M as defined in Eq. (4.5).

4.4 Multiple-Tap Filter

For a pitch predictor using more than one tap, say m taps, the inverse filter in general can be expressed as

$$\begin{aligned} A_p(z) &= 1 - P(z) \\ &= 1 - \sum_i \beta_i z^{-M_i}, \end{aligned} \tag{4.15}$$

where $i = 1, 2, ..., m$ and β_i, M_i are the i^{th}-coefficient value and its associated lag. The computation of the m optimal coefficients β_i are similar to the one described previously for a single-tap filter in Section 4.2 and is described in the following.

The normalized correlation coefficient $\alpha(\tau)$ as defined in Eq. (4.3) is computed for lags running from a minimum value of 20 to a maximum value of 120, and the lag M corresponding to the maximum value of $\alpha(\tau)$ is deemed to be the pitch value for the frame of speech being processed. We can position the first coefficient β_1 to have this lag value, in which case $M_1 = M$; but the following scheme which positions the taps depending on the number of coefficients provides a better alignment between the pitch and the location of the taps, and results in a better prediction.

m	M	M_1	M_2	M_3	M_4	M_5
1	M_1	M				
2	M_1	M	$M+1$			
3	M_2	$M-1$	M	$M+1$		
4	M_2	$M-1$	M	$M+1$	$M+2$	
3	M_3	$M-2$	$M-1$	M	$M+1$	$M+2$

Using the above scheme, the segmental mean square prediction error is expressed as

$$\begin{aligned}\overline{e^2} &= \sum_k e_k^2 \\ &= \Big[\sum_k s_k - \sum_{i=1}^{m} \beta_i s_{k-M_i}\Big] \end{aligned} \tag{4.16}$$

To minimize the residual energy, the derivatives of Eq. (4.16) with respect to $\beta_i, i = 1, 2, ..., m$ are evaluated and set to zero. These result in m equations relating the m coefficients β_i to various correlation terms. For instance, for a 3-tap filter ($m = 3$), the following three equations are generated.

$$\begin{aligned}\frac{\delta \overline{e^2}}{\delta \beta_1} &= \Big[\beta_1 \sum_k s_{k-M_1}^2 + \beta_2 \sum_k s_{k-M_1} s_{k-M_2} + \beta_3 \sum_k s_{k-1} s_{k-M_3}\Big] \\ &= \sum_k s_k s_{k-M_1} \end{aligned} \tag{4.17.a}$$

$$\begin{aligned}\frac{\delta \overline{e^2}}{\delta \beta_2} &= \Big[\beta_1 \sum_k s_{k-M_2} s_{k-M_1} + \beta_2 \sum_k s_{k-M_2}^2 + \beta_3 \sum_k s_{k-M_2} s_{k-M_3}\Big] \\ &= \sum_k s_k s_{k-M_2} \end{aligned} \tag{4.17.b}$$

$$\begin{aligned}\frac{\delta \overline{e^2}}{\delta \beta_3} &= \Big[\beta_1 \sum_k s_{k-M_3} s_{k-M_1} + \beta_2 \sum_k s_{k-M_3} s_{k-M_2} + \beta_3 \sum_k s_{k-M_3}^2\Big] \\ &= \sum_k s_k s_{k-M_3} \end{aligned} \tag{4.17.c}$$

The three optimal coefficients are evaluated by simultaneously solving Eqs. (4.17). In general, the optimal coefficients for an m^{th}-order pitch synthesis filter can be computed from the following matrix equation

$$\begin{bmatrix} \phi(M_1,M_1) & \phi(M_1,M_2) & \phi(M_1,M_3) & \cdots & \cdots & \phi(M_1,M_m) \\ \phi(M_2,M_1) & \phi(M_2,M_2) & \phi(M_2,M_3) & \cdots & \cdots & \phi(M_2,M_m) \\ \phi(M_3,M_1) & \phi(M_3,M_2) & \phi(M_3,M_3) & \cdots & \cdots & \phi(M_3,M_m) \\ \vdots & \vdots & \vdots & \cdots & \cdots & \vdots \\ \vdots & \vdots & \vdots & \cdots & \cdots & \vdots \\ \vdots & \vdots & \vdots & \cdots & \cdots & \vdots \\ \phi(M_m,M_1) & \phi(M_m,M_2) & \phi(M_m,M_3) & \cdots & \cdots & \phi(M_m,M_m) \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \\ \vdots \\ \vdots \\ \vdots \\ \beta_m \end{bmatrix}$$

$$= \begin{bmatrix} \phi(0,M_1) \\ \phi(0,M_2) \\ \phi(0,M_3) \\ \vdots \\ \vdots \\ \vdots \\ \phi(0,M_m) \end{bmatrix} \tag{4.18}$$

where $\phi(M_i,M_j) = \sum_k s_{k-M_i} s_{k-M_j}$. In compact form, the above matrix equation can be written as:

$$\sum_{j=1}^{m} \beta_j \phi(M_i,M_j) = \phi(0,M_i), \qquad i = 1,2,3,...,m. \tag{4.19}$$

The lack of direct relation between the predictor coefficients and the distribution of the poles makes the stabilization in multi-tap system more difficult to analyze than in the single-tap case, and it also makes method(2) — the reciprocal replacement scheme ineffective. With method(1) — the unity replacement scheme, we examine two different approaches of implementation. Basically, we need to scale each of $(\beta_1,\beta_2,\beta_3)$ so that $\sum_{i=1}^{3} |\beta_i| = 1$. One way to achieve this is to scale each of the coefficients by a common factor F_c, but one can also scale the coefficients by different factors F_i, $i = 1,2,3$. In the following, we describe these two different approaches separately, and study the implications of using F_i as opposed to F_c.

4.4.1 Common Scaling Factor

To multiply each of the unstable coefficients by a common factor F_c when normalizing the SOM, let the new set of coefficient be

$$\hat{\beta}_i = F_c \beta_i \tag{4.20}$$

where F_c is restricted to

$$F_c < \frac{1}{\sum_i |\beta_i|}. \tag{4.21}$$

The new SOM thus becomes

$$\sum_i \hat{\beta}_i = F_c \sum_i \beta_i < 1 \tag{4.22}$$

Using the common scaling factor F_c, each of the optimal coefficients during an unstable frame is reduced by the same proportion to unity. Theoretically, this process shrinks the magnitudes of the original poles to different extents, but it distorts the shape of the spectrum.

4.4.2 Differential Scaling Factor

In order to preserve the spectral shape of the system, it is the pole magnitudes of the system that must be radially down-scaled with the same proportion. This can be accomplished by multiplying the coefficients by differential factors. The process is equivalent to transforming the original inverse system function from $A_p(z)$ to $A_p(z')$, such that

$$|z'| = a|z|, \qquad 0 < a < 1. \tag{4.23}$$

Suppose that the new system function of 3-tap filter has characteristic equation

$$1 - \beta_1 (z')^{-(M-1)} - \beta_2 (z')^{-M} - \beta_3 (z')^{-(M+1)} = 0. \tag{4.24}$$

Substitution of Eq. (4.23) into Eq. (4.24) gives

$$1 - (\beta_1 a^{-(M-1)})z^{-(M-1)} - (\beta_2 a^{-M})z^{-M} - \beta_3(a^{-(M+1)})z^{-(M+1)} = 0$$

$$or: \qquad 1 - \hat{\beta}_1 z^{-(M-1)} - \hat{\beta}_2 z^{-M} - \hat{\beta}_3 z^{-(M+1)} = 0. \tag{4.25}$$

For stability, we require

$$|\hat{\beta}_1| + |\hat{\beta}_2| + |\hat{\beta}_3| < 1 \tag{4.26.a}$$

$$\Rightarrow \quad |\beta_1 a^{-(M-1)}| + |\beta_2 a^{-M}| + |\beta_1 a^{-(M+1)}| < 1 \tag{4.26.b}$$

$$\Rightarrow \quad |\beta_1 F_1| + |\beta_2 F_2| + |\beta_1 F_3| < 1. \tag{4.26.c}$$

From Eqs. (4.26.b,c), the differential scaling factors F_i for the three coefficients are identified as

$$F_1 = a^{-(M-1)}$$

$$F_2 = a^{-M}$$

$$F_3 = a^{-(M+1)}.$$

Setting $b = a^{-1}$ for notational convenience, we have from Eq. (4.26.c)

$$(|\beta_1| + |\beta_2|b + |\beta_3|b^2)b^{(M-1)} < 1 \tag{4.27}$$

To evaluate F_i from the above equation, the method of linear interpolation is applied to the limiting case of Eq. (4.27), which is

$$f(b) = (|\beta_1| + |\beta_2|b + |\beta_3|b^2)b^{(M-1)} - 1 = 0 \tag{4.28}$$

F_i are solved by an iterative process until Eq. (4.28) is satisfied within a given error constraint. The number of iterations is found to be directly proportional

to both the accuracy required and the original value of the SOM of the coefficients. Although some relatively high SOM's do occur occasionally, the SOM in a 3-tap filter during unstable frames lie mostly between 1 and 2. Taking into account only those coefficients with an amplitude range up to 3, the number of iterations required to achieve certain levels of accuracy are recorded in Table 4.2, where the number in bracket indicates the average of the number of iterations in that particular category. On the average, only 5 to 7 iterations are required to compute the differential factors F_i for $i = 2, 3$ respectively to achieve a relatively high level of accuracy.

2-TAP			
Error Constraint	10^{-3}	10^{-4}	10^{-5}
Range of Iterations (Ave.)	$1 \rightarrow 6$ (3)	$2 \rightarrow 10$ (4)	$3 \rightarrow 12$ (5)
3-TAP			
Error Constraint	10^{-3}	10^{-4}	10^{-5}
Range of Iterations (Ave.)	$2 \rightarrow 8$ (5)	$3 \rightarrow 11$ (6)	$3 \rightarrow 13$ (7)

Table 4.2 Number of iterations required to compute $F_i, i = 2, 3$.

In comparison, F_i are found to be very close to each other as well as to the constant F_c. Their close proximity to one another makes it impossible to compare them in actual values. But taking F_c as a reference point and expressing F_i as F_i/F_c in percentage, it is possible and interesting to see how their magnitudes are related to one another. For 2-tap filter, F_1 and F_2 lie respectively above and below F_c; whereas in the 3-tap case, F_1 and F_3 lie above and below F_c and F_2 lies more or less on the same level as F_c. Figs. 4.5.(a),(b) show respectively the traces of the ratios F_i/F_c in percentage for speech file 'VOICF5'[†] using 2- and

[†] Up to frame #100.

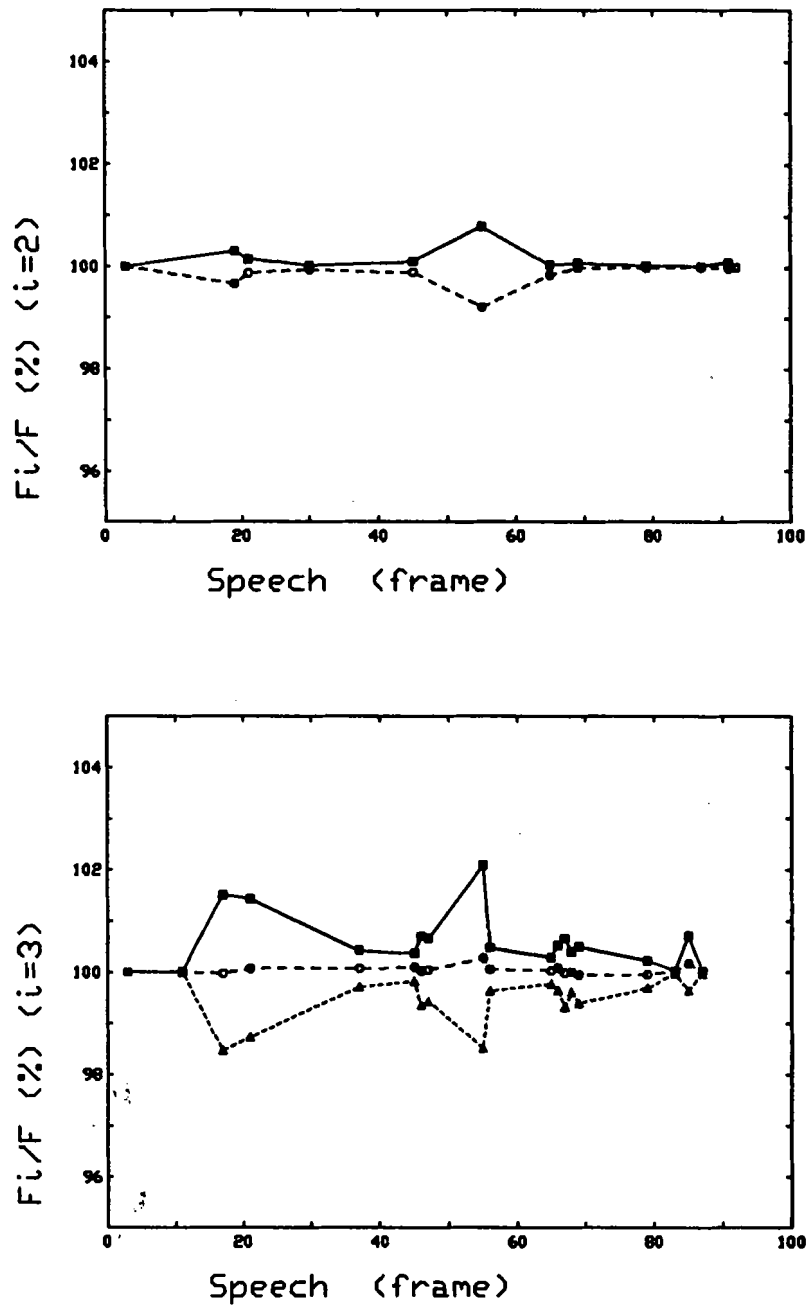


Fig. 4.5 **(a)** Differential factors F_1, F_2 for 2-tap (upper figure); **(b)** Differential factors F_1, F_2, F_3 for 3-tap filter.

3-tap filters. In actual values, F_i and F_c are very close to one another. Appendix J provides a listing of the actual values of F_c and F_i corresponding to Fig.4.5.

4.4.3 Experimental Results

The stability criterion (crit(SOM)) requires that any $\sum_i |\beta| > 1$ be suppressed to unity. We have introduced two schemes for the stabilization process; namely by multiplying each of the coefficients β by either (1)-a common factor F_c, or (2)-differential factors F_i. From the point of view of the spectral shape, the latter approach preserves the characteristic of the system impulse response, and should prove to be perceptually better than the former.

Using the common factor F_c to rescale each of the coefficients β_i until their SOM equals unity, the resulting prediction gain, when compared to the gain obtained by using the original unstable coefficients, drops by about 0.18 dB for 2-tap, and by about 0.87 dB for 3-tap.

Switching to the use of the differential factors F_i, we noticed a slight improvement in the resulting prediction gains; but the improvement for both cases (2- and 3-tap) averages only 0.03 dB. Thus, despite of the ease in the computation of F_i, this marginal improvement on the prediction gain is not a sufficient enticement to make the use of F_i a definite preference over F_c. Nevertheless, we have shown that F_i gives a slightly better improvement than F_c at least in terms of the prediction gain. Further tests on the output in the next chapter may present a more complete picture concerning the actual effect of F_i on the quality of the coded output.

In Table 4.3, we indicate the cost of the stabilization in multi-tap system, where SNR is the optimal prediction gain before stabilization. SNR(1a) and

SNR(1b) indicate the prediction gains after stabilization by unity replacement using constant and differential factors. The slight advantage in using F_i is shown in terms of the slightly higher prediction gain in column SNR(1b).

Prediction Gains (2-Tap)			
Speech File	SNR	SNR(1a)	SNR(1b)
CATM8	3.22	3.18	3.18
PB1M1	4.14	4.02	4.01
DOUG5	5.14	4.83	4.87
PIPM8	3.81	3.73	3.74
PB1M5	4.44	4.06	4.22
Male Average:	4.15	3.97	4.01
CATF8	6.68	6.60	6.60
PB1F1	9.02	8.82	8.88
VOICF5	8.98	8.57	8.60
PIPF8	9.52	9.45	9.45
PB1F5	8.55	8.44	8.45
Female Average:	8.55	8.38	8.40
Total Average:	6.35	6.18	6.21

Prediction Gains (3-Tap)			
Speech File	SNR	SNR(1a)	SNR(1b)
CATM8	3.34	2.69	2.69
PB1M1	4.27	3.81	3.82
DOUG5	5.25	4.35	4.38
PIPM8	3.95	3.05	3.06
PB1M5	4.59	3.78	3.89
Male Average:	4.08	3.54	3.57
CATF8	6.82	6.06	6.60
PB1F1	9.22	7.94	8.00
VOICF5	9.08	7.76	7.79
PIPF8	9.64	8.14	8.14
PB1F5	8.69	7.52	7.53
Female Average:	8.69	7.48	7.50
Total Average:	6.39	5.51	5.53

Table 4.3 Prediction gains for 2- and 3-tap filters

Chapter 5 Effects of Stabilization

The purpose of this chapter is to examine the effect of the instability of the pitch synthesis filter and the consequent improvement of the output speech as a result of the stabilization process. The CELP system described in Chapter 2 is used for the experimental tests carried in this chapter. In this study, the stabilization means a modification of the unstable coefficient(s) of the predictors in the analysis and the synthesis.

The computation required to select the optimal residual model in CELP is rather time consuming; it is unnecessarily wasteful to use it in the initial stage of the investigation. Therefore, a rough model is used for experimental purpose. The residual signal generated by the CELP analyzer has been shown to have a Gaussian density distribution [ATAL(85)]. Using the central limit theorem, a noise generator is used to produce a random signal r_n with Gaussian density distribution. Hence, the experimental residual model is simply a Gaussian noise with a segmental energy equal to that of the true residual. The coarse residual model used in the present study differs from the actual model in that, instead of being the optimal random signal which has the closest match in energy level to the residual, it is a random signal completely uncorrelated to the residual. Nevertheless, we will see that even with this rough model, the resemblance of the

resulting synthesized output to the original signal is amazingly high.

5.1 Preliminary Test

Using the above described crude residual model, the decoded output reflects only the general envelope (quality) of the input speech. The changes which result from the stabilization of the pitch synthesis filter should also be taken as indicative, rather than the actual responses. Nevertheless, these changes would give a global perspective of what is to be expected when the actual optimal model is used. In Section 5.2, we will use the true model as used in the actual coder to verify the results obtained in this section.

From the simulation tests, we observe that the envelope of the output generated by the synthesis filter approaches the envelope of the input speech. But careful examination shows that the output is quite random and it lacks the pitch characteristics. The corresponding spectrogram also reveals that despite of the proper distribution of the energy, the output in response to the crude model lacks the well-defined formant bands and the pitch striations which are present in the input speech.

In the following, we will first study the distortion due to the instability using 1-tap pitch synthesis filter, and then extend the study to using 3-tap filter.

5.1.1 Type(I) Degradation

Leaving the pitch synthesis filter unstabilized, one symptom of degradation in the output speech is a sharp burst of energy as shown in frame #7 in Figs. 5.1.(b),(c). The magnitude of the burst, or the seriousness of the degradation depends essentially on: (1)-the magnitude of the coefficient β, and (2)-the

level of input speech energy in the corresponding frame. We note that the distortion is always proportional to the magnitude of β, and is significant only when the coefficient magnitude exceeds 1.25. Experimental evidence indicates that the energy burst caused by the instability also depends on the position of the unstable frame in the speech file, i.e., if it occurs at a segment where the energy level is low, the effect is not as severe as if the frame energy level is high, such as in a voiced segment. The reason for this is that in an adaptive filter where the instability lasts only momentarily, it requires a substantial initial energy to prompt a large response during the moment the filter is unstable.

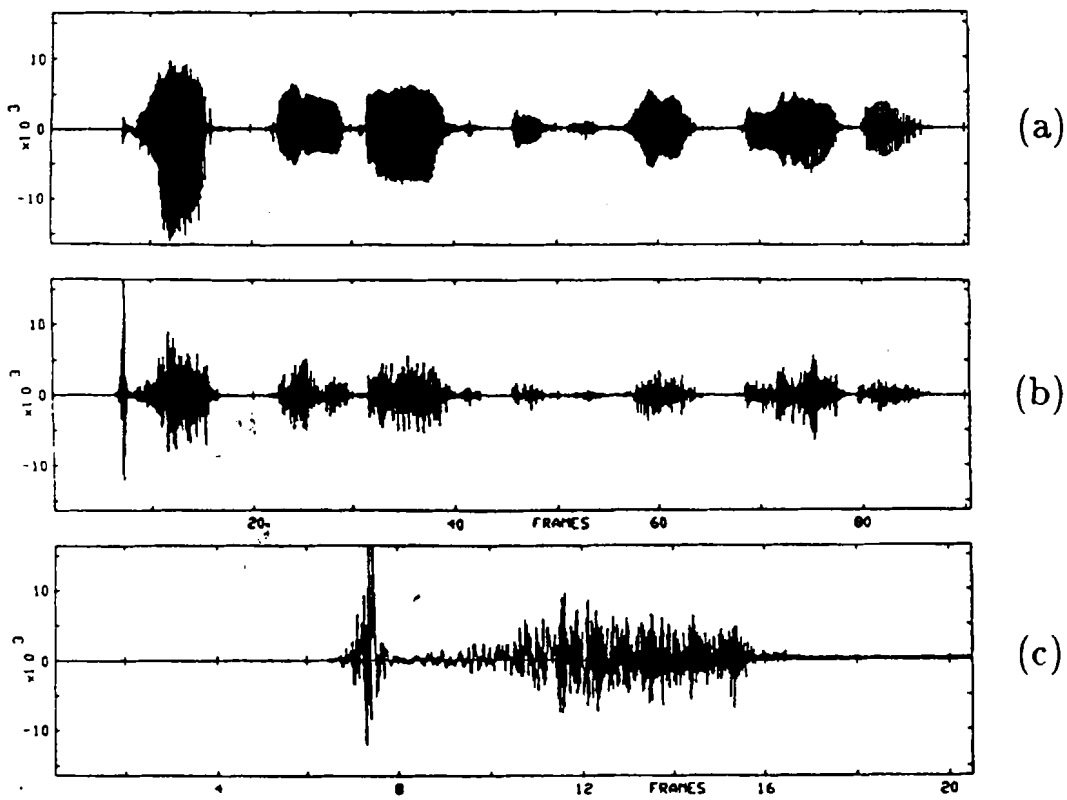


Fig. 5.1 **(a)** Original input speech,**(b)** Unstable output speech, **(c)** Close-up view of the instability at frame#7.

Our studies also show that reducing the unstable coefficient value of an unstable frame has a large effect in removing the above described distortion on the speech output. Fig.5.1 shows an original speech 'CATF8' in (a) along with the unstabilized output in (b). The three unstable frames in the output speech occur at frames #7, #41 and #68, with respective coefficient values of 3.518, 1.304 and 1.048. The high coefficient at frame #7 results in an impulse-type distortion as depicted in (c). To study the relationship between the magnitude of the coefficient and the degree of the degradation, we reduced the coefficient value of frame #7 in steps of 0.5. As the magnitude of the coefficient was gradually suppressed through 3.5, 3.0, 2.5, 2.0, 1.5 and 1.0, we observed the distortion diminishing accordingly. Fig. 5.2 records the snapshots of the responses at various values of β. The distortion seems to have been eliminated at β=1.0, when the pitch synthesis is marginally stabilized.

We proceeded to examine the distortion due to unstable synthesis filter and the effect of its stabilization from the view point of the segmental energy level. Comparing against the energy level of the input signal, we find in most cases that when a frame is unstable in isolation (i.e., when it is imbedded by stable frames), not only the energy level of that unstable frame rises, but those of the subsequent frames are also affected as a result of the past memory. The number of subsequent frames affected generally depends on the magnitude of the unstable coefficient, the property of the speech segment during the unstable frame as well as the frame size. But regardless of these factors, the frame immediately following the unstable frame is almost always influenced.

5.1.2 Type(II) Degradation

Another type of degradation observed in the study is characterized by a grad-

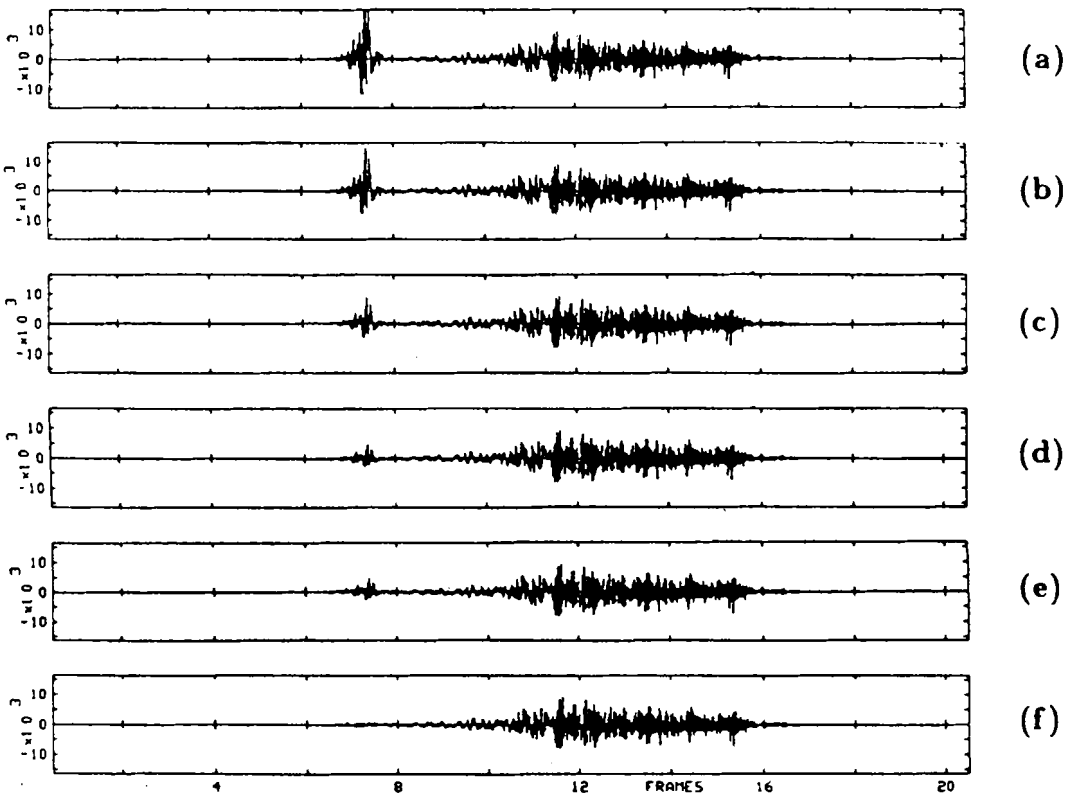


Fig. 5.2 Diminishing distortion in response to decreasing unstable coefficient

ually growing energy, and is caused by several consecutive unstable frames. We used the speech file 'TOMF8' for experimental observation. Our approach is to intentionally destabilize the frames which follow an isolated unstable frame, and then examine the effect after each destabilization. With 1-tap filter, 'TOMF8' has only three unstable frames as shown in Table. 5.1. The portion of the speech file which contains the unstable frame #47 is shown in Fig. 5.3, where (a) is the original input segment, and (b) is the unstable output due to a single unstable frame #47 with coefficient 1.575. The coefficient magnitude is relatively high during this frame; but the small energy of the input signal in that segment does not cause a large degradation. When we destabilized the next frame (#48) by raising its coefficient to 1.0, although being only marginally unstable, the dis-

tortion contributed by this second consecutive unstable frame is quite significant as shown in (c). The introduction of a third consecutive (marginally) unstable frame (#49) generated even larger distortion as shown in (d). Should the unstable frame have occurred at a higher energy segment, the distortion from the above experiment would have been more severe than the one in the current example. In reality however, consecutive unstable frames are not prevalent in 1-tap system; it is a more common sign in a higher-order (e.g., 3-tap) filter, which we will investigate next.

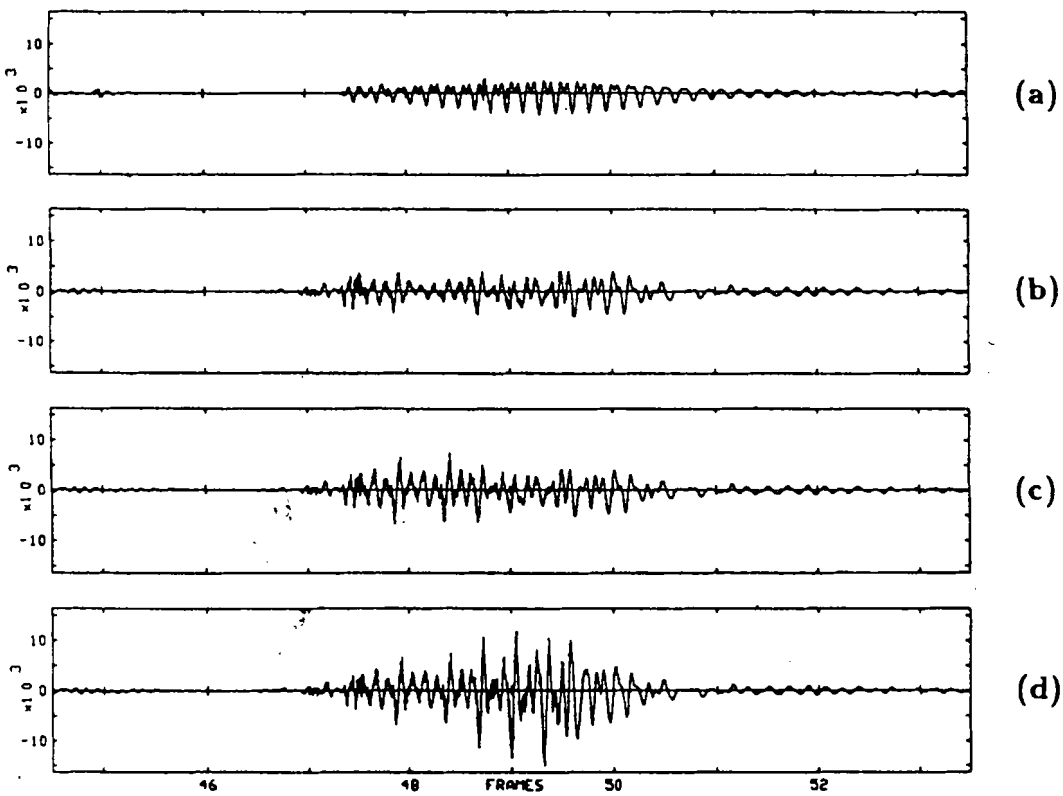


Fig. 5.3 Illustration of distortion caused by three consecutive unstable frames

Since more unstable frames occur in a system using multi-tap filter than in one using single-tap filter. Consequently, the distortion is more abundant in 3-tap

filter than it is in the single-tap case. In the study, we observed that the type(II) degradation, which results from a series of unstable frames, is more common than the type(I) distortion in the 3-tap filter. The effect of type(II) degradation is also more damaging, because when unstable frames occur in this manner, the speech samples in the second frame are synthesized from the unstable output of the first frame, whereas the samples in the third frame are in turn constructed from the even more energetic output of the second frame. This process leads to an accelerated rate of 'explosion' or energy build-up, which is normally perceived as the type(II) distortion. Our tests confirm that the stabilization process does not have a substantial effect in correcting this type of distortion.

To summarize the findings of this preliminary study, we showed in 1-tap filter that type(I) distortion could be suppressed to a large extent by the unity replacement method. Using the second method, in which the coefficient is further reduced to a critical point†, we observed a slightly better result in terms of less distortion when compared to that generated by the first method. This observation may appear to be in contradiction with the one made earlier in Section 4.3, where it was shown that the unity replacement (Table 4.1, SNR(1c)) is more optimal than the reciprocal replacement (Table 4.1, SNR(2)). But we should note that the criterion used in Section 4.3 was to minimize the residual, while the criterion used in the present study is to minimize the actual distortion to the coder output, which is perceptually more significant. For better perception, method(2) which replaces $\beta > 1$ with its reciprocal is the better stablization scheme.

In the 3-tap case, we found that type(II) degradation was more common, and that this type of degradation had more resistance to suppression. Our ex-

† The reciprocal of the unstable coefficient.

perimental studies also indicate that using differential factors F_i as opposed to constant factor F_c in reducing SOM to unity generates a better quality in the decoded speech.

5.2 Test Using Optimal Residual Model

The observations in this section are based on tests using the optimal residual model as used in the CELP coder. This model, compared to the rough model used in the previous section, is much cleaner and therefore is a much better representation of the actual residual signal. Consequently, the resulting coded speech is closer to the input than the one constructed from the crude residual model.

But generally speaking, most of what we observed in the previous section — in terms of distortions, responses to stabilization processes with various algorithms in the last section — are still true in the current test. One noted difference may be the general quality of the coded signal, in that the output using the optimal residual model is more refined, less random, and has a stronger pitch characteristics than the one generated from the random noise. In this section, we provide some concrete examples of the degradations caused by the instability of the pitch synthesis filter, as well as some evidence of the possible improvements from the stabilization process.

A negative aspect brought by the incorporation of the pitch filter is the speech degradation during frames where the pitch synthesis filter is unstable. As observed in the previous section, the distortion can be separated into two categories. As described earlier, the type(I) distortion has a more dramatic effect but is easily corrected; whereas the effect of the type(II) distortion is less severe but it is more stubborn.

For the 3-tap synthesis filter, the stabilization requires the reduction of the SOM of the three coefficients to unity, either with a common factor or with three differential factors — one for each coefficient. Despite of the fact that the differential factors F_i differ only marginally from the constant factor F_c, our tests indicate that using the differential factors to reduce the coefficients produces a slightly better improvement in the coded speech signal. The improvement can be seen from the less deviation in the energy levels between the input and output signals, or the higher degree of resemblance between the two time waveforms.

Frame No.	β_1(1-tap)	SOM (3-tap)
9	(stable)	1.082
10	1.031	1.160
11	(stable)	1.090
12	(stable)	1.222
13	(stable)	1.229
24	(stable)	1.786
25	(stable)	1.511
26	1.070	1.429
27	(stable)	1.137
28	(stable)	1.375
40	(stable)	1.047
41	(stable)	1.227
42	(stable)	1.079
43	(stable)	1.012
47	3.590	7.078
55	(stable)	1.079
62	(stable)	3.734
67	(stable)	1.252
71	(stable)	1.250

Table 5.1 Unstable frames in speech file 'TOMF8' using 1-tap and 3-tap filters.

Figs.(5.4) to Figs.(5.7) illustrate what we have discussed thus far. The speech file 'TOMF8' is again used to test cases in which 1-tap or 3-tap filters are employed, and Table 5.1 contains the frame locations and the corresponding coefficients where the filters are unstable. The two types of distortions previously discussed are present in the speech file 'TOMF8'. Using 1-tap filter, and referring to Fig.5.4, (a) displays the original input waveform whereas (b) is the unstable output with distortion. Note again that although the unstable coefficient at frame #47 is extremely large ($\beta = 3.590$), the resulting distortion is not very serious because the unstable frame occurs at the speech segment where the signal energy is low. The coded outputs with stabilized synthesis filters are shown in (c) and (d), where they correspond to stabilization processes using unity- and reciprocal-replacement methods respectively. We see in this case a better improvement in the output waveform on (d) due to the second method, which replaces the unstable coefficient with its reciprocal value.

The segmental energy levels corresponding to the waveforms in Fig.5.4 are shown in Fig.5.5. It shows the segmental energy difference between the input and the output waveforms before [in (b)] and after [in (c),(d)] the stabilization of the pitch synthesis filter. Note that some of the observed energy level distortions in the figure are actually due to the courseness of the residual model. Only at frame #47 and the subsequent frames (see Table 5.1) that the energy difference is the result of instability before and after stabilization.

In a similar fashion, Fig.5.6 illustrates and compares the output waveforms before and after stabilization process in the 3-tap case. When compared to the original signal in (a), the unstable output in (b) is greatly distorted, especially around frame #26 where there is a series of unstable frames (see Table. 5.1), and on frame #62 in which the SOM of the coefficients is large (SOM=3.734).

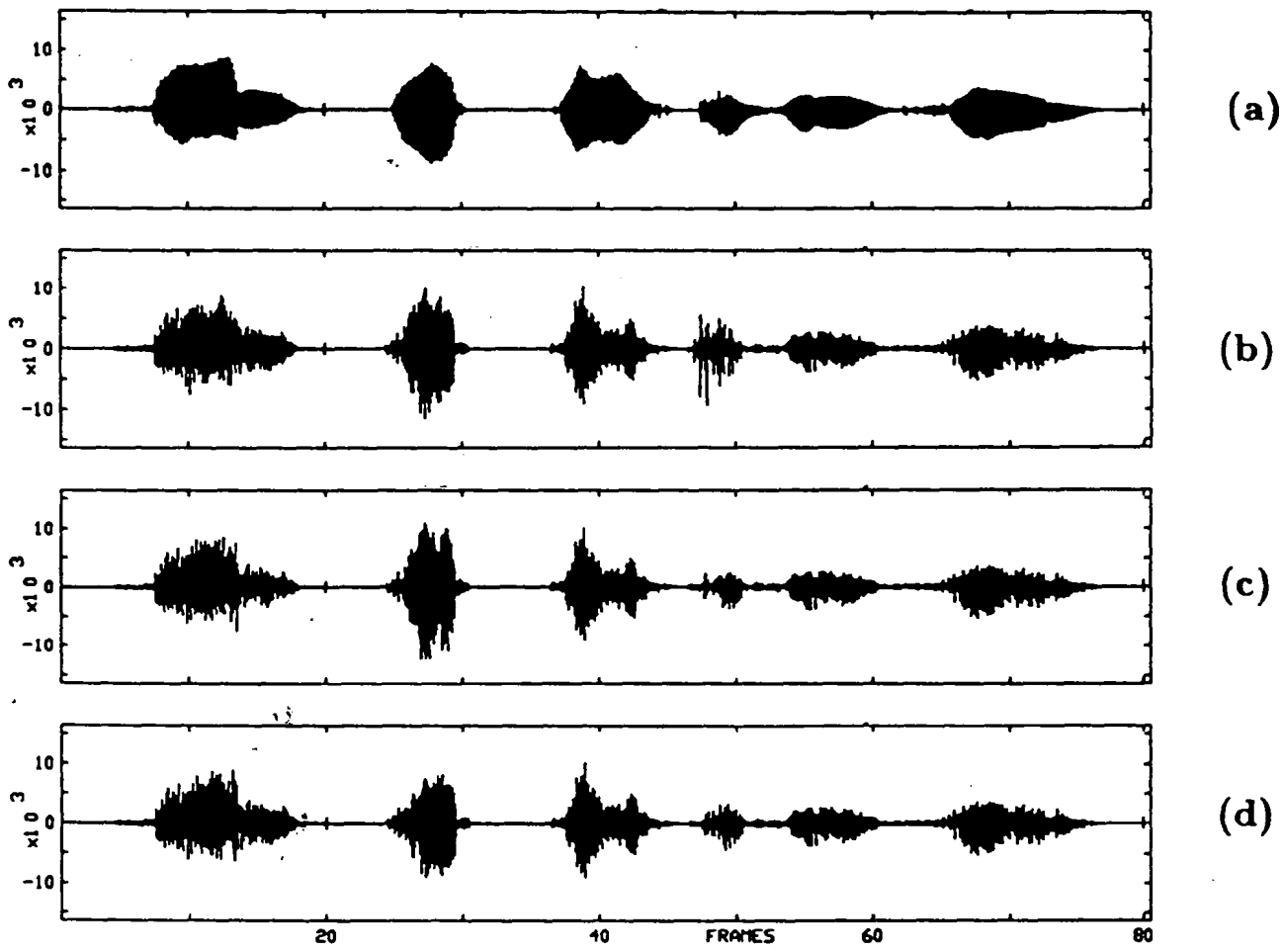


Fig. 5.4 Speech waveforms 'TOMF8': (a)original input, (b)unstable output with distortions (using 1-tap pitch predictor), (c)stabilized output using unity-replacement method, (d)stabilized output using reciprocal replacement method.

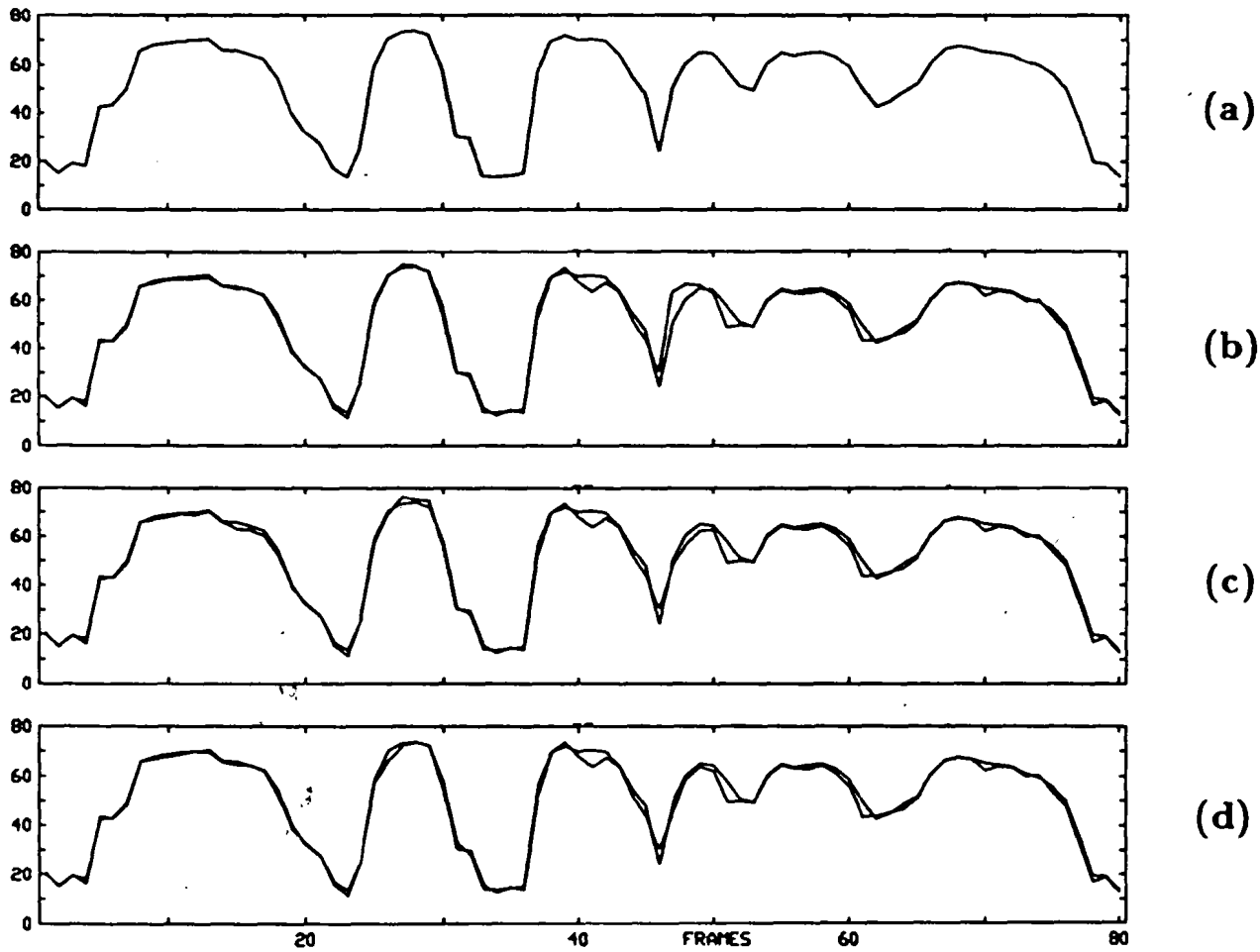


Fig. 5.5 Energy levels corresponding to figure 5.4

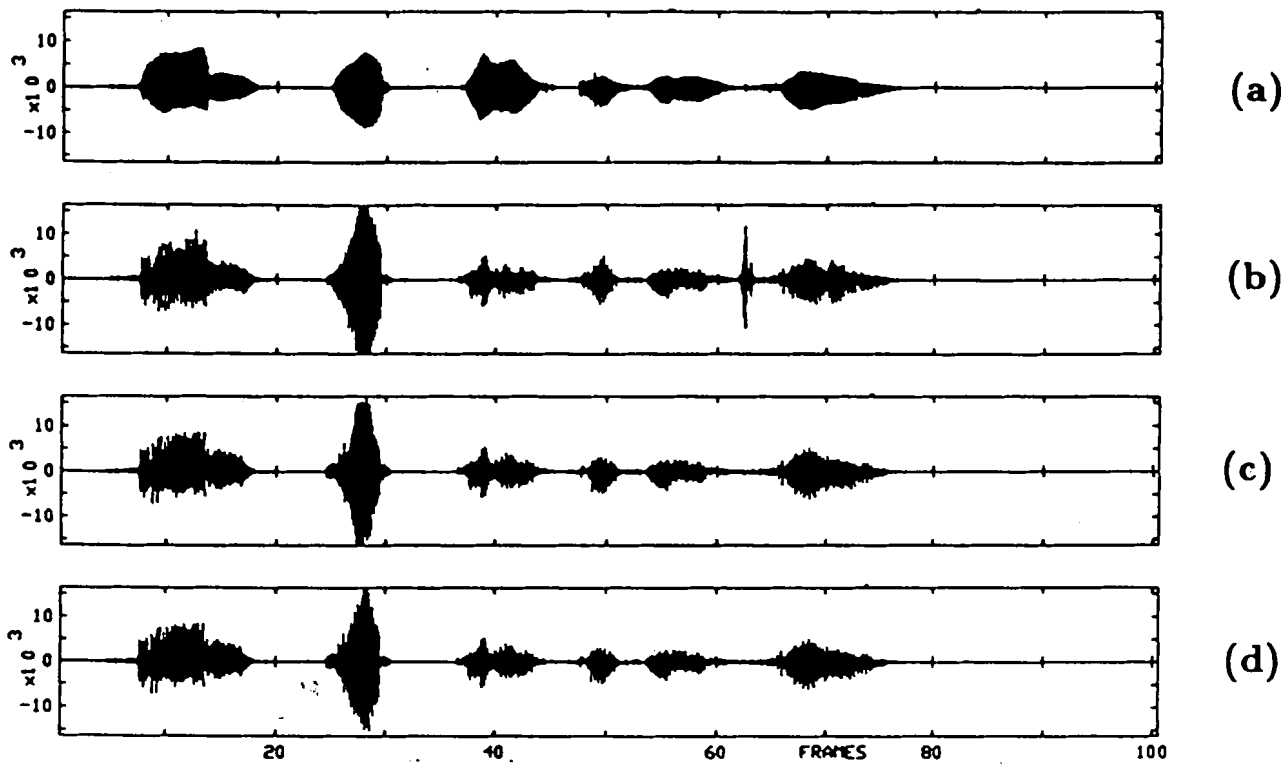


Fig. 5.6 Speech waveforms 'TOMF8': (a)original input, (b)unstable output with distortions (using 3-tap pitch predictor), (c)stabilized output using common factor, (d)stabilized output using differential factors.

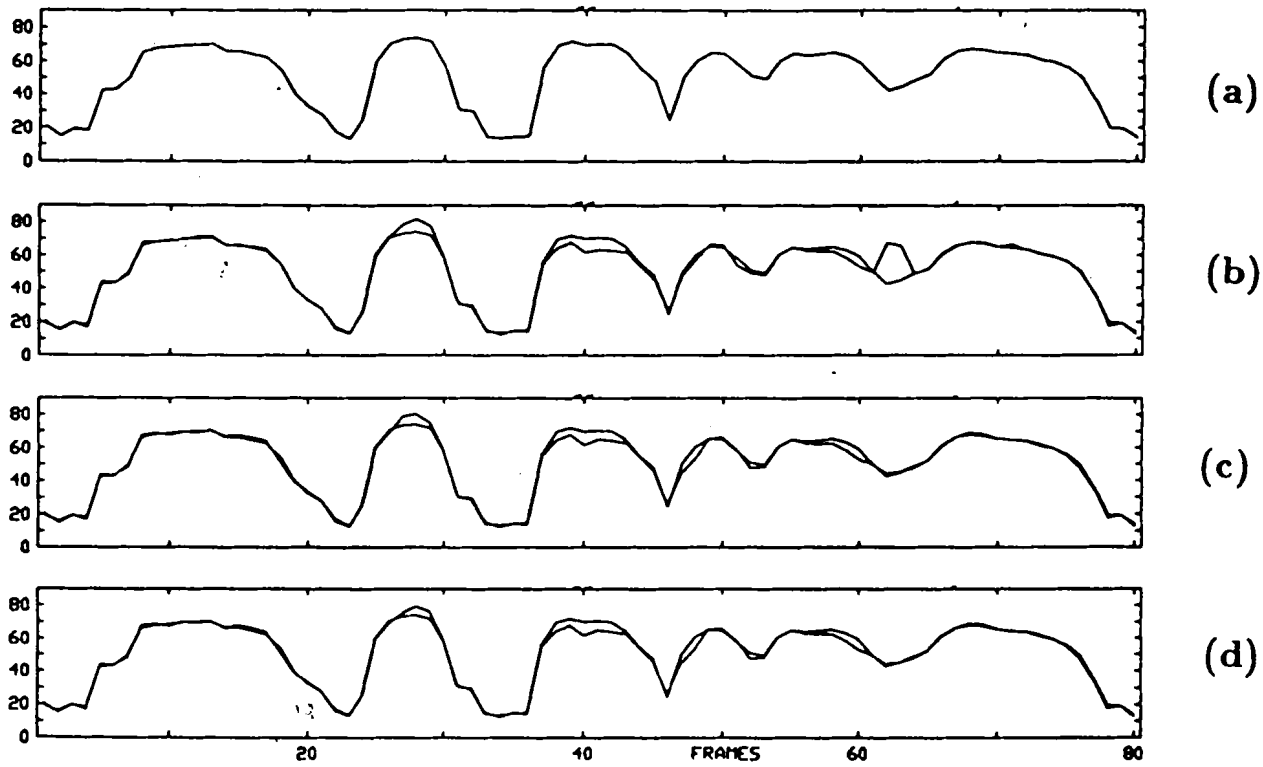


Fig. 5.7 Energy levels corresponding to figure 5.6

Again, just as in the previous 1-tap case, the very large SOM at frame #47 (SOM=7.078) hardly disturbs the output since the signal energy (consequently the residual energy) in this frame is quite low.

Two types of distortions exist in this case. The first type, which is in the form of a sudden outburst of energy, occurs in frame #62; while the other type — a gradual noise build-up, is seen between frames #24 and #28. Upon stabilization using common factor [in (c)] and utilizing the differential factors [in (d)], we notice the disappearance of the energy spike at frame #62, but only a slight attenuation of the distortion between (frames #24 to #28). In this particular speech file, the stabilized outputs using the first two schemes do not seem to show noticeable difference. But during the course of the experimental test, the second scheme which uses differential factors generally gives a better result than using a constant factor. Fig.5.7 displays the energy levels (with respect to the original energy level) which correspond to the files in Fig.5.6.

5.3 Perceptual Test

Other than examining the effects of stabilizing the pitch synthesis filter upon the output speech by observing the time waveforms, we also tested the the effect perceptually by listening to the output waveforms before and after the stabilization. We paid a special attention to differentiate the perceptual distortions corresponding to the two types of degradations described earlier, and examined the effect of the stabilization on the distortion.

Generally speaking, the more abrupt the distortion in terms of increase in the energy level, the more easily perceptible is the degradation in the output speech. Thus the type(I) distortion, which takes the form of a sharp burst of energy level (as in Fig. 5.6, frame #62), is more easily heard than the type(II) distortion as

shown around frame #26 on the same figure. Listening† to the waveforms in Fig.5.4 and Fig.5.6 in sequence, and comparing the unstable waveforms with the original and with the stabilized outputs, the type(I) distortion was perceived as a distinct 'click' sound in the output speech. This has a similar effect to the sound produced by a gramophone as the stylus traces a damaged groove on the disc. The perceptual degradation due to type(II) distortion was not easily noticed, partly because of the gradualness of the energy growth, and partly because of the relatively large coding noise which tends to mask the distortion.

Examining the stabilized output waveforms, it shows that the type(I) distortion is always cleanly removed; whereas the type(II) distortion, which affects its subsequent neighbouring frames, is more persistent and harder to eliminate.

In both 1-tap and 3-tap cases, the major improvement upon stabilization occurs at the scattered segments where the distortion is caused by an isolated instability (type(I)); as for the degradation due to the second type of instability — a gradual increase in energy level, the stabilization does alleviate it but does not suppress it to a substantial degree. But on the whole, the evidence shows that stabilizing the pitch synthesis filter indeed improves the overall quality of the coded output by eliminating the annoying 'clicks' in the output speech.

5.4 Applicability of Test Results to APC System

The tests we have just carried out should be equally applicable to the APC system under certain valid assumptions. In APC system, if we place the quantizer outside the analyzer, it can be shown that the output of the quantizer is the sum of (1)-the unquantized residual and (2)-the quantization error (see Fig. 5.8). The

† All listening tests were done in a sound-proof environment.

linear synthesis system driven by the quantizer output consequently generates an output which consists of (1)-the input itself due to the unquantized residual, and (2)-the filtered quantization noise. The following analysis verifies the separability of the above APC output.

Referring to Fig. 5.8 and assuming, for simplicity, that only one-tap pitch synthesis filter is involved, the input s_n and output y_n of the system can be written as[†]:

$$s_n = \tilde{s}_{ni} + e_n \tag{5.1}$$

$$\begin{aligned} y_n &= \tilde{s}_{no} + \hat{e}_n \\ &= \tilde{s}_{no} + e_n + q_n \end{aligned} \tag{5.2}$$

where:

$\tilde{s}_{ni} = predicted\ signal\ in\ the\ analysis$

$\tilde{s}_{no} = predicted\ signal\ in\ the\ synthesis$

$e_n = residual\ signal$

$q_n = quantization\ noise$

Denoting $f[\cdot]$ as the filtering operation of the pitch synthesis filter and assuming linear operation, the output of the system is

$$\begin{aligned} y_n &= f[\hat{e}_n] \\ &= f[e_n + q_n] \\ &= f[e_n] + f[q_n] \end{aligned} \tag{5.3}$$

[†] Neglecting channel error.

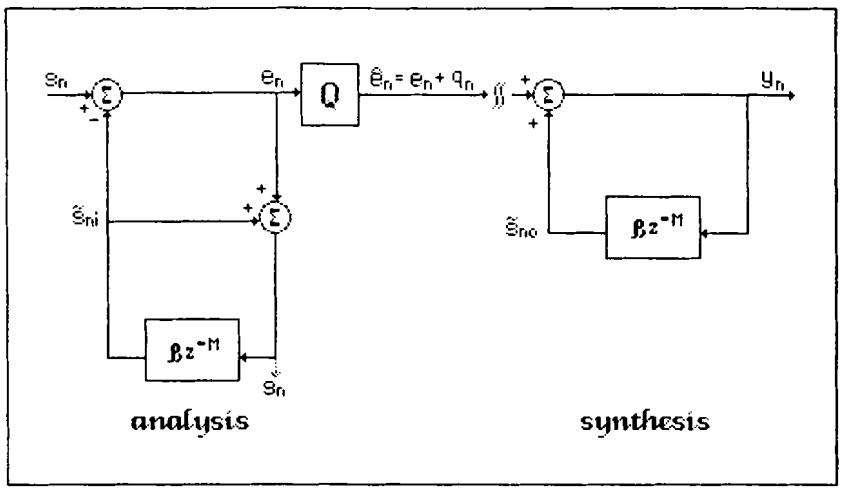


Fig. 5.8 Modified APC Coder.

From the analysis stage in Fig. 5.8, $f[e_n]$ is simply the input signal s_n. Therefore,

$$y_n = s_n + f[q_n]$$

or

$$y_n - s_n = f[q_n] \tag{5.4}$$

In the absence of channel error, the output due to the unquantized residual e_n gives rise to the input signal s_n, leaving the quantization noise q_n the only source of distortion. Hence, the effect of stabilizing the synthesis filter on the APC system output is essentially the same as the effect on the quantization noise alone.

In the preliminary test described in Section 5.1, we used a Gaussian distributed random noise to model the residual signal. Theoretically (and it has been shown experimentally), the quantization noise in APC can equally well be simulated by the same model used in the previous tests for CELP. The only required modification is to reduce the noise energy to such a level that it properly

models the quantization noise. This can be accomplished by either (1)-using the properly scaled original crude model as used in Section 5.1, in which case the quantization noise is assumed to be Gaussian distributed, or (2)-using properly scaled random number generator output which will yield a desired $S_r N_q R^{\dagger}$. In either case, the outcome has been observed to be quite similar to what we had observed previously when testing the stabilization effect on CELP, except that the output now represents the filtered noise $f[q_n]$, instead the decoded speech output as in the case of CELP.

In general, the effect of the stabilization of the pitch synthesis filter on the quantization noise in the above described APC is essentially the same as that observed on the residual in CELP, hence all the observations made in the earlier sections when testing the CELP are applicable to the APC system as well. Since Eq. (5.4) indicates that the filtered quantization noise $f[q_n]$ is the difference between the input and the distorted output signal, the removal of the distortion to $f[q_n]$ means an improvement in the quality of output signal.

---

† Residual signal to quantization noise ratio.

Chapter 6 Conclusions

We have established, in this study, the sufficient condition for the stability of the pitch synthesis filter $H_p(z)$; it states that for a set of the pitch predictor coefficients $\{\beta_i,\ i = 1,2,...m\}$, the stability of the pitch synthesis filter is guaranteed if the sum of the magnitude of all the coefficients is less than unity, i.e., $\sum_i |\beta_i| \leq 1$. The boundaries of the sufficient conditions for one, two and three tap filters are respectively a straight line, a diamond and the 8-faced figure shown in Fig. 3.5, while those of the necessary conditions have been numerically derived, and verified by Jury's criterion to converge rather quickly with increasing pitch lag.

Comparing the necessary condition and the sufficient condition boundaries, the two conditions for the single-tap filter are the same. But for multi-tap filters and at large lag and for $M > 10$, only approximately half of the regions defined by the two conditions match each other. For the remaining regions, the region defined by the necessary condition always exceeds that defined by the sufficient condition. As $M \rightarrow \infty$, the stability region as defined by the SOM-criterion (sufficient condition) for 2-tap filter approaches a perfect diamond (Fig. 3.4), and the 3-tap stability region has the shape depicted in Fig. 3.8 where an anti-symmetry exists between the upper and the lower regions. Because of the irregular shapes,

the necessary conditions are difficult to describe and to generalize mathematically. We use Jury's stability criterion to generate the approximate models for the necessary conditions for 2-tap and 3-tap pitch synthesis filters, with which the deviation between the sufficient condition and the necessary condition with the number of taps as a parameter is computed. The deviation is expressed in terms of the number of unstable frames over-estimated by the crit(SOM) over the total number of frames in a speech file in percentage. But based on the statistics for one, two and three-tap filters, the Sum-of-Magnitude criterion in Eq. (3.20) is justified to be a reliable algorithm for testing the stability of the pitch synthesis filter.

As measures to correct the instability, we proposed several methods of stabilization. The algorithms available to the 1-tap system include: the replacement of the unstable coefficient by (1) $\alpha(\tau = M)$, the normalized correlation sum evaluated at lag M, (2) unity and (3) $\frac{1}{\beta}$. Using $\alpha(\tau = M)$ to replace β unconditionally is feasible because these two quantities approach each other closely except when $\beta > 1$, and $\alpha(\tau = M)$ is always less than unity. This scheme is the simplest since there is no need to check for the system stability; however, the price in terms of loss in prediction gain is relatively significant as shown in Table 4.1 [SNR(1a)]. To minimize the loss in prediction gain, a selective scheme which replaces the coefficient only when it exceeds unity is to be used. The other two methods of stabilization substitute the unstable β with 1 and $\frac{1}{\beta}$ respectively. Purely judging these various schemes in terms of the accompanied loss in prediction gain, our tests showed that using the unity replacement scheme involved the minimum loss in prediction gain, followed by the reciprocal replacement scheme, and then the correlation coefficient replacement scheme (selective and unconditional) in that order.

In stabilizing a multi-tap system ($m = 2, 3$) using the unity replacement, we investigated the benefit of using differential factors F_i to scale the predictor coefficients. Theoretically, down scaling the coefficients by a common factor distorts the system function. In order to preserve the spectral property of the system, we must replace z by az where $0 < a < 1$. According to the analysis in Section 4.4.2, this is equivalent to down scaling each coefficient by a different factor F_i, which can be easily computed by linear interpolation method. Tests show that using F_i results in a slightly better prediction gain and a better perception compared to when the constant factor F_c is employed.

It was found that the instability of the pitch synthesis filter led to two types of distortions. The type(I) distortion (due to a single isolated unstable frame) occurs in the form of an impulse, and the type(II) distortion (caused by a series of unstable frames) results in a gradual growth of noise in the output signal. The impulse-like distortion occurs in both the single-tap and the multi-tap systems, but the second type of distortion is more common in a multi-tap system. Perceptually, the impulse-like distortion is easily perceived as an annoying 'click' sound, whereas the noise build-up type of distortion is harder to distinguish from the background noise. Upon stabilization process, the type(I) distortion can be easily suppressed and eliminated almost completely; whereas the type(II) distortion is stubborn, and the stabilization process does not suppress it to a substantial degree. Nevertheless, the perceptual test indicates that the stabilization of the pitch synthesis filter improves the overall quality of the decoded speech, therefore it should be an essential part of the coding algorithm in Adaptive Predictive Coding of speech.

Appendices

Appendix A. The Stability of Formant Synthesis Filter

For a p^{th}-order formant predictor, the corresponding inverse filter can be expressed as

$$A_f(z) = 1 - \sum_{i=1}^{p} \alpha_i z^{-i} \tag{A.1}$$

or

$$= k \prod_{i=1}^{p} (1 - \gamma_i z^{-1}) \tag{A.2}$$

where k in Eq. (A.2) is a combined gain factor.

Assuming that the instability of $H_f(z) = \frac{1}{A_f(z)}$ is caused only by a single root $z = \gamma_o e^{j\theta_o}$ outside the unit circle, the inverse filter can be expressed as

$$A_f(z) = B(z)(1 - \gamma_o z^{-1}), \tag{A.3}$$

which is a cascade of the stable (minimum phase) filter $B(z)$ and the unstable first order filter $(1 - \gamma_o z^{-1})$.

The output of the inverse filter $A_f(z)$ is the residual signal e_n with the following z-transform

$$\begin{aligned} E(z) &= S(z)A_f(z) \\ &= S(z)B(z)(1 - \gamma_o z^{-1}) \\ &= C(z)(1 - \gamma_o z^{-1}). \end{aligned} \tag{A.4}$$

$S(z)$ is the z-transform of the input signal s_n, and $B(z)$ generates an intermediate output c_n, which when subsequently filtered by $(1 - \gamma_o z^{-1})$ gives the residual e_n.

In terms of c_n, the residual can be expressed according to Eq. (A.4) as

$$e_n = c_n - \gamma_o c_{n-1}, \tag{A.5}$$

from which the residual energy becomes

$$\begin{aligned} E &= \sum_n |e_n|^2 \\ &= \sum_n |c_n - \gamma_o c_{n-1}|^2 \\ &= \sum_n (c_n - \gamma_o c_{n-1})(c_n - \gamma_o c_{n-1})^*. \end{aligned} \tag{A.6}$$

Substituting $\gamma_o = r_o e^{j\theta_o}$ into Eq. (A.6),

$$\begin{aligned} E &= \sum_n [c_n - (r_o e^{j\theta_o}) c_{n-1}][c_n^* - (r_o e^{-j\theta_o}) c_{n-1}^*] \\ &= \sum_n c_n c_n^* - r_o (e^{j\theta_o} c_n^* c_{n-1} + e^{-j\theta_o} c_n c_{n-1}^*) + r_o^2 c_{n-1} c_{n-1}^* \\ &= \sum_n |c_n|^2 - 2 r_o Re\{e^{j\theta_o}(c_n^* c_{n-1})\} + r_o |c_{n-1}|^2 \end{aligned} \tag{A.7}$$

The derivative of E with respect to r_o is

$$\frac{\delta E}{\delta r_o} = 2 r_o \sum_n |c_{n-1}|^2 - 2 Re\{e^{j\theta_o}(c_n^* c_{n-1})\}. \tag{A.8}$$

According to Cauchy-Schwartz inequality, we have

$$Re\{e^{j\theta_o}(c_n^* c_{n-1})\} \le \Big[\sum_n |c_n|^2\Big]^{\frac{1}{2}} \Big[\sum_n |c_{n-1}|^2\Big]^{\frac{1}{2}}. \tag{A.9}$$

Let

$$F = \Big[\sum_{n=L}^{U} |c_{n-1}|^2\Big]^{\frac{1}{2}} \tag{A.10}$$

where L and U now represent the limits of summation. Eq. (A.10) implies that

$$\begin{aligned} F^2 &= \sum_{n=L}^{U} |c_{n-1}|^2 \\ &= |c_{L-1}|^2 + |c_L|^2 + |c_{L+1}|^2 + ... + |c_{U-1}|^2 \\ &= |c_{L-1}|^2 + \sum_{n=L}^{U} |c_n|^2 - |c_U|^2 \end{aligned} \tag{A.11}$$

Using the relation in Eq. (A.9), together with Eq. (A.10) and Eq. (A.11), the residual energy differential in Eq. (A.8) can be written as

$$\begin{aligned}\frac{\delta E}{\delta r_o} &\geq 2r_o \sum_{n=L}^{U} |c_{n-1}|^2 - 2\Big[\sum_{n=L}^{U} |c_n|^2\Big]^{\frac{1}{2}} \Big[\sum_{n=L}^{U} |c_n|^2\Big]^{\frac{1}{2}} \\ &= 2r_o F^2 - 2\Big[F^2 - |c_{L-1}|^2 + |c_U|^2\Big]^{\frac{1}{2}} F \end{aligned} \tag{A.12}$$

By *autocorrelation* method, the energy is minimized over the entire range from $L = 0$ to $U = \infty$. The causality and the finite energy respectively require that

$$c_{L-1} = c(-1) = 0 \tag{A.13}$$

$$c_U = c(\infty) = 0 \tag{A.14}$$

In conforming to the above constraints, Eq. (A.12) is finally reduced to:

$$\begin{aligned}\frac{\delta E}{\delta r_o} &\geq 2r_o F^2 - 2\Big[F^2 - 0 + 0\Big]^{\frac{1}{2}} F \\ &= 2F^2(r_o - 1) \end{aligned} \tag{A.15}$$

The above relation clearly states that if the residual energy is minimized such that

$$\frac{\delta E}{\delta r_o} = 0,$$

then r_o must not exceed unity. This implies that $A_f(z)$ is minimum phase and therefore $H_f(z)$ must be stable.

Appendix B. Derivation of the Sufficient Condition

The characteristic equation for 2-tap filter is

$$F(z) = z^{M+1} - \beta_1 z - \beta_2 = 0 \tag{B.1}$$

Or equivalently, it can be expressed as

$$z^{M+1} = \beta_1 z + \beta_2 \tag{B.2}$$

$$= z^{\frac{1}{2}}(\beta_1 z^{\frac{1}{2}} + \beta_2 z^{-\frac{1}{2}}) \tag{B.3}$$

On the unit circle where $z = e^{j\theta}$, Eq. (B.3) can be written as

$$e^{j\theta(M+1)} = e^{j\frac{\theta}{2}}[(\beta_1 + \beta_2)cos(\frac{\theta}{2}) + j(\beta_1 - \beta_2)sin(\frac{\theta}{2})] \tag{B.4}$$

Using the result derived in Section 3.4.1, for stability, the roots of the right hand side of Eq. (B.2) must be confined inside the unit circle. Therefore, taking the magnitudes of Eq.(B.2), and using the following substitutions

$$a = |\beta_1 + \beta_2|$$

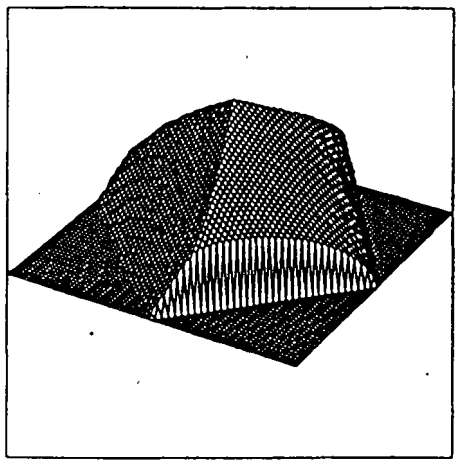
$$b = |\beta_1 - \beta_2|,$$

The right hand side of Eq. (B.4) is equivalent to, and must satisfy

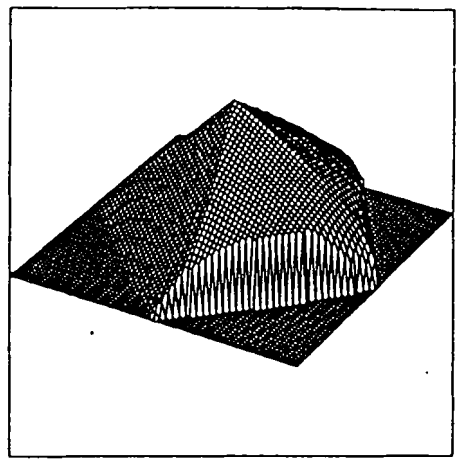
$$F(\theta) = a^2 cos^2(\frac{\theta}{2}) + b^2 sin^2(\frac{\theta}{2}) \leq 1 \tag{B.5}$$

Eq. (B.5) describes an ellipse which is enclosed by a unit circle, where the major axis of the ellipse lies in the direction $\theta : 0 \leftrightarrow \pi$. When $a > b$, which implies that β_1 and β_2 both have the same signs, $F(\theta)$ reaches a maximum at $\theta = 0$. But when $b > a$, i.e., when β_1 and β_2 have opposite signs, the maximum value of $F(\theta)$ occurs at $\theta = \pi$. These correspond to the two ellipses shown in Figs.3.3(a) and (b) respectively.

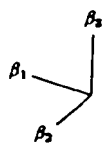
Appendix C Numerically derived 3-tap pitch synthesis filter stability regions for M=4,5,6 and 7

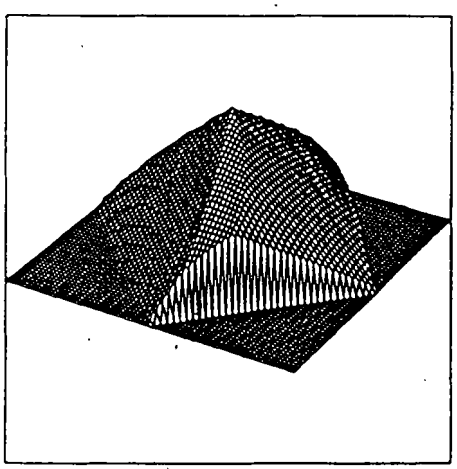


$M = 4$

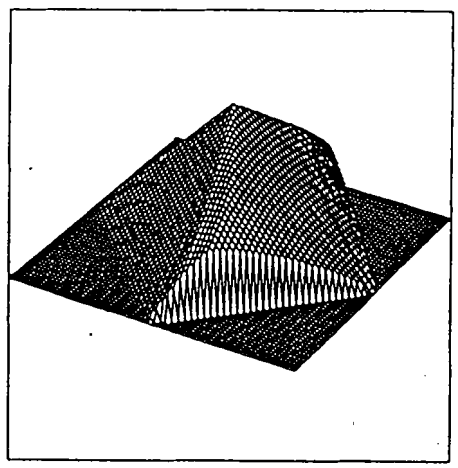


$M = 5$





$M = 6$



$M = 7$

Appendix D Schur-Cohn criterion for stability

Given a system whose denominator polynomial is

$$F(z) = a_0 + a_1 z + a_2 z^2 + ... + a_k z^k + ... + a_n z^n,$$

the system stability is guaranteed if the following constraints are satisfied.

$$|\Delta_k| < 0, \quad k - odd$$

$$|\Delta_k| > 0, \quad k - even$$

where:

$$\Delta_k = \begin{bmatrix} a_0 & 0 & 0 & ... & 0 & a_n & a_{n+1} & ... & a_{n-k+1} \\ a_1 & a_0 & 0 & ... & 0 & 0 & a_n & ... & a_{n-k+2} \\ . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . \\ a_{k-1} & a_{k-2} & a_{k-3} & ... & a_0 & 0 & 0 & ... & a_n \\ \bar{a}_n & 0 & 0 & ... & 0 & \bar{a}_0 & \bar{a}_1 & ... & \bar{a}_{k-1} \\ \bar{a}_{n-1} & \bar{a}_n & 0 & ... & 0 & 0 & \bar{a}_0 & ... & \bar{a}_{k-2} \\ . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & . & . \\ \bar{a}_{n-k+1} & \bar{a}_{n-k+2} & \bar{a}_{n-k+3} & ... & \bar{a}_n & . & . & ... & \bar{a}_0 \end{bmatrix}$$

$$k = 1, 2, ...n$$

$$\bar{a}_k = complex\ conjugate\ of\ a_k$$

Appendix E Matrices X_k and Y_k in Jury's Stability Criterion

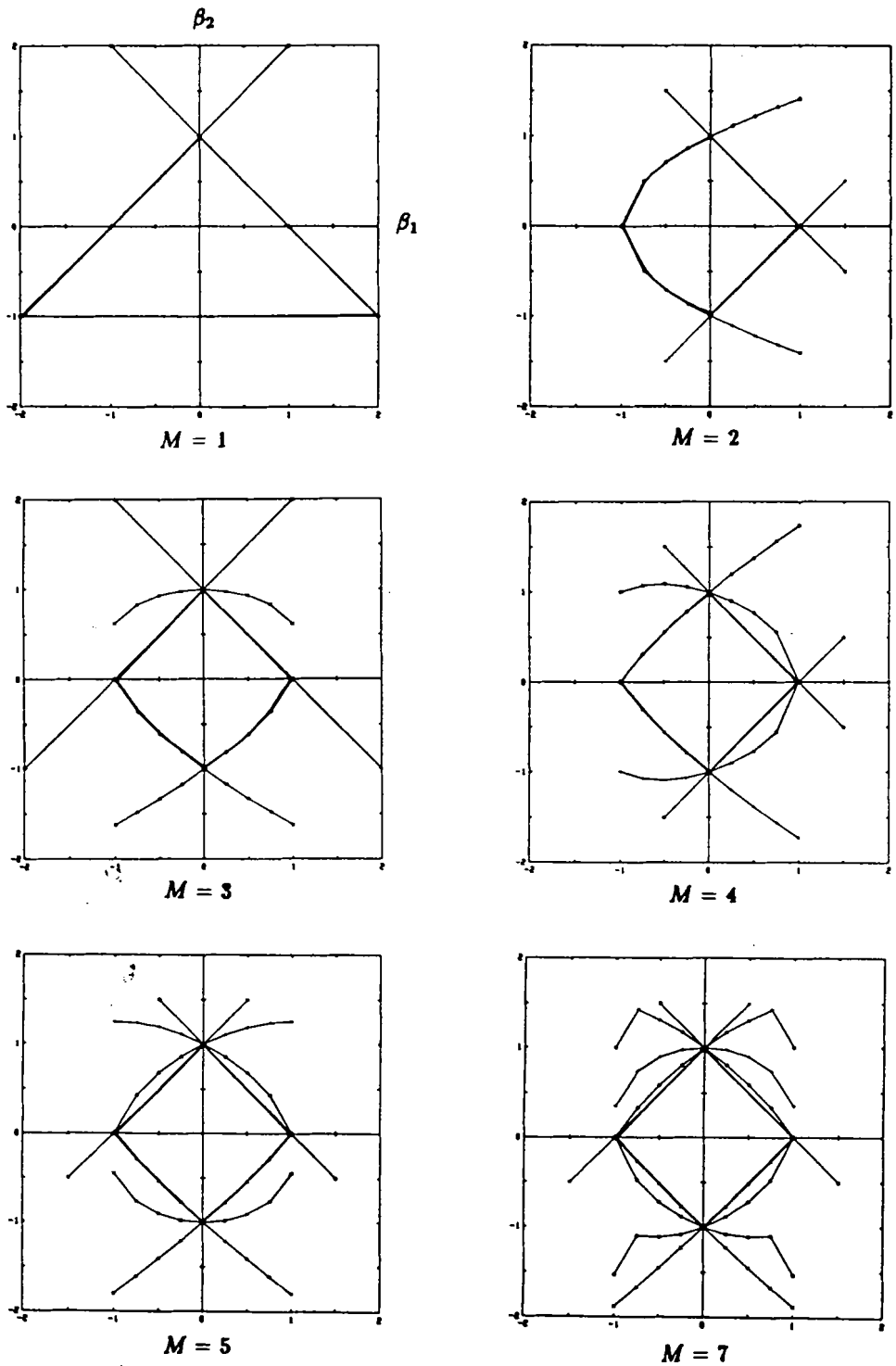
$$X_k = \begin{bmatrix} a_0 & a_1 & a_2 & \dots & a_{k-1} \\ 0 & a_0 & a_1 & a_2 & a_{k-2} \\ . & . & . & . & . \\ . & . & . & . & . \\ . & . & . & . & . \\ 0 & 0 & 0 & \dots & a_1 \\ 0 & 0 & 0 & \dots & a_2 \end{bmatrix}$$

$$Y_k = \begin{bmatrix} a_{n-k+1} & \dots & \dots & a_{n-1} & a_n \\ a_{n-k+2} & \dots & \dots & a_n & 0 \\ . & . & . & . & . \\ . & . & . & . & . \\ . & . & . & . & . \\ a_{n-1} & \dots & \dots & 0 & 0 \\ a_n & 0 & \dots & 0 & 0 \end{bmatrix}$$

where:

$$A_k - B_k = |X_k - Y_k|$$

Appendix F 2-tap pitch synthesis filter stability regions by Jury's criterion for M=1,2,3,4,5 and 7



Appendix G Speech Files (I)

FILE NAME	SPEAKER	DURATION (second)
CATM8	Male,English	2.464
CANM8	Male,English	2.002
PB1M2	Male,French	2.240
PB1M3	Male,French	2.080
CATF8	Female,English	2.208
TOMF8	Female,English	2.002
PB1F2	Female,French	2.432
PB1F3	Female,French	2.112

FILE CONTENTS:

CAT*8[†] — Cats and dogs hate the other.

CANM8 — The red canoe is gone.

TOMF8 — Tom's birthday is in June.

PB1*2 — C'est toujours comme ça depuis huit ans, tu sais.

PB1*3 — Ce cheval ne peut pas marcher au pas.

[†] (*) indicates both M/F have the same utterances

Appendix H Speech Files (II)

FILE NAME	SPEAKER	DURATION (second)
CATM8	Male	2.464
PB1M1	Male	2.112
DOUG5	Male	3.232
PIPM8	Male	2.304
PB1M5	Male	2.432
CATF8	Female	2.208
PB1F1	Female	2.272
VOICF5	Female	4.512
PIPF8	Female	2.464
PB1F5	Female	2.592

FILE CONTENTS:

CAT*8[†] — Cats and dogs hate the other.

PB1*1 — Est-ce que le conducteur arrête l'auto?

DOUG(VOICF)5 — Rice is often served in round bowls.

PIP*8 — The pipe began to rust while new.

PB1*5 — Ici il fait toujours très froid en hiver.

[†] (*) indicates both M/F have the same utterances

Appendix J Actual Values of F_c and F_i

For 2-tap filter

Frame	β	$(F_c = \frac{1}{\beta})$	F_1	F_2
3	1.002	0.998	100.001	99.999
19	1.295	0.772	100.300	99.661
21	1.124	0.890	100.145	99.870
30	1.023	0.977	100.010	99.928
45	1.087	0.920	100.089	99.877
55	1.563	0.640	100.778	99.210
65	1.072	0.933	100.030	99.833
69	1.035	0.966	100.067	99.973
79	1.025	0.975	100.033	99.989
87	1.006	0.994	100.011	99.997
91	1.045	0.957	100.094	99.981
92	1.005	0.995	100.013	99.999

For 3-tap filter

Frame	β	$(F_c = \frac{1}{\beta})$	F_1	F_2	F_3
3	1.001	0.999	0.999	0.999	0.999
11	1.005	0.995	0.995	0.995	0.995
17	1.924	0.520	0.528	0.520	0.512
21	1.766	0.566	0.574	0.567	0.559
37	1.126	0.888	0.892	0.889	0.886
45	1.110	0.901	0.904	0.901	0.899
46	1.311	0.763	0.768	0.763	0.758
47	1.288	0.777	0.782	0.777	0.772
55	1.625	0.615	0.628	0.617	0.606
56	1.122	0.891	0.896	0.892	0.888
65	1.092	0.916	0.918	0.916	0.913
66	1.166	0.857	0.862	0.858	0.854
67	1.272	0.786	0.792	0.786	0.781
68	1.152	0.868	0.871	0.868	0.865
69	1.220	0.820	0.824	0.819	0.815
79	1.168	0.856	0.858	0.856	0.854
83	1.011	0.989	0.989	0.989	0.989
85	1.221	0.819	0.825	0.821	0.816
87	1.017	0.983	0.984	0.983	0.983

References

[**ATAL(70)**] Atal, B.S. and Schroeder, M.R., "Predictive Coding of Speech Signals," *Bell System Tech. J.*, pp.1973–1986, Oct. 1970.

[**ATAL(71)**] Atal, B.S. and Hanauer, S.L., "Speech Analysis and Synthesis by Linear Prediction of the Speech Wave," *J. Acoust. Soc. Amer.*, vol. 50, no. 2, pp. 637–655, Aug. 1971.

[**ATAL(79)**] Atal, B.S. and Schroeder, M.R., "Predictive Coding of Speech Signals and Subjective Error Criteria," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-27, pp. 247–254, June 1979.

[**ATAL(80)**] Atal, B.S. and Schroeder, M.R., "Improved Quantizer for Adaptive Predictive Coding of Speech Signal at Low Bit Rates," *Proc. Int. Con. Acoust., Speech, Signal Processing*, Denver CO. pp. 535–538, Apr. 1980.

[**ATAL(82.A)**] Atal, B.S., "Predictive Coding of Speech at Low Bit Rates," *IEEE Trans. Commun.*, vol. COM-30, pp. 600–614, Apr. 1982.

[**ATAL(82.B)**] Atal, B.S. and Schroeder, M.R., "Speech Coding Using Efficient Block Code," *Proc. Int. Conf. on Acoustics, Speech and Signal Proc.*, vol. 3, pp. 1668–1671, May 1982.

[**ATAL(82.C)**] Atal, B.S. and Remde, J.R., "A New Model of LPC Excitation for Producing Natural-Sounding Speech at Low Bit Rate," *Proc. Int. Conf. on Acoustics, Speech and Signal Proc.*, Paris, France, pp. 614–617, May 1982.

[**ATAL(85)**] Atal, B.S. and Schroeder, M.R., "Code-Excited Linear Prediction (CELP): High-Quality Speech at Very Low Bit Rates," *Proc. Int. Conf. on Acoustics, Speech and Signal Proc.*, vol. 3, pp. 25.1.1–25.1.4, Mar. 1985.

[**BERG(71)**] Berger, T., *Rate Distortion Theory, A mathematical Basis for Data Compression*, Prentice-Hall, Englewood Cliffs, NJ, 1971.

[**BERO(78)**] Berouti, M. and Makhoul, J., "High Quality Adaptive Predictive Coding of Speech," *Int. Conf. Acoust., Speech and Signal Processing*, Tulsa, OK, pp. 303–306, Apr. 1978.

[**BEYR(85)**] Beyrouti, M., Kabal, P., Mermelstein, P. and Garten, H., "Computationally Efficient Multi-Pulse Speech Coding," *Proc. Int. Conf. Acoust., Speech and Signal Processing*, pp. 10.1.1–10.1.4., Mar. 1984.

[**DALD(80)**] Del Degan N. and Scagliola C., "Optimal Noise Shaping in Adaptive Predictive Coding of Speech," *Int. Conf. Acoust., Speech, Signal Processing*, pp. 539–542, Apr. 1980.

[**ELIA(55)**] Elias, P., "Predictive Coding," *IRE Trans. Inform. Theory*, vol. IT-1, pp. 16–33, Mar. 1955.

[**FELD(80)**] Feldman, J.A. and McAulay. R.J., "A Split Band Adaptive Predictive Coding (SBAPC) Speech System," *Int. Conf. Acoust., Speech, Signal Processing*, pp. 526–533, Apr. 1980.

[**FLAN(79)**] Flanagan, J.L., Schroeder, M.R., Atal, B.S., Crochiere, R.E., Jayant, N.S. and Tribolet, J.M., "Speech Coding," *IEEE Trans. Commun. Syst.*, vol. COM-27, pp. 710–736, Apr. 1979.

[**GERA(62)**] Gerald, C.F., *Applied Numerical Analysis*, Addison-Wesley Publishing Company, 1980.

[**GIBS(80)**] Gibson, J.D., "Adaptive Prediction in Speech Differential Encoding Systems," *Proc. IEEE*, vol. 68, no. 4, pp. 488-525, Apr. 1980.

[**GIBS(84)**] Gibson, J.D., "Adaptive Prediction for Speech Encoding," *IEEE ASSP magazine*, vol. 1, no. 3, pp. 12–26, July 1984.

[**GOLB(76)**] Golberg, A.J., Freudberg, R.L. and Cheung, R.S., "High Quality 16 Kb/s Voice Transmission," *Proc. Int. Conf. Acoust., Speech, Signal Processing*, pp. 244-246, Apr. 1976.

[**HAYK(78)**] Haykin, S., *Communication Systems*, John Wiley & Sons, inc., 1978.

[**JAYA(84)**] Jayant, N.S., Noll, P., *Digital Coding of Waveforms* , Prentice-Hall, Inc., 1984.

[**JURY(64)**] Jury, E.L., *Theory and Application of The z-Transform Method*, John Wiley & Sons, Inc., 1964.

[**KRAS(81)**] Krasner, M., Berouti, M. and Makhoul, J., "Stability Analysis of APC Systems," *Int. Conf. Acoust., Speech, Signal Processing*, pp. 627-630, Mar. 1981.

[**KABA(83)**] Kabal, P.,"The Stability of Adaptive Minimum Mean Square Error Equalizers Using Delayed Adjustment," *IEEE Trans. Comm.*, Vol. Com-31, pp. 430-432, Mar. 1983.

[**LANG(79)**] "A Simple Proof of Stability for All-Pole Linear Prediction Models," *Proc. IEEE*, vol. 67, no. 5, pp. 860-861, May 1979.

[**MAKH(75)**] "Linear Prediction: A tutorial Review," *Proc. IEEE*, vol. 63, no. 4, pp.561-580, Apr. 1975.

[**MAKH(77)**] "Stable and Efficient Lattice Methods for Linear Prediction," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-25, pp. 423-428, Oct. 1977.

[**MAKH(79.A)**] Makhoul, J. and Berouti, M., "Adaptive Noise Spectral Shaping and Entropy Coding in Predictive Coding of Speech," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-27, pp. 63-73, Feb. 1979.

[**MAKH(79.B)**] Makhoul, J. and Berouti, M., "Predictive and Residual Encoding of Speech," *J. Acoust. Soc. Amer.*, vol. 66(6), pp. 1633-1641, Dec. 1979.

[**OPPE(75)**] Oppenheim, A.V. and Schafer, R.W., *Digital Signal Processing*, Prentice-Hall, New Jersey, 1975.

[**PAEZ(72)**] Paez, M.D. and Glisson, T.H., "Minimum Mean Squared-Error Quantization in Speech," *IEEE Trans. Comm.*, Vol. Com-20, pp. 225-230, Apr. 1972.

[**QURE(75)**] Qureshi, S.U.H. and Forney, G.D.Jr., "Adaptive Residual Coder - An Experimental 9.6/16Kb/s Speech Digitizer," EASCON 1975, pp. 29A-29E.

[**RABI(78)**] Rabiner, L.R. and Schafer, R.W., *Digital Processing of Speech Signals*, Prentice-Hall, New Jersey, 1978.

[**SCHR(66)**] Schroeder, M.R., "Vocoders: Analysis and Synthesis of Speech," *Proc. IEEE* vol. 54, pp. 720-734, May 1966.

[**SCHR(84)**] Schroeder, M.R., "Linear Prediction, Entropy and Signal Analysis," *IEEE ASSP magazine*, vol. 1, no. 3, pp.3–11, July 1984.

[**VISW(75)**] Viswanathan, R. and Makhoul, J., "Quantization Properties of Transmission Parameters in Linear Predictive Systems," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-23, pp. 309–321, June 1975.

[**VISW(80)**] Viswanathan, R., Russell, W., Higgins, A., Berouti, M. and Makhoul, J., "Speech-Quality Optimization of 16 Kb/s Adaptive Predictive Coders," *Int. Conf. Acoust., Speech, Signal Processing*, vol. 2, pp. 520–525, Apr. 1980.