NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Reproduction in full or in part of this microform is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30, and subsequent amendments.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.

Parameter Retrieval of Auditory evoked Brainstem Response using Constrained Adaptive Notch Filters and Cross-Correlation Techniques

Rajeev Agarwal

A Thesis

in

The Department

of

Electrical and Computer Engineering

Presented in Partial Fulfilment of the Requirements
for the Degree of Masters of Applied Science at
Concordia University
Montreal, Quebec, Canada.

April 1991

ISBN 0-315-64697-7

ABSTRACT

Rajeev Agarwal

Auditory evoked Brainstem Response (ABR) audiometry is used to determine certain aspects of hearing capacity of an individual. Because ABRs are highly marred by the background electroencephalographic (EEG) activity, some means of signal processing is required to enhance these ABRs from the measured waveforms.

ABRs can be modelled as two sinusoidal components at around 500 and 1000 Hz. Using this knowledge, it is possible to use Constrained Adaptive Notch Filters (CANFs) as line enhancers, i.e. IIR-ALEs. When such IIR filters are implemented in the direct form, problems associated with local minima are inherent. In this thesis, it is shown how this local minima problem is dealt with, to improve the performance of CANFs. To deal with the short nature of the data, two modified algorithms based on the cascade implementation are presented. Validity of these algorithms is shown via simulation results. Application of these algorithms to real data are shown to have very encouraging results.

A clinically important parameter of ABRs is their latencies relative to the stimulus intensity level. Latency or time delay can be detected using Generalized Cross Correlators (GCCs). One section of this thesis will explore the detection of time delays for short data, with emphasis on the Maximum Likelihood GCC. Results of real ABR data latency detection are shown to be not better than the visual inspection.

Acknowledgement

I would like to express my deep gratitude, to my thesis supervisors, Dr. Eugene I. Plotkin and Dr. M.N.S. Swamy, for their continual interest and guidance throughout this work. Their constructive criticism and advice throughout this work was invaluable.

Last, but certainly not the least, I wish to express my deep gratitude and heartfelt appreciation to Dr. H.J. Ilecki, in the Department of Otolaryngology at the Royal Victoria Hospital, for introducing me to ABR audiometry and the need for signal processing in ABRs. Making his laboratory available for some experiments was of great help.

This work was supported through grants from Natural Science and Engineering Research Council and Fonds Pour la Formation de Chercheurs et L'Aide a la Recherche.

Rajeev Agarwal

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Evoked responses are a type of non-behavioural testing tool that has gained widespread acceptability over the last several years. Because it is a non-invasive technique and does not require subject cooperation, it is ideally suited for patients not easily testable by behavioural means. One type of evoked responses falls in the category of Auditory evoked Brainstem Response (ABR); they are used to assess a patient's hearing capacity. Because ABR are recorded without subject cooperation, this technique is ideally suited for infants, comatose and mentally handicapped.

ABRs are a form of electoencephlographic (EEG) recordings, difference being that they are acquired in a controlled manner. As the name suggests, ABRs are obtained by applying a known auditory stimulus to the ear and then measuring the transient response of the brain using surface electrodes pasted to the scalp. The response is a set of peaks and valleys in the 10 ms window at the onset of the application of the stimulus. Thus, if a subject does not hear the auditory stimulus, the response will not conform to the standard pattern and the deduction is hearing loss for that intensity level stimulus. In this manner, a clinician may acquire a set of waveforms at different stimulus intensity levels and determine the level of hearing loss (hearing threshold). There are other byproduct results from this type of testing, such as possible tumour detection.

As the stimulus intensity is decreased, the recorded response is expected to be latent with decreased signal energy. By acquiring these responses at various intensity levels, it is possible to generate a Latency-Intensity (LI) curve. If this curve deviates from the accepted standards, then it is an indication to possible development of a tumour in the auditory tract.

Having described the idea of ABR audiometry, it is important to understand the drawbacks or problems associated with this type of non-behavioural testing. Surface electrodes not only pick up the desired evoked response, but also pick up all the background EEG. The power of the background EEG can be as high as 200 times greater than the desired response. Thus, a major signal processing task is the extraction or enhancement of the evoked response from the noise (background EEG) when SNR is much less than 0 dB.

Audiology literature is rich in the techniques used to extract the elicited response. However, none have proven to be satisfactory. Even with the existence of the plethora of signal processing techniques, the most widely used technique in clinical evaluation of the acquired responses is still the visual inspection of the waveforms. One can immediately appreciate the lack of consistency between testers. The problem can then be defined for our purposes as the extraction of ABR from the evoked EEG recordings.

Through various studies over the years, the ABR spectra has been postulated to be containing three narrowband (NB) components, two of which (centred around 500 and 1000 Hz) are of importance in this study. The model can then be considered as two NB components immersed in assumed block stationary noise. The problem at hand now can be reformulated as sinusoid or NB signal enhancement.

In the last several years Infinite-Impulse Response Constrained Adaptive Notch Filters (IIR-CANFs) have been considered for the above NB component enhancement/suppression problem by various people. It has attracted attention due to its robustness and computational efficiency as compared to the standard Finite-Impulse Response Adaptive Line Enhancers (FIR-ALEs).

The existing studies of IIR-CANF are all based on the assumption of large observation intervals. The scope of this work is to study IIR-CANFs for short data and then use it in the application of ABR audiometry. In this work cascade realizations of the IIR-CANF are explored, and it is shown how improved performance over the Direct Form realization for the short data case can be obtained. Guaranteed convergence of cascade realization of second order CANFs is also shown. Two other implementations based on second order cascade realizations are shown to have further improved results.

Latency of the ABRs is directly proportional to the stimulus intensity levels. An ABR obtained at suprathreshold intensity (stimulus intensity much higher than the hearing threshold) is used as a reference, then all subsequent ABRs for lower intensity will yield a response delayed by a finite amount of time. Thus, pairing each ABR with the reference ABR, the estimating the latency is simply a problem of time delay estimation.

A classical method of estimating time delay between two similar signals is cross correlation; performance can be improved by multiplying the two signals by a frequency weighting function depending on SNR, prior to cross correlation. Resulting estimators are known as Generalized Cross Correlators (GCCs). The weighting function is generally

estimated from the received signals. Thus, after extraction of the ABRs using IIR-CANFs, we may be able to determine the inter-intensity latencies using GCCs.

The following describes the breakdown of this thesis by chapters:

Chapter 2 will cover the theory of the classical method of cross correlation to estimate time delays between two signals. The improved Generalized Cross Correlators (GCCs) will be presented, with an emphasis on the Maximum Likelihood cross correlator.

Chapter 3 will start with a general discussion of the IIR filter formulation and then cover the theoretical background of the IIR-CANFs. Formulation of its structure, two different parameterizations as well as the adaptive schemes will be developed. The performance of the CANFs will also be discussed.

Chapter 4 will describe alternate realizations of the IIR-CANFs. It is attempted to study the behaviour of the error or energy surfaces of various realization of the CANFs. Emphasis is placed on modifications to obtain improved results for short data.

Chapter 5 will cover the simulation study of the various GCCs and the different realizations of the CANFs presented in the previous chapters, for short data.

Chapter 6 discusses in detail the concept of Auditory evoked Brainstem Responses as used in the hearing assessment of individuals. Currently used techniques in the clinical assessment of hearing loss are also presented. It is attempted to familiarize the reader with this biomedical application.

Chapter 7 will present the results of using CANFs and GCC in the assessment of hearing loss using real data.

Chapter 8 will conclude this study of CANFs, GCCs and their application in the ABR audiometry environment.

Chapter 2

Time Delay Estimation Using Cross Correlation

Estimation of time delay between a noise corrupted signal received at two spatially removed sensors has attracted a great deal of attention in recent years. This time delay, in the arrival of signal, has many applications. One such application is in the area of audiology. It is the detection of latency in the auditory evoked response of the midbrain at different stimulus intensity levels. This latency vs. intensity curve can then be used for the clinical diagnosis of the hearing [1]. A discussion on the use of this idea in the audiological assessment will be deferred to the chapter on application in the area of Auditory Evoked Brainstem Response. In this chapter, the concept of general time delay estimation using Cross Correlation will be discussed. More efficient Generalized Cross Correlation algorithms will also be presented with a particular emphasis on the Maximum Likelihood processor.

2.1 Cross Correlation

Cross Correlation (CC) is a technique that is frequently used in the quantitative as well as the qualitative measurements in digital signal processing, examples of which are the above time delay estimation and the detection of signals buried in noise, respectively. Correlation techniques are most often used when the signals are contaminated by noise or if there is no knowledge of the transfer function. Correlation can be regarded as a relationship between two signals x(t) and y(t), as shown in Fig. 2.1. Fig.

2.1a shows an input/output relationship and Fig. 2.1b shows s(t) received as x(t) and y(t) at two spatially separated sensors, respectively. The term relationship refers to how the input x(t) is affected at the output y(t) by the transmission path. It can be viewed as a measure of the path. Here, we are interested in measuring the time delay caused by the system (whether the system can or cannot be characterized) at the output [2].

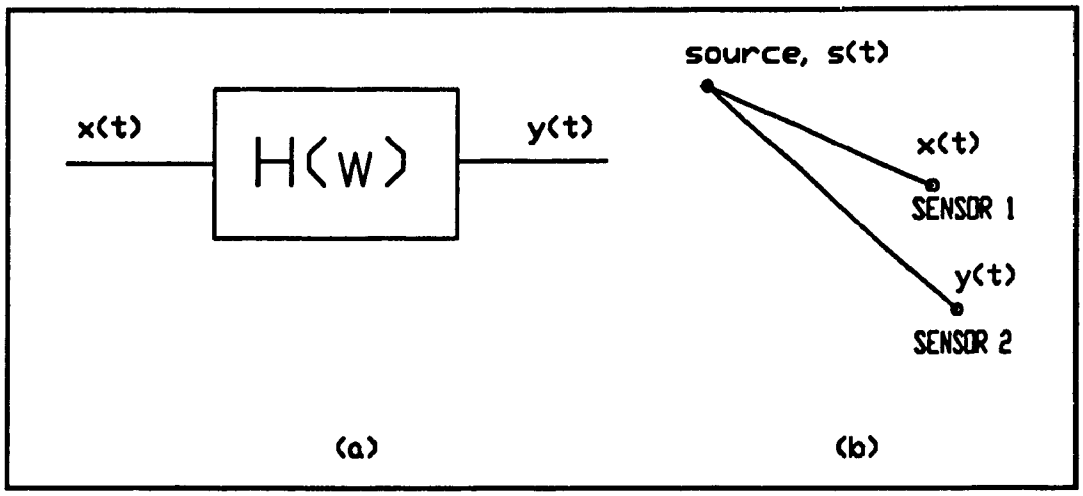

Figure 2.1. (a) Input/Output relationship of x(t) and y(t); (b) source signal s(t) received as x(t) and y(t) at spatially removed sensors.

Consider the signals x(t) and y(t) of Fig. 2.2, where y(t) is a delayed version of x(t). One may consider, them to be input/output signal of a linear system or one signal measured at two spatially separated sensors in Fig. 2.1. The following simplified model can then be formulated as

$$\begin{aligned} x(t) &= s(t) + n_1(t) \\ y(t) &= \alpha s(t-D) + n_2(t), \end{aligned} \tag{2.1}$$

where $n_1(t)$ and $n_2(t)$ are noise signals uncorrelated with the source signal, s(t).

The estimate of the relationship between x(t) and y(t) over the observation interval, T, is given by the Cross Correlation function [3]

$$R_{xy}(\tau)=E\{x(t)y(t+\tau)\}, \tag{2.2}$$

where $R_{xy}(\tau)$ is the cross correlation function of x(t) and y(t) and τ is a new variable defining the lag between x(t) and y(t). When the cross correlation function is a high value for some τ, the two signals of concern are said to be highly related or similar and when the value approaches 0 than they are said to be not related. If one considers sliding the signal y(t) along the time axis (i.e. varying the lag τ) then it is easy to visualize that the maximum value of the CC function will occur when the lag, τ, between the two signals is equal to the value of the delay, D. Thus, the time delay between the two signals can be estimated as the lag that causes the CC function to achieve the maximum value [4].

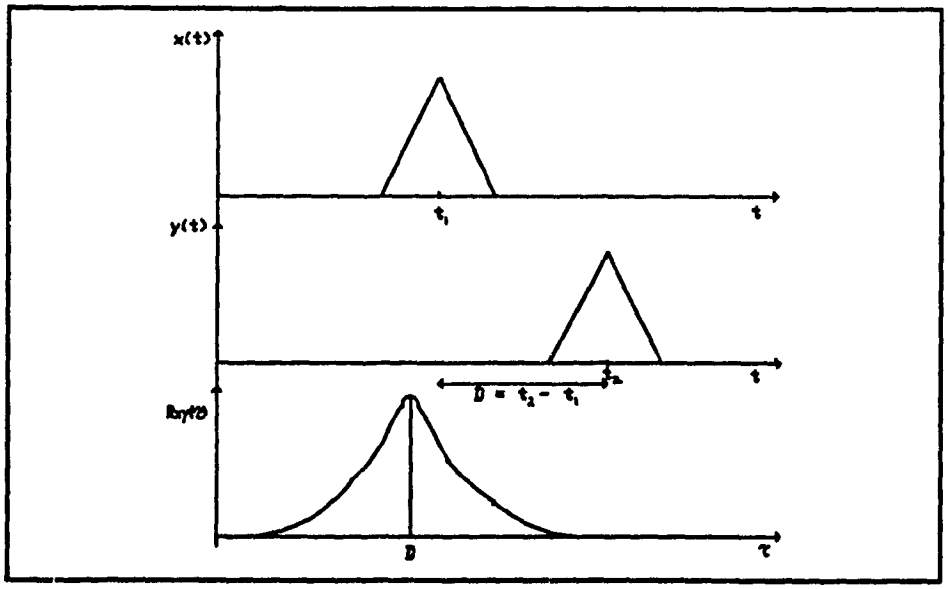

Figure 2.2. (a) x(t); (b) y(t), the delayed signal x(t); (c) cross correlation of x(t) and y(t).

Performance of Cross Correlation

The CC defined by Eq 2.2 assumes that a joint probability density function is available to calculate $R_{xy}(\tau)$. However, in practice, only a finite duration of the signal is available (in the interval [0,T]). In this case, the CC function is approximated by its instantaneous values under the assumption that x(t) and y(t) are jointly ergodic processes over the observation interval [0,T] as [3,4]

$$\hat{R}_{xy}(\tau) = \frac{1}{T}\int_0^T x(t)y(t+\tau)dt \tag{2.3}$$

where the hat over R_{xy} indicates an estimated value.

The true value of the CC function is represented by $R_{xy}(\tau)$. It is desirable that the expected value (or the statistical mean) of $\hat{R}_{xy}(\tau)$ equals $R_{xy}(\tau)$ without any error, i.e that the proposed estimator is unbiased. If, $E[\hat{R}_{xy}(\tau)] \neq R_{xy}(\tau)$, then the bias is defined as [4,5]

$$b[\hat{R}_{xy}(\tau)] = E\{\hat{R}_{xy}(\tau)\} - R_{xy}(\tau) = E\{\hat{R}_{xy}(\tau) - R_{xy}(\tau)\}. \tag{2.4}$$

For the considered situation

$$\begin{aligned} E\{\hat{R}_{xy}(\tau)\} &= E\left\{\frac{1}{T}\int_0^T x(t)y(t+\tau)dt\right\} = \frac{1}{T}\int_0^T E\{x(t)y(t+\tau)\}dt \\ &= \frac{1}{T}\int_0^T R_{xy}(\tau)dt = R_{xy}(\tau) \end{aligned} \tag{2.5}$$

Thus, the proposed estimator given by Eq 2.3 is unbiased. For a fixed value of T, an unbiased estimate of $\hat{R}_{xy}(\tau)$, does not imply that $\hat{R}_{xy}(\tau)$ will be close to the true value $R_{xy}(\tau)$. It may also happen that as T' is increased, the bias may not decrease [4]. To analyze such a case, the *mean square error (mse)*, is defined to be the expected value of square of the deviation of the estimate from true value, that is [4]

$$mse = E\left\{(\hat{R}_{xy}(\tau)-R_{xy}(\tau))^2\right\} \tag{2.6}$$

As the observation interval T is increased, it is desirable that *mse* approach zero. Thus, if

$$\lim_{T \to \infty} mse = \lim_{T \to \infty} E\left\{(\hat{R}_{xy}(\tau)-R_{xy}(\tau))^2\right\} = 0 \tag{2.7}$$

then the estimate is termed consistent [5]. For the non asymptotic case (the limited observation interval T), if the *mse* approaches a minimum theoretical lower bound - usually the Cramer-Rao Lower bound, described later in this chapter - we speak of the tightness or the efficiency of the estimator.

2.2 Generalized Cross Correlation

In the previous section it was described how Cross Correlation can be used to estimate the time delay. An improvement in the accuracy of the time delay estimation can be achieved by using a more efficient Generalized Cross Correlator, (GCC). GCC is simply a Cross Correlator with its inputs x(t) and y(t) being prefiltered as is shown in Fig. 2.3. The prefilters, $H_1(f)$ and $H_2(f)$ of x(t) and y(t), respectively, are selected to accentuate the signal at frequencies where the signal to noise ratio (SNR) is the highest and simultaneously suppress noise [6].

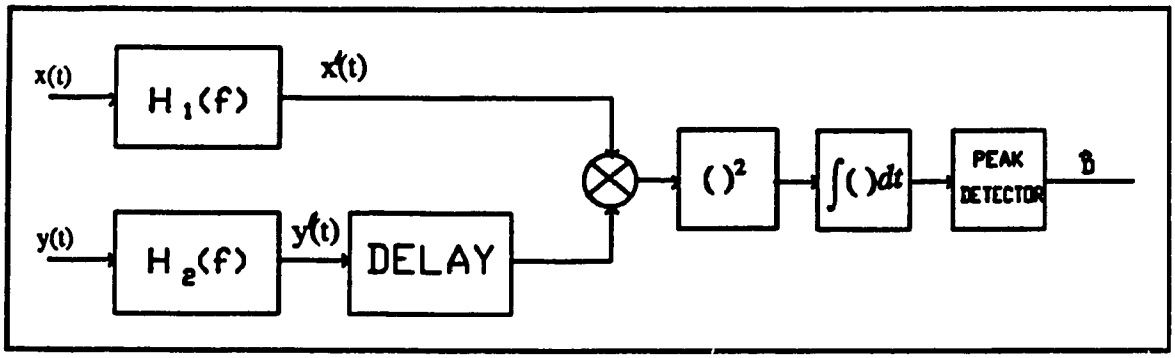

Figure 2.3. Generalized Cross Correlator.

Cross power spectral density (PSD) between the outputs of the prefilters in Fig. 2.3 is given as [7]

$$S_{x'y'}(f) = H_1(f)H_2^*(f)S_{xy}(f) = W(f)S_{xy}(f), \tag{2.8}$$

where W(f) is the equivalent frequency weighting function and $S_{xy}(f)$ is the cross PSD of x(t) and y(t) and the * indicates complex conjugate. Now, the GCC is the inverse Fourier transform of $S_{x'y'}(f)$ [6,7],

$$R_{x'y'}(\tau) = \int_{-\infty}^{+\infty} S_{x'y'}(f)e^{j2\pi f\tau}df = \int_{-\infty}^{+\infty} W(f)S_{xy}(f)e^{j2\pi f\tau}df\,. \tag{2.9}$$

Due to the finite observation interval, only an estimate of $S_{xy}(f)$ of the input data is available. Hence, only an estimate of $R_{x'y'}(\tau)$ is used in practice. Since the prefilters accentuate the input signal at frequencies where SNR is the highest, thus, to choose the prefilters, *a priori* knowledge of the input spectra is necessary. As this is usually not available in practice, an estimate of the prefilters (or the frequency weighting function, W(f)) must also be utilized. For good resolution, the prefilters should be chosen such that

the resulting GCC has large sharp peaks. Table 2.1 lists the processors that will be discussed in this chapter; the Roth and SCOT processors will be briefly described, while the remainder of the chapter will be devoted to the Maximum Likelihood processor.

Table 2.1. Frequency weighting functions for the various GCCs.

Processor Name	Weighting Function, W(f)
Cross Correlation (CC)	1
ROTH	$\frac{1}{S_{xx}(f)}$
SCOT	$\frac{1}{\sqrt{S_{xx}(f)\,S_{yy}(f)}}$
ML	$\frac{\lvert\gamma_{xy}(f)\rvert^2}{\lvert S_{xx}(f)\rvert\,[1-\lvert\gamma_{xy}(f)\rvert]}$

2.2.1 The Roth Processor

The Roth processor is an ad hoc prefilter that has the desired feature of suppressing the frequencies where the noise spectrum is large and the cross spectrum of the input is more likely to be erroneous (i.e. low SNR cases). The proposed Roth processor is given as the inverse of the PSD of the signal x(t) as [2],

$$W^R(f) = \frac{1}{S_{xx}(f)} \tag{2.10}$$

where the superscript R denotes Roth processor. The auto and cross PSDs of x(t) and y(t) are defined as

$$\begin{aligned} S_{xx}(f) &= S_{ss}(f)+S_{n_1n_1}(f) \\ S_{yy}(f) &= \alpha^2 S_{ss}(f)+S_{n_2n_2}(f) \\ S_{xy}(f) &= \alpha S_{ss}(f)e^{-j2\pi fD} \end{aligned} \tag{2.11}$$

Using Eq 2.10 as $W^R(f)$, the Roth GCC can now be written as

$$\begin{aligned} R_{xy}^R(\tau) &= \int_{-\infty}^{+\infty} \frac{S_{xy}(f)}{S_{xx}(f)} e^{j2\pi f\tau} df \\ &= \int_{-\infty}^{+\infty} \frac{\alpha S_{ss}(f)e^{-j2\pi fD}}{S_{ss}(f) + S_{n_1n_1}(f)} e^{j2\pi f\tau} df \end{aligned} \tag{2.12}$$

The rationale for dividing the cross PSD, $S_{xy}(f)$, by the PSD of x(t) ($S_{xx}(f)$) is that this technique has an intuitive meaning when the model of Fig. 2.1a is used. It can be considered as the impulse response of the linear input/output relationship of x(t) and y(t), which in certain situations has a higher resolution than the cross-correlation techniques. When there is no such model, then there is no justificaion for this normalization. One might, therefore, be puzzled as to whether the PSD of y(t) should be used for the weighting of $S_{xy}(f)$ [2,8]. From Eq 2.12, it can be seen that the Roth processor has the desirable feature of suppressing those frequency bands corresponding to large $S_{n_1n_1}(f)$, where $S_{xy}(f)$ is more likely to be erroneous. A technique that favours neither $1/S_{xx}(f)$ or $1/S_{yy}(f)$ as the weighting function is described in the next section [6].

2.2.2 The Smooth Coherence Transform (SCOT)

In the Roth processor of the previous section, it is necessary to choose the frequency weighting function as the PSD of either x(t) or y(t). The errors in $S_{xy}(f)$ may be due to frequency bands where $S_{n_1n_1}(f)$ is large as wells as bands where $S_{n_2n_2}(f)$ is large. One, therefore, has to make a choice between $W^R(f) = 1/S_{xx}(f)$ or $1/S_{yy}(f)$. For this reason, another ad hoc processor, SCOT, is motivated from the Roth processor. It favours neither of the two signals [6,8]. It selects

$$W^S(f) = \frac{1}{\sqrt{S_{xx}(f)S_{yy}(f)}} \tag{2.13}$$

Thus, giving the SCOT GCC as

$$R^S_{x_1y}(\tau) = \int_{-\infty}^{\infty} \frac{S_{xy}(f)}{\sqrt{S_{xx}(f)S_{yy}(f)}} e^{j2\pi f\tau}\, df. \tag{2.14}$$

where the superscript S is used to denote SCOT. With $H_1(f) = [S_{xx}(f)]^{-1/2}$ and $H_2(f) = [S_{yy}(f)]^{-1/2}$, the SCOT processor can be viewed as a prewhitening filter [5,6].

2.2.3 The Maximum Likelihood Processor (ML)

In order to make the derivation of the ML estimator mathematically tractable, it is necessary to assume all the signal components (i.e. s(t), $n_1(t)$ and $n_2(t)$) of model of Eq 2.1 have a gaussian distribution.

The received signals, x(t) and y(t), with a finite observation time interval, can be represented as Fourier Coefficients in the frequecncy domain as [6,9]

$$X(k) = \frac{1}{\sqrt{T}}\int_0^T x(t)e^{j\omega_k t}dt$$
$$\text{and} \qquad (2.15)$$
$$Y(k) = \frac{1}{\sqrt{T}}\int_0^T y(t)e^{j\omega_k t}dt$$

where $\omega_k = k\omega_0$ are the discrete frequencies, while ω_0 ($= 2\pi/T$) being the fundamental frequency. The Fourier Coefficients (FCs), X(k) and Y(k), can be arranged in a stacked vector form represented by

$$R = \begin{bmatrix} R(1) \\ R(2) \\ \vdots \\ R(N) \end{bmatrix} \quad \text{where} \quad R(k) = \begin{bmatrix} X(k) \\ Y(k) \end{bmatrix}. \qquad (2.16)$$

Under the assumption that the observation time is large, compared to the delay plus the correlation time, the FCs corresponding to different frequencies are uncorrelated [6]. Thus, a diagonal form of the covariance matrix can be obtained as

$$E\{RR^H\} = Q_R = \begin{bmatrix} Q_R(1) & & \\ & \ddots & \\ & & Q_R(N) \end{bmatrix} \quad \text{where} \quad Q_R(k) = \begin{bmatrix} S_{xx}(k) & S_{xy}(k) \\ S_{xy}^*(k) & S_{yy}(k) \end{bmatrix} \qquad (2.17)$$

where H represents the complex conjugate transposition. Thus, the entries of the covariance matrix, Q_R, represent the auto and cross PSD values of signals x(t) and y(t) evaluated at discrete frequencies. Q_R can be considered to be a diagonal matrix with sub-matrices as its elements. The cross terms of the sub-matrices are functions of the delay, D, as shown in Eq 2.18

$$Q_R = \frac{1}{T}\begin{bmatrix} \begin{bmatrix} S_{xx}(1) & S_{xy}(1)e^{j\omega_1 D} \\ S_{xy}(1)e^{-j\omega_1 D} & S_{yy}(1) \end{bmatrix} & & \\ & \ddots & \\ & & \begin{bmatrix} S_{xx}(N) & S_{xy}(N)e^{j\omega_N D} \\ S_{xy}(N)e^{-j\omega_N D} & S_{yy}(N) \end{bmatrix} \end{bmatrix} \tag{2.18}$$

Since Fourier transform is a linear operation, it follows that FCs are also gaussian. Hence, the conditional probability can be written as [5]

$$p(R/D) = \frac{1}{\sqrt[n]{2\pi|Q_R|}} e^{-\frac{1}{2}\sum_{i=1}^{N} R^H(k)Q_R(k)R(k)} = Ce^{-\frac{1}{2}J}. \tag{2.19}$$

Therefore, in order to obtain the maximum likelihood delay estimate, $\hat{D}_{ML}$, it is necessary to maximize p(R/D) with respect to D [5]. C in Eq 2.19 can be made to be independent of D (see Appendix A), thus, it is only necessary to minimize J, such that

$$\min_{D} J(D) \rightarrow \hat{D}_{ML} \tag{2.20}$$

Using the convergence theory of Fourier transform and under the assumption that the observation time is larger than the correlation time, the summation in Eq. 2.19 can be written as [6]

$$J(D) = \int R^H(f)Q_r^{-1}R(f)df \tag{2.21}$$

J can be further written as $J = J_2 + J_3$, where J_2 is independent of D and is +'ve and J_3 is (see Appendix B)

$$J_3 = -2T\int \frac{S_{xy}(f)}{|S_{xy}(f)|}\frac{|\gamma_{xy}(f)|}{1-|\gamma_{xy}(f)|^2}e^{j2\pi fD}df \qquad (2.22)$$

$$\text{where } \gamma_{xy}(f) = \frac{|S_{xy}(f)|^2}{S_{xx}(f)S_{yy}(f)}$$

Now, it is only necessary to minimize J_3 in order for J to be minimized. We, thus, choose

$$R_{x'y}^{ML}(\tau) = \int_{-\infty}^{+\infty} \frac{S_{xy}(f)}{|S_{xy}(f)|}\frac{|\gamma_{xy}(f)|}{1-|\gamma_{xy}(f)|^2}e^{j2\pi f\tau}df. \qquad (2.25)$$

as the Maximum Likelihood GCC such that Eq 2.22 achieves a sharp peak [6,9]. Therefore, the ML processor has the following frequency weighting function [6,9]

$$W^{ML}(f) = \frac{1}{|S_{xy}(f)|}\frac{|\gamma_{xy}(f)|^2}{1-|\gamma_{xy}(f)|^2}. \qquad (2.24)$$

Performance of Maximum Likelihood Estimator

The effectiveness of an estimator can be expressed in terms of bias and variance. Since *a priori* knowledge is limited in most cases, the evaluation of the bias and the variance represents a difficult task. An alternate form of evaluation of the estimator performance is to compute a lower bound for the variance (being confident that the estimator is an unbiased one) and then compare the performance against it [5,10]. One such bound, that is used to describe the performance of an *unbiased* estimator, is the Cramer-Rao lower bound (CRLB). The performance of the ML estimate can be described by the CRLB because it is an efficient estimate when the observation time is large [5]. To see this, the variance of the time delay (within the neighbourhood of some true delay) of GCC with an arbitrary weighting function, W(f), is given by [6]

$$var(D) = \frac{\int_{-\infty}^{\infty} |W(f)|^2 (2\pi f)^2 S_{xx}(f)[1-\gamma_{xy}(f)]^2 df}{T\int_{-\infty}^{\infty} (2\pi f)^2 |S_{xy}(f)| W(f) df}. \quad (2.25)$$

Substituting the weighting function, W(f), equal to $W^{ML}(f)$ of Eq 2.24 in this expression, results in Eq 2.26, which is exactly the considerd CRLB [6].

$$var^{ML}(D) = \left[2T\int_{-\infty}^{\infty} (2\pi f)^2 |\gamma_{xy}(f)|^2 \frac{1}{[1-|\gamma_{xy}(f)|^2} df\right]^{-1}. \quad (2.26)$$

Evaluation of CRLB

For model of Eq 2.1, with $\alpha = 1$, the MSC ($|\gamma_{xy}(f)|^2$) can be written as

$$|\gamma_{xy}(f)|^2 = \frac{|S_{ss}(f)|^2}{[S_{ss}(f) + S_{nn}(f)]^2} \quad (2.27)$$

Thus,

$$\frac{|\gamma_{xy}(f)|^2}{1 - |\gamma_{xy}(f)|^2} = \frac{S_{ss}(f)^2}{S_{ss}^2(f) + 2S_{ss}(f)S_{nn}(f)} \quad (2.28)$$

Now, considering the evaluation of the CRLB (Eq 2.26), for the case of source signal having a flat Power Spectral Density (PSD) over the signal bandwidth and the two noise signals and the source signal being mutually uncorrelated, the MSC is constant over the signal bandwidth and zero elsewhere. Thus, dividing the numerator and the denominator of Eq 2.28 by $S_{nn}^2(f)$ and removing the dependence on frequency, yields,

$$\frac{\gamma_{xy}^2}{1 - \gamma_{xy}^2} = \frac{S^2/N^2}{1 + 2S/N} \tag{2.29}$$

where $S = S_{ss}(f)$, $N = S_{nn}(f)$ and S/N = SNR, denotes the signal to noise ratio. Substituting Eq 2.29 in Eq 2.26 and evaluating the integral results in the desired closed form expression for the CRLB for this case,

$$\sigma_{CRLB}^2 = \frac{3}{8T\pi^2} \frac{1 + 2SNR}{SNR^2} \frac{1}{(f_2^3 - f_1^3)} \tag{2.30}$$

2.3 Comparison of Processors

Up to now various different processors for time delay estimation were briefly described. A particular emphasis was placed on the Maximum Likelihood estimator. In this section, the ML estimator will be compared to the other estimators. Since good time delay estimation is most difficult under low SNR, the comparisons will be done under this assumption, that is,

$$\frac{S_{ss}(f)}{S_{n_1n_1}(f)} \ll 1 \quad and \quad \frac{S_{ss}(f)}{S_{n_2n_2}(f)} \ll 1 \tag{2.31}$$

For the purpose of comparison the value of α, in Eq 2.1, is chosen to be 1. Under these assumptions the Roth, SCOT, and the Maximum Likelihood processors can be rewritten as Eqs 2.32, 2.33, and 2.34, respectively.

$$W^R(f) = \frac{1}{S_{n_1n_1}(f)} \text{ or } \frac{1}{S_{n_2n_2}(f)} \tag{2.32}$$

$$W^{S}(f) = \frac{1}{\sqrt{S_{n_1n_1}(f)S_{n_1n_1}(f)}} \tag{2.33}$$

$$W^{ML}(f) = \frac{S_{ss}(f)}{S_{n_1n_1}(f)S_{n_2n_2}(f)} \tag{2.34}$$

Comparing the Roth and the ML processor, it is seen that ML processor is a Roth processor with an additional weighting depending on SNR:

$$W^{ML}(f) = \frac{S_{ss}(f)}{S_{n_1n_1}}W^{R}(f) \text{ or } \frac{S_{ss}(f)}{S_{n_2n_2}(f)}W^{R}(f) \tag{2.35}$$

Similarly, ML processor can be represented as the SCOT processor with additional weighting depending on SNR, see Eq 2.36.

$$W^{ML}(f) = \frac{S_{ss}(f)}{\sqrt{S_{n_1n_1}(f)S_{n_2n_2}(f)}}W^{S}(f) \tag{2.36}$$

Comparing, the ML processor to the other two processors in Eqs 2.35 and 2.36, it is seen that ML is equivalent to all three with additional weighting based on the SNR. And thus, in this way it removes certain restrictions or shortfalls of the other three processors. In the Roth weighting, a choice has to made with respect to which of the two signals to use in the weighting, the SCOT processor as with Roth displays the spreading (or broadening) of the correlation function. All of these existing problems are minimized with the Maximum Likelihood processor leading to a minimum variance estimate, as is apparent from the simulation study in chapter 5.

2.4 References

1. W . Woodsworth, S. Reisman, A.B. Fontaine, " The detection of auditory evoked responses using a matched filter," IEEE Trans. Biomedical Eng., vol. 36, p.81, 1970.

2. P.R. Roth, " Effective measurements using signal analysis," IEEE Spectrum, vol. 8, p. 62, Apr. 1971.

3. A. Papoulis, Signal Analysis, McGraw Hill, 1977.

4. J.S. Bendat and A.G. Piersol, Measurement and Analysis of Random Data, Wiley, 1966.

5. H.L. Van Trees, Detection, Estimation, and Modulation Theory, Part I Wiley, 1968.

6. C.H. Knapp and G.C. Carter, "The Generalized correlation method for estimation of time delay," IEEE Trans. Acoust. Speech and Signal Processing, vol. ASSP-24, p. 320, 1976.

7. A. Papoulis, Probability, Random Variable and Stochastic Processes, McGraw Hill, 1966.

8. G.C. Carter, A.H. Nuttall and P.G. Cable, " The Smooth Coherence Transform," IEEE Proceedings, October 1973.

9. M. Wax, " The estimation of time delay between two signals with random relative phase shift," IEEE Trans. Acoust, Speech and Signal Processing, vol. ASSP-29, June 1981.

10. M.D. Srinath and P.K. Rajeskaran, An introduction to Statistical Signal Processing with Applications, Wiley 1979.

Chapter 3

Development of Constrained Adaptive Notch Filter (CANF)

The problem of enhancement of narrowband (NB) or sinewave components in signal processing falls in the category of Adaptive Line Enhancement (ALE). It is a multifaceted problem, in that, an estimate of the sinewave frequency, enhancement of NB signal immersed in broadband (BB) noise, or the tracking of signal with time varying frequency may be desired. Applications of this kind of signal processing are numerous. The most popular of the ALE has been the tapped-delay line structure [1] and is now termed the finite impulse response ALE (FIR-ALE). The advantage of this type of filter is the simplicity of the adaptive scheme. However, as the power of the noise is increased in the observed time series, the FIR model requires large tapped delay line and thus defeating its simple adaptive structure. This all-zero filter structure can be replaced by an IIR filter that uses fewer number of delay elements.

This chapter will deal with the development of IIR-ALE, an idea that has emerged in recent years. A constrained IIR filter known as the Constrained Adaptive Notch Filter (CANF) will be explored. The first section will give a brief overview of adaptive IIR filters. In the following sections, the structure of a general constrained notch filter will be developed, along with an alternate realization using minimal number of parameters. Adaptive parameter estimation algorithms will also be derived. And finally, the performance of this structure will be discussed.

3.1 Infinite-Impulse Response (IIR) Filter

In the last several years adaptive Infinite-Impulse-Response (IIR) filters have been a topic of active research. They have been considered for a wide range of problems, in areas like echo cancellation, linear prediction, channel equalization, adaptive differential pulse code modulation, adaptive array processing and notch filtering, just to name a few. Application of notch filtering will be discussed in detail in the next section. Many of the techniques used in adaptive IIR filtering have arisen from the theory of system identification, where it is assumed that the system under investigation has a pole-zero model [2].

IIR filters have been contemplated for the replacement of the FIR filter due to their greater latitude in generating diverse spectral shapes [3]. An adaptive IIR filter shows significantly better results than the adaptive FIR filter having the same number of coefficients. This is primarily due to the fact that an infinite impulse response is attained with a relatively small number of coefficients, whereas the FIR would require a large number of coefficients. Frequency response can be modeled more effectively with a pole-zero configuration than with an all zero configuration [2]. The decrease in computations in estimating the fewer number of coefficients is the primary reason for the advances towards the IIR filter.

When IIR filters used to represent a process have rational functions of the form,

$$H(Z) = \frac{P(Z)}{Q(Z)} = \frac{\sum_{k=0}^{M} b_k Z^{-k}}{1 - \sum_{k=1}^{N} a_k Z^{-k}} \qquad (3.1)$$

it is termed an autoregressive-moving average (ARMA) model [3]. Fig 3.1 depicts the general structure of the adaptive IIR filter that has the ARMA structure for the filter H(Z). The desired response is characterized by the signal d(k), x(k) and y(k) represent the input and the output, respectively. As can be seen, this is a time varying filter, the filter coefficients being adjusted by the recursive algorithm to minimize some performance criterion. That is, the recursive algorithm adjusts the coefficients such that the output response, y(k), will simulate the desired response, d(k). To be more specific, the recursive algorithm tries to minimize the prediction error defined by e(k) = y(k) - d(k). Some of the common error criteria are the mean-squared error (MSE) and the least square method that results in the recursive least square (RLS).

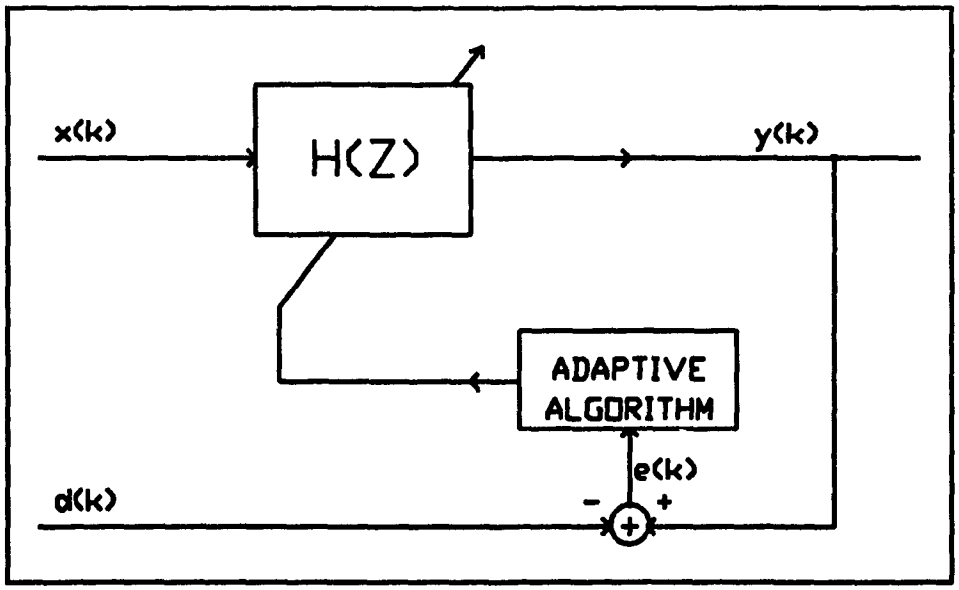

Figure 3.1. General structure of an adaptive filter.

Typically, there are two formulations of the prediction error, e(k), in Fig. 3.1. They are the equation error and the output error. In the equation error, the coefficient are updated by first updating all the zero coefficients and then copying them to the pole section of the filter. This formulation leads to a biased estimate of coefficients. The output error formulation on the other hand updates the coefficients directly in a pole-zero recursive form. This method does not lead to a biased estimate; however, the coefficients may converge to a local minima, thereby producing erroneous results. Both of these methods are outlined in more detail next.

3.1.1 Equation Error Formulation

As mentioned above, the equation-error formulation leads to a simple recursive algorithm. Fig 3.2 and the difference equation (Eq 3.2) describe the simple structure of the equation-error formulation, where the subscript e denotes equation error formulation.

$$y_e(k) = \sum_{m=0}^{M-1} b_m(k)x(k-m) + \sum_{m=1}^{N-1} a_m(k)d(k-m) \qquad (3.2)$$

Equation error is considered to be a two-input single-output system. From Eq 3.2, it is seen that $y_e(k)$ is not a function of previous values of itself, it depends on delayed values of the desired response, d(k) and the input data, x(k), which are not functions of the coefficients. It is clearly a linear function in the parameters a(k) and b(k), thus gradient based algorithms can be applied to recursively update the coefficients [4].

Equation-error is given by $e_e(k) = d(k) - y_e(k)$, it is so called because it is the subtraction of two difference equations. Performance criterion, the mean-square equation error (MSEE), based on prediction error, $e_e(k)$ will be a quadratic surface with only a

global minima, resulting in a unique estimate of the coefficients. In many ways the properties of the equation-error IIR filters and FIR filters (where A(Z,k) = 0, in equation error IIR) are similar and have similar adaptive properties [5].

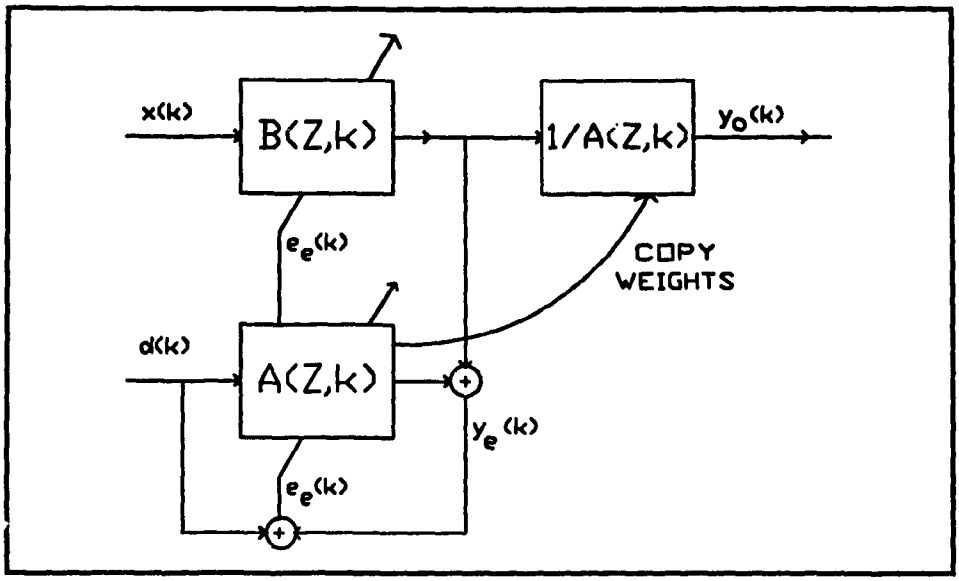

Figure 3.2. Equation-error adaptive filter.

Figure 3.2 shows that the coefficient are estimated at each update in an all-zero form and then are copied to the pole section of the filter. But, before the coefficients can be copied, the inverse of 1 - A(Z,k) must be monitored for stability at each instant of time. If the zeros of 1 - A(Z,k) are not within the unit circle then some method of projecting them back in the unit circle must be used.

Equation 3.2 can be compactly written as the inner product of the parameter vector, $\boldsymbol{\theta}(k)$, and the signal vector, $\boldsymbol{\phi}_e(k)$,

$$y_e(k) = \theta^T(k)\phi_e(k) \tag{3.3}$$

where,

$$\theta(k) = [a_1(k), a_2(k), \cdots, a_{N-1}(k), b_0(k), \cdots, b_{M-1}(k)] \tag{3.3a}$$

$$\phi_e(k) = [d(k-1), d(k-2), \cdots, d(k-N+1), x(k), x(k-1), \cdots, x(k-M+1)] \tag{3.3b}$$

This equation has a form of linear regression, which is commonly used in statistical modelling, where θ contains the parameters to be estimated and ϕ_e is the regression vector containing the data. Since the regression is linear, (i.e. the data vector is independent of the parameter to be estimated) many of the techniques used in statistical modelling can be used to obtain the optimal parameter vector [6].

3.1.2 Output-Error Formulation

Figure 3.3 and the difference equation (Eq 3.4) describe the output-error formulation of the direct form implementation of the adaptive IIR filter, where the subscript o indicates output-error formulation.

$$y_o(k) = \sum_{m=1}^{N-1} a_m(k) y_o(k-m) + \sum_{m=0}^{M-1} b_m(k) x(k-m) \tag{3.4}$$

Unlike the equation-error formulation, output-error formulation is a single-input single-output system. In this formulation the coefficients are recursively updated in a pole-zero form. Of course, as before stability monitoring is still essential and may be done similarly as in the equation-error formulation. From Eq 3.4, it is easily seen that $y_o(k)$ is

a function of its previous values, that is, there exists feedback. This feedback greatly influences the form and complexities of the adaptive algorithms used to find optimal coefficients.

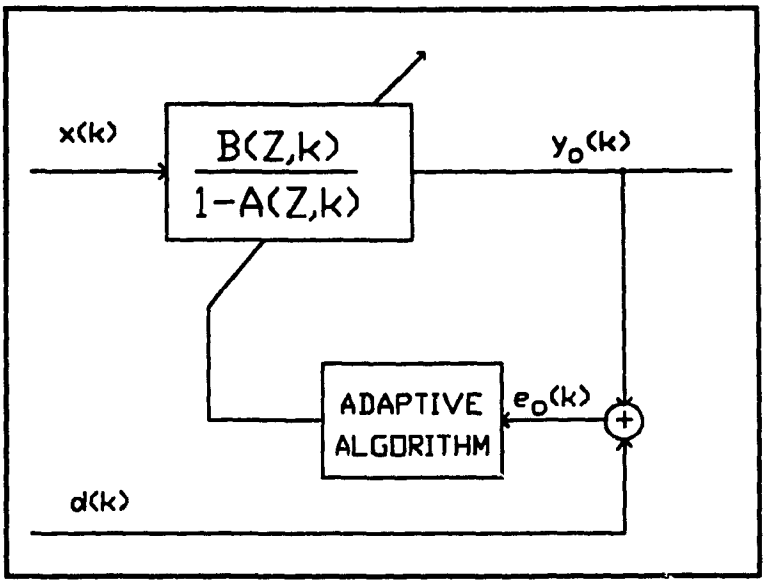

Figure 3.3. Output-error formulation of adaptive filter.

Now, as in equation-error formulation, Eq 3.4 can be rewritten in the form of inner product of the parameter vector, θ, and the data vector, ϕ_o,

$$y_o(k) = \theta^T(k)\phi_o(k), \tag{3.5}$$

where $\phi_o(k)$ is

$$\phi_o(k) = [y_o(k-1), y_o(k-1), \cdots, y_o(k-N+1), x(k), x(k-1), \cdots, x(k-M+1)], \tag{3.5a}$$

and θ is as in Eq 3.3a. It is easily seen that $y_o(k)$ is a nonlinear function of θ, since $y_o(k-m)$ depend on previous values of the coefficients, namely $\theta(k-m)$, m= 1,2,...,k-M-1. Thus, the prediction error $e_o(k) = d(k) - y_o(k)$, is also a nonlinear function of the parameters and the mean square output error, MSOE, will not be a quadratic surface. It will have local minima. If gradient search algorithms are employed to find the optimal coefficient, they may converge to one of the local minima yielding inaccurate results.

As was mentioned earlier, the adaptive output-error formulation is somewhat more complicated than the equation-error formulation. Equation-error, the simpler formulation, leads to an estimate of the coefficients that are biased away from the true value. It can be shown that the bias is directly influenced by the power of the noise at the input [2]. In the more complicated formulation, there exist multiple local minima and the estimate of the coefficients may converge to a local minima yielding erroneous results [2]. One can then realize the tradeoff that exists between a biased estimate and the convergence to a local minima.

3.2 Formulation of Constrained Notch Filter

A natural course for the formulation of the CANF is in the frequency domain as we are discriminating signals based on frequency, i.e. narrowband (NB) or broadband (BB). To start with, all the filter requirements will be developed in the steady-state and then the steps required to obtain appropriate filter parameters will be shown. The goal is to achieve a filter that will reject a NB (or sinusoidal) component in the observed time series and reveal the BB component, which can then be subtracted from the original signal to reveal the NB component.

The observed time series can be represented as the sum of the NB component and noise (BB component) as

$$y(k) = x(k) + n(k), \tag{3.6}$$

where x(k) is the undisturbed NB component and n(k) is the additive noise, sampled at time k. In the following formulation, the two components of the observed signal, y, are assumed to be independent. Consider x as a sum of P sinewaves

$$x(k) = \sum_{i}^{P} A_i \sin(\omega_i k) \tag{3.7}$$

then, the observed signal y in frequency domain is given by the power spectral density (PSD),

$$S_y(\omega) = S_x(\omega) + S_n(\omega) = \sum_{i}^{P} \frac{A_i^2}{2}\delta(\omega - \omega_i) + S_n(\omega). \tag{3.8}$$

where $S_n(\omega)$ is the PSD of the noise. Now, consider a digital filter, $H(Z,\theta)$, as in Fig 3.4 with input y, where θ is the parameter vector that characterizes the filter, much like the weight of each tap in the FIR-ALE. The output power of such a filter, with the input y, is defined by [7]

$$J(\theta) = \frac{1}{2\pi}\int_{-\pi}^{\pi} |H(Z,\theta)|^2 S_y(\omega) d\omega \tag{3.9}$$

$$J(\theta) = \frac{1}{2\pi}\int_{-\pi}^{\pi} |H(Z,\theta)|^2 S_n(\omega) d\omega + \frac{1}{2\pi}\int_{-\pi}^{\pi} |(H(Z,\theta)|^2 \sum_{i-1}^{P} \sigma_i^2 \delta(\omega - \omega_i) d\omega. \tag{3.9b}$$

Replacing Z by $e^{j\omega}$ this can be written as

$$J(\theta) = A(\theta) + B(\theta) \tag{3.10}$$

where

$$A(\theta) = \frac{1}{2\pi}\int_{-\pi}^{\pi} |H(e^{j\omega},\theta)|^2 S_n(\omega)d\omega \tag{3.11}$$

and

$$B(\theta) = \frac{1}{2\pi}\sum_{i}^{P} \sigma_i^2 |H(e^{j\omega_i},\theta)|^2. \tag{3.12}$$

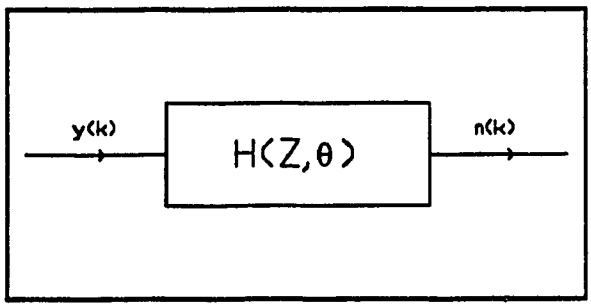

Figure 3.4. Digital filter H(Z, θ).

In order for the digital filter of Fig 3.4, to function as a notch filter, the output power must be minimized with respect to the filter parameters such that the frequency response has zeros at the sinusoidal frequencies. As can be seen from Eq. 3.10, ideally it is desired to minimize B(θ) alone, as this is the portion of the signal that we wish to eliminate at the output. But, in the presence of noise, A(θ) is also a function of θ, hence minimizing J(θ) w.r.t. θ will affect the results. The effect of A(θ) is proportional to the SNR, or the

power of the noise [8]. Thus, one would expect that the contribution of noise power affects the determination of the parameters, θ, unless special measures are taken.

One way to eliminate the affect of noise is to keep the $A(\theta)$ term constant. Since no *a priori* information is available about the additive noise, the filter is constrained to have a response of unity at all frequencies other than the sinusoid frequencies, where it will be zero as described by Eq 3.13. In this manner, the filter will have no effect on the shape of the noise. Fig 3.5 shows the frequency response of the digital filter.

$$H(e^{\omega_i},\theta) = \begin{cases} 0 & \omega=\omega_i \quad \textit{for } i=1,2,\cdots,P \\ 1 & \textit{otherwise} \end{cases} \tag{3.13}$$

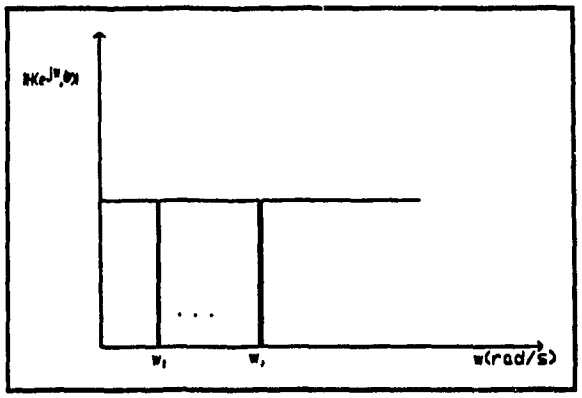

Figure 3.5. Frequency response of the desired CANF

Using the desired filter response of Eq 3.13 in the output power equation, (Eq 3.10), yields

$$J(\theta) = A + B(\theta) \tag{3.14a}$$

$$J(\theta) = \frac{1}{2\pi}\int_{-\pi}^{\pi} S_n(\omega)d\omega + \sum_{i}^{P}\frac{\sigma_i^2}{2\pi}|H(e^{j\omega_i},\theta)|^2 \qquad (3.14)$$

where it is seen that the first term, A, becomes independent of the parameter vector θ and hence minimizing J will indeed minimize $B(\theta)$ only. The obtained parameter vector, θ, corresponds to an ideal notch filter, $H(Z,\theta)$. It must be noted that the above reasoning used for eliminating the dependence of θ in A, holds only for the ideal case. That is, as the bandwidth of the notch becomes larger, the more effect will the noise shape have on the parameterization, θ [8].

3.2.1 Pole-Zero Configuration of CANF

Before any adaptive procedure for estimating the parameter vector, θ, of $H(Z,\theta)$ can be developed, a structural form of the filter must exist. That is , constraints on the pole-zero placement exist. In order to derive this pole-zero configuration, we will start with NB signals for x in Eq 3.1 and then take the limiting operation on the bandwidth to yield a sum of sinusoids. A NB signal can be considered as the output of the filter represented by the rational function [9]

$$H(Z) = \frac{A\prod(Z-e^{-\mu_i+j\omega_i})(Z-e^{-\mu_i-j\omega_i})}{\prod(Z-e^{-\gamma_i+j\omega_i})(Z-e^{-\gamma_i-j\omega_i})} \qquad (3.15)$$

where the zeros satisfy the following conditions:

$$\mu_i = (\gamma_i + SNR_i)^{\frac{1}{2}} \qquad i = 1,2,\cdots,P \qquad (3.16)$$
$$\mu_{i+P} = \mu_i$$

where P = the number of NB signal, SNR_i = the signal to noise ratio and γ_i = bandwidth of i'th NB signal. Now, taking the limit of Eq 3.15 as the bandwidths, γ_i's, approach zero, we obtain

$$\lim_{\gamma_i's \to 0} H(Z) = \frac{A\Pi(Z-e^{j\omega_i})(Z-e^{-j\omega_i})}{\Pi(Z-e^{j\omega_i})(Z-e^{-j\omega_i})} = H'(Z) \tag{3.17}$$

In the limit, it is observed that the poles and zeros of $H'(Z)$ approach the unit circle at the same location. In order to maintain stability and to keep the poles and zeros from cancelling each other, the poles must be kept away from the unit circle.

From the preceding arguments, an approximate IIR notch filter realization can be $H'(Z)$ with the constraints that each pole-zero pair must lie on the same radial line and that the i'th pole must lie between the origin and i'th zero. In this way, it will be possible to maintain the above formulation without cancelling the poles and zeros. That is,

$$P_i = \alpha Z_i = \alpha r e^{j\omega_i}, \tag{3.18}$$

where $0 \le \alpha \le 1$ and $0 \le r \le 1$. The constrained notch filter can then be given as

$$H'(Z) = \frac{A\Pi(1-Z_iZ^{-1})(1-Z_i^*Z^{-1})}{\Pi(1-P_iZ^{-1})(1-P_i^*Z^{-1})} = \frac{A\Pi(1-Z_iZ^{-1})(1-Z_i^*Z^{-1})}{\Pi(1-\alpha Z_iZ^{-1})(1-\alpha Z_i^*Z^{-1})} \tag{3.19}$$

and the * denotes complex conjugate.

3.2.2 Consideration of the Choice of α

As can be seen from the above structural analysis of the notch filter, performance is better as α approaches 1. The closer α is to unity, the flatter is the response of H(Z) in the spectrum away from the frequencies of the sinusoids to be rejected.

Let $\alpha = 0$, then the transfer function of the CANF becomes

$$H(Z) = 1 + \cdots + \theta_{2P} Z^{-2P} \tag{3.20}$$

This is clearly an all-zero filter (moving average process) as was mentioned in the IIR section. It was said that this is inadequate in modelling a noisy process, without extending the order.

Let $\alpha = 1$; this is the case of poles and zeros cancelling each other, and corresponds to a filter that has a constant magnitude response almost all over. This can be viewed as a special ARMA model that has extremely sharp notches (the ideal case), however the model becomes unstable.

For practical purposes (i.e. stability, performance and convergence) the extreme values of α are not of interest and intermediate values must be used.

3.3 Derivation of Adaptive Algorithm

This section is devoted to the derivation of the algorithm to determine the filter parameter vector, θ. Two separate paths will be taken to obtain the filter parameterization. The first will follow that of [8], the approach of estimating 2P parameters, where P is equal to the number of sinusoids. The second approach will estimate only P parameters, following results of [7].

3.3.1 2P-Parameter Estimation

In order to develop a 2P-parameter estimation algorithm, a pole-zero direct form implementation of the CANF will be utilized, as shown in Fig 3.6. First, we will give a few definitions that will be used in the subsequent derivation:

$$\Lambda = diag(\alpha, \alpha^2, \cdots, \alpha^{2P}) \tag{3.21}$$

$$\theta = [\theta_1, \theta_2, \cdots, \theta_{2P}]^T \tag{3.22}$$

$$\tilde{Y}(k) = [\tilde{y}(k-1), \tilde{y}(k-2), \cdots, \tilde{y}(k-2P)]^T \tag{3.23}$$

$$\tilde{N}(k) = [\tilde{n}(k-1), \tilde{n}(k-2), \cdots, \tilde{n}(k-2P)]^T \tag{3.24}$$

where (see Fig. 3.6)

$$\tilde{n}(k) = \hat{n}(k) + \tilde{N}^T(k)\Lambda\theta(k-1)$$

Referring to Fig 3.6, the input-output relationship can be written as

$$\tilde{y}(k) = y(k) + \Lambda\theta^T\tilde{Y}(k) \tag{3.25}$$

and

$$\tilde{n}(k) = \tilde{y}(k) - \theta^T\tilde{Y}(k) \tag{3.26}$$

Now, for the adaptive parameter estimating scheme, the Gauss-Newton [10] method is used. The parameters can be updated by the following two equations,

$$\theta(k) = \theta(k-1) + \mu\bar{R}^{-1}(k)\Psi(k) \tag{3.29}$$

and

$$R(k) = R(k-1) + \mu[\Psi(k)\Psi^T(k) - R(k-1)], \tag{3.28}$$

where θ is the parameter vector to be estimated, Ψ is the gradient (defined in the next paragraph) μ is the rate of adaptation and R is the covariance matrix.

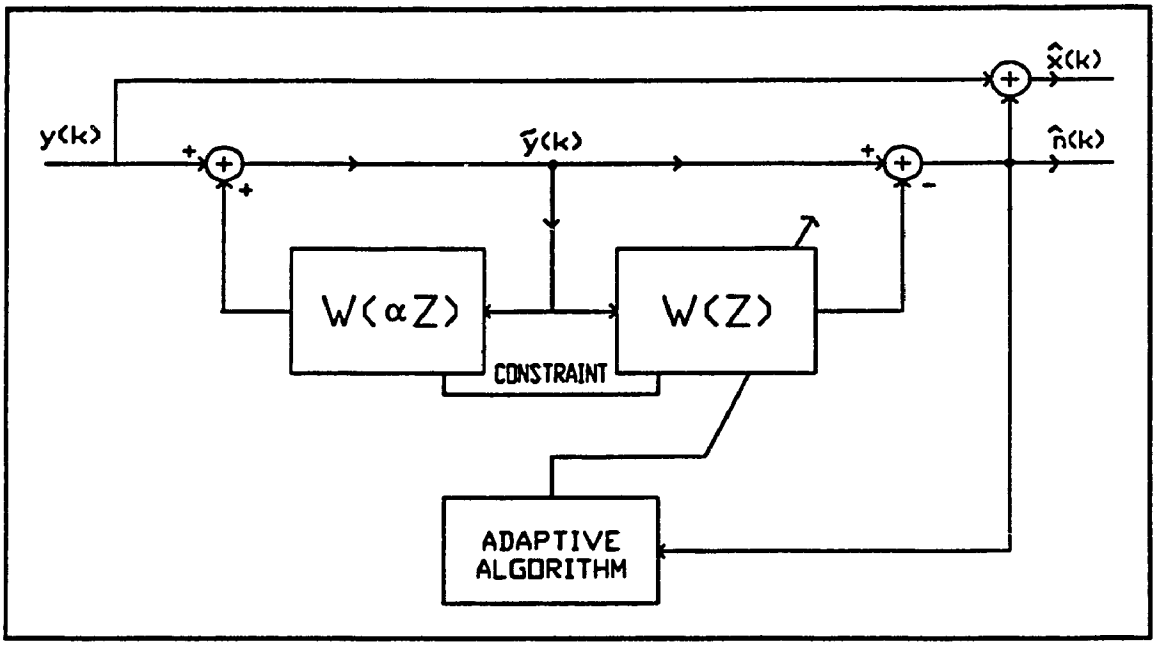

Figure 3.6. Adaptive Line Enhancer (ALE) using the CANF.
[adapted from [8]]

As the number of sinusoid gets large, the inverse of the covariance matrix, R, becomes computationally cumbersome. To avoid such computation problems, using the Hessian matrix and the matrix inversion lemma [10], the recursive update can be approximated by

$$\theta(k) = \theta(k-1) + K(k)\hat{n}(k) \tag{3.29}$$

$$K(k) = \left[\frac{P(k-1)\Psi(k)}{\lambda(k)+\Psi^T(k)P(k-1)\Psi(k)}\right] \tag{3.30}$$

$$P(k) = \left[P(k-1)-\frac{P(k-1)\Psi(k)\Psi^T(k)P(k-1)}{\lambda(k)+\Psi^T(k)P(k-1)\Psi(k)}\right] \tag{3.31}$$

$$\lambda(k) = \lambda_o\lambda(k-1)+(1-\lambda_o) \tag{3.32}$$

where λ is the forgetting factor and $\Psi(k)$ is the gradient defined by

$$\Psi(k) = -\frac{\partial\hat{n}(k)}{\partial\theta} = \left[-\frac{\partial\hat{n}(k)}{\partial\theta_1},-\frac{\partial\hat{n}(k)}{\partial\theta_2}, \cdots ,-\frac{\partial\hat{n}(k)}{\partial\theta_{2P}}\right]^T \tag{3.33}$$

$$\Psi(k) = [\tilde{y}(k-1)-\alpha\tilde{n}(k-1),\tilde{y}(k-2)-\alpha^2\tilde{n}(k-2), \cdots ,\tilde{y}(k-2P)-\alpha^{2P}\tilde{n}(k-2P)]^T \tag{3.33a}$$

$$\Psi(k) = [\tilde{Y}(k)-\Lambda\tilde{N}(k)] \tag{3.33b}$$

3.3.2 P-Parameter Estimation

As was mentioned in earlier sections, two desirable properties of the CANF are to have the zeros on the unit circle and to have the zeros and poles of the i'th sinusoid lie on the same radial line. The above algorithm estimates 2P parameters but it does not guarantee that the zeros will lie on the unit circle. The algorithm, to be presented next, guarantees that the zeros will lie on the unit circle. A necessary but not sufficient condition for zeros to lie on the unit circle is that the coefficients of the numerator of the CANF transfer function be symmetric [7] as described by

$$H(Z,\theta) = \frac{1+\theta_1 Z^{-1}+\theta_2 Z^{-2}+\cdots+\theta_P Z^{-P}+\cdots+\theta_1 Z^{-2P+1}+Z^{-2P}}{1+\alpha\theta_1 Z^{-1}+\alpha^2\theta_2 Z^{-2}+\cdots+\alpha^P\theta_P Z^{-P}+\cdots+\alpha^{2P-1}\theta_1 Z^{-2P+1}+\alpha^{2P}Z^{-2P}} \quad (3.34)$$

where α is as defined earlier. The effect of reducing the number of parameters by one half, results in a significant decrease in computations. Further advantage will be discussed subsequently. Steps in estimating the filter parameters will now be shown. As in the 2P algorithm, the Gauss-Newton recursive prediction error, RPE, algorithm will be used. The output of the filter is denoted by n(k) and the parameter vector θ is now P in length

$$\theta = [\theta_1, \theta_2, \cdots, \theta_P]^T \quad (3.35)$$

The input-output relationship can be written as (where q^{-1} is the delay operator as defined earlier)

$$\hat{n}(k) = \frac{A(q^{-1})}{A(\alpha q^{-1})} y(k) \quad (3.36)$$

This can be expanded as

$$\hat{n}(k) = y(k) + y(k-2P) + \alpha^{2P}\hat{n}(k-2P) - \Phi^T(k)\theta(k-1) \quad (3.37)$$

where

$$\Phi(k) = [\Phi_1(k), \Phi_2(k), \cdots, \Phi_P(k)] \quad (3.38)$$

$$\Phi_i(k) = \begin{cases} -y(k-i) - y(k-2P+i) + \alpha^i n(k-i) + \alpha^{2P-i} n(k-2P+i) & \text{for } i \le i \le P-1 \\ -y(k-P) + \alpha^P n(k-P) & \text{for } i = P \end{cases} \quad (3.38a)$$

Let the gradient of $\hat{n}(t)$ w.r.t. θ be defined as,

$$\Psi(k) = [\Psi_1(k), \Psi_2(k), \cdots, \Psi_P(k)]^T \tag{3.39}$$

where

$$\Psi_i(k) = -\frac{\partial n(k)}{\partial \theta_i}$$

Therefore, differentiating both sides of Eq 3.36 w.r.t. θ, we obtain

$$A(\alpha q^{-1})\frac{\partial n(k)}{\partial \theta_i} + n(k)\frac{\partial A(\alpha q^{-1})}{\partial \theta_i} = \frac{\partial A(q^{-1})}{\partial \theta_i} y(k) \tag{3.40}$$

$$A(\alpha q^{-1})\frac{\partial n(k)}{\partial \theta_i} + n(k)[\alpha^i q^{-1} + \alpha^{2P-i} q^{-2P+i}] = [q^{-1} + q^{-2P+i}]y(k) \quad \text{for} \quad i = 1,2,\cdots,P-1$$

$$A(\alpha q^{-1})\frac{\partial n(k)}{\partial \theta_i} + \alpha^P n(k-P) = y(k-P) \quad \text{for} \quad i = P \tag{3.41}$$

which leads to

$$A(\alpha q^{-1})\frac{\partial n(k)}{\partial \theta_i} = y(k-i) + y(k-2P+i) + \alpha^i n(k-i) - \alpha^{2P-i} n(k+2P+i) \text{ for } i = 1,2,\cdots,P-$$

$$A(\alpha q^{-1})\frac{\partial n(k)}{\partial \theta_P} = y(k-p) - \alpha^P n(k-p) \quad \text{for } i = P \tag{3.42}$$

$$A(\alpha q^{-1})\frac{\partial n(k)}{\partial \theta_i} = -\Phi(k) \qquad i = 1,2,\cdots,P \tag{3.43}$$

$$\Psi(k) = \frac{\Phi(k)}{A(\alpha q^{-1})} \tag{3.44}$$

From the above expressions we find that the gradient vector, Eq 3.44, has the same form as the regression vector in Eq 3.38, it is simply the regression vector filtered by $1/A(\alpha q^{-1})$. The actual coefficient values of the filter, $A(\alpha q^{-1})$, are not known and hence its immediate estimate is used instead. The adaptive algorithm can now be described by the following recursive equations

$$\hat{n}(k) = y(k) + y(k-2P) - \alpha^{2P}\hat{n}(k-2P) - \Phi^T(k)\hat{\theta}(k-1) \tag{3.45}$$

$$P(k) = \frac{1}{\lambda(k)}\left[P(k-1) - \frac{P(k-1)\Psi(k)\Psi^T(k)P(k-1)}{\lambda(k) + \Psi^T(k)P(k-1)\Psi(k)}\right] \tag{3.46}$$

$$\hat{\theta}(k) = \hat{\theta}(k-1) + P(n)\Psi(k)\hat{n}(k) \tag{3.47}$$

$$\bar{n}(k) = y(k) + y(k-2P) - \alpha^{2P}(k)\bar{n}(k-2P) - \Phi^T(k)\hat{\theta}(k) \tag{3.48}$$

Equation 3.48 is used to improve the convergence rate, (called *a posteriori* prediction error) since it results in better prediction error estimates then $\hat{n}(k)$.

Unlike the first algorithm of 2P parameter, this algorithm uses a time varying value for the debiasing parameter, α. It is forced to increase with an exponential rate described by

$$\alpha(k+1) = \alpha(k)\alpha_o + (1-\alpha_o)\alpha(\infty) \tag{3.49}$$

Note that the rate of increasing α is dependent on the value of α_o. The rationale for using a time varying debiasing parameter is two fold. First, if at the start of the adaptation scheme α is chosen very close to unity, then the bandwidth of the notch will be very narrow. It is possible that in such a case the sinewave frequency will not fall within the bandwidth and thus, the algorithm may not converge to the desired transfer function. Therefore, at the start of the adaptation, a small value of α is chosen and then it is exponentially increased to a predetermined final value close to unity. In this manner, the sinusoid is forced to fall within the notch bandwidth at the start and then the notch bandwidth is gradually closed while fine tuning the filter parameters corresponding to the correct frequency. Secondly, at the beginning of the adaptation, the estimate of $A(\alpha q^{-1})$ is poor and thus its effect on the gradient in Eq 3.44 can be large. This effect may be diminished by forcing the poles near the origin during the transient values of k [6].

3.4 Performance

There are several advantages in the P-parameter (henceforth, referred to as P algorithm) algorithm over the 2P-parameter algorithm (henceforth referred to as 2P algorithm), the foremost of which is the ability to achieve narrower notches at the desired frequency. In the P algorithm, the bandwidth of each pole-zero pair can be evaluated as (the subscript refers to the algorithm)

$$BW_P = (1-\alpha)\pi. \tag{3.50}$$

In the case of the 2P algorithm, the zeros do not always converge to a modulus of unity, but converge to a modulus r less than unity. Thus, the bandwidth is a little larger expressed by

$$BW_{2P} = (1 - r\alpha)\pi. \tag{3.51}$$

Narrow bandwidths are important as was discussed earlier. That is, as the notches become wider, the influence of the noise on the parameterization, θ, can be extreme. In this respect, P algorithm is much more efficient. In this algorithm the zeros are constrained to be on the unit circle and the notch bandwidths are determined by the value of α. To cope with the varying bandwidths, the time varying debiasing coefficient (as defined earlier in section 3.3.2) can be employed. In this case at the start of the adaptation, a large notch bandwidth can be used to minimize the transient time of the filter.

Consider the case, where the bandwidth (we will only consider the case of a single notch for this explanation) is allowed to be very small and is slightly biased. In this case, as described in frequency domain in Fig 3.7, the notch will not fall on the sinusoid that it is supposed to suppress. That is, the notches will be too sharp and narrow for any cancellation to be effective. Now consider the case, where the notch point is created at some frequency near the desired value and the notch is not as narrow, as illustrated in Fig 3.8. In this case there will be a considerable cancellation of the sinusoid. A part of the notch will fall on the sinusoid and thus some cancellation will take place. From this discussion it is apparent that there exists some tradeoff between the bandwidths of the notches and the notch points. Thus, it may not be desirable to have very sharp notches, if the notch points are biased. If the purpose of these two algorithms is for signal (NB

component) suppression, then the tradeoff in the parameterization and the bandwidth must be considered. This, of course, will be especially critical in the case of cascade connection (to be discussed in the next chapter) where the first sinewave must be sufficiently cancelled in order for the second section, in the cascade connection, not to create a notch at the same frequency value. More on the cascade connection will be discussed in the following chapter.

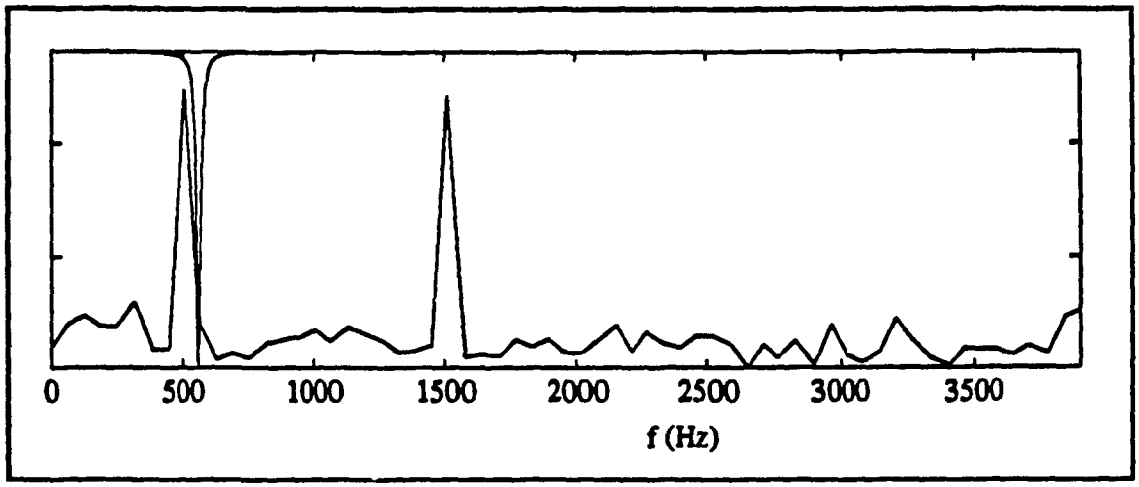

Figure 3.7. Performance of Narrow notch.

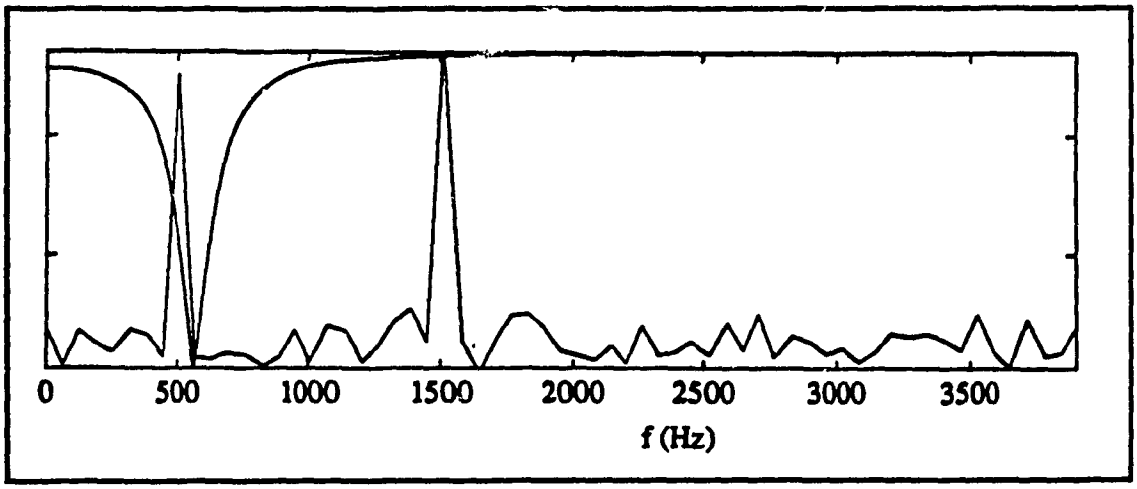

Figure 3.8. Performance of Wide notch.

Since the bandwidth is controlled by the value of the debiasing coefficient (α) then in the 2P algorithm, the bandwidth of the notches are fixed to be small throughout the adaptation process. Thus, if at start of the algorithm, the notches are too narrow and not near the correct frequencies, then the algorithm may not sense the presence of the sine waves and may converge to erroneous values. For this reason, the 2P algorithm requires that α be chosen such that the notches are not too narrow. But in the P algorithm, the debiasing parameter, α, is exponentially increasing, the rate and the final value is user controlled. Therefore, the algorithm can be started with a bandwidth as wide as desired and be forced to close as slowly or as fast as desired. This would ensure that the notches converge to the correct values of the frequencies.

ACCURACY: P algorithm is minimally parameterized. That is, by reducing the number of parameters to estimate P, we have reduced the search space to a small subset of all possible filters. By doing this, it is possible to expect the parameterization to approach CRB [6]. Such claims cannot be made of the 2P algorithm.

STABILITY: In the convergence analysis on the class of RPE algorithms [5], it is shown that stability monitoring is a necessity. This entails, that at every iteration, the poles of the CANF be checked for stability and if not stable then they must be projected back to a stable system [11]. P algorithm exhibits high stability and does not need to have poles monitored, which is different from RPE (recursive prediction error) algorithms in general, and enables the use of poles close to the unit circle. In the 2P algorithm, this is not the case; stability must be monitored and a projection scheme needs to be used. Both of the

algorithms use the Gauss-Newton recursive procedures. The difference may be due to the fact that the initial values of the poles are not close to the unit circle (i.e. they are forced close to the origin by the small value of the debiasing coefficient).

3.5 References

1. B. Widrow et al., "Adaptive Noise Cancellation: Principles and Applications", Proc. IEEE, vol 63, pp. 1692-1716, Dec. 1975.

2. J.H. Shynk, "Adaptive IIR Filtering", IEEE ASSP Magazine, pp. 4-21, April 1989.

3. L. Marple Jr., Digital Spectral Analysis with Application, Englewood Cliffs, NJ, Prentice Hall Inc., 1987.

4. B. Widrow, J.M. McCool, M.G. Larimore and C.R. Johnson Jr., "Stationary and nonstationary learning characteristics of the LMS adaptive filter," Proc. IEEE, vol. 64, August 1976

5. J.V. Candy, Signal Processing: The Modern Approach, USA: McGraw-Hill, 1988.

6. D.M. Bates and D.G. Watts, Nonlinear Regression Analysis and its Applications, USA: John Wiley & Sons, 1988.

7. A. Nehorai, "A Minimal Parameter Adaptive Notch Filter With Constrained Poles and Zeros", IEEE Trans. on Acoustics, Speech and Signal Processing, vol ASSP-33, pp. 983-996, August 1985.

8. D.V. Rao and S.Y. Kung, " Adaptive Notch Filtering for the Retrieval of Sinusoids in Noise", IEEE Trans. on Acoustics, Speech and Signal Processing vol ASSP-32, August 1984.

9. S.T. Alexander, E.H. Satorius and J.R. Ziedler, "Linear Prediction and Maximum Entrophy Spectral Analysis of finite Band ·idth Signals in Noise", in Proc. IEEE, vol 70, p. 188-192, 1978.

10. T. Soderstrom, L. Ljung and I. Gustavasson, "A Theoretical Analysis of Recursive Identification Methods," Automatica, vol 14, pp. 231-244, 1978.

11. L. Ljung and T. Soderstrom, Theory and Practice of Recursive Identification, Cambridge, MA: M.I.T. Press 1983.

Chapter 4

Consideration of Short Data

In the previous chapter, the concept of Constrained Adaptive Notch Filter was presented under the assumption that large amount of data is available. Assumptions of this type are true for most existing methods used in narrow band (NB) suppression/ enhancement problem [1]. Often, in practice, situations arise where the available data is extremely short, i.e. the data length available is much shorter than required for the convergence of adaptive methods commonly used for high resolution spectral analysis. The ABR application considered in this thesis is a problem associated with short data. Thus, this chapter will consider the general concept of short data spectral analysis using the CANF as the building block.

As was described in the previous chapter, CANF is formulated as a 2P order autoregressive-moving average model (ARMA). When implemented in the direct form, problems associated with adaptive convergence become major obstacles, particularly with short data. This chapter will serve to analyze the error surface structure of the CANF in the direct and the cascade form. Using the analysis of the error surface, it will be shown how to handle the problem of short data to obtain improved performance. As a consequence of the error-surface analysis two modified cascade structures will be introduced.

4.1 Error Surface Analysis

Very often, it is necessary to examine the error surface of an adaptive system to realize its convergent behaviour [2]. The choice of the adaptive scheme used may depend on the characteristics of the error surface. For example, if the error surface is multimodal than a simple gradient based algorithm may not be a suitable choice [2,3]. One therefore, needs to evaluate the modality of the error surface associated with the system of concern.

It is known, as was described in the IIR section of the previous chapter, that ARMA model of the output-error formulation type behaves such that the associated error surfaces may be multimodal [4]. Such behaviour leads to the implementation of adaptive algorithms having the requirement that sufficient *a priori* knowledge of the parameter be available to use as the starting guess. Without such knowledge, the most generally used gradient algorithms may not converge to the correct solution, but instead converge to one of the local minima. Thus, in this section, expressions for error surface associated with constrained notch filters are presented. Behaviour of the error surface in the single as well as the multiple notch CANF module will be shown.

From Fig. 3.6, the error signal is given as the output of the notch filter. The objective of the adaptive algorithm is to adjust the parameters, θ, such that the error signal (the output of the filter) is minimized in the minimum mean square sense, MMSE. Now, the expected value of the squared error can be written as [5,6],

$$E[e^2] = \frac{1}{2\pi j}\oint |H(Z,\theta)|^2 \{S_{xx} + S_{nx} + S_{nn}\}\frac{dZ}{Z}, \qquad (4.1)$$

where S_{xx}, S_{nn} *and* S_{nx} are the PSDs of the input signal component x and n and their cross PSD, respectively and the path of integration is the unit circle in the Z-plane. Equation 4.1 can now be used to determine the modality of the error surface by differentiating it with respect to θ and then setting it equal zero. If the result yields a linear equation in the parameter θ then the error surface can be said to be unimodal, otherwise there exist local minima. In Eq. 4.1 only $|H(Z,\theta)|^2$ is a function of the parameter θ thus, differentiating with respect to θ_i results in a set of equations

$$\frac{\partial E[e^2]}{\partial \theta_i} = \frac{1}{2\pi j}\oint \frac{\partial |H(Z,\theta)|^2}{\partial \theta_i}\{S_{xx} + S_{nx} + S_{nn}\}\frac{dZ}{Z}, \qquad (4.2)$$

which are used to determine the modality of the error surface.

At this point, the idea of sufficient and insufficient filter models should be clarified. By sufficient filter model, it is meant that, the filter order is chosen such that it matches the input signal. That is, the filter order is twice the number of sinusoids or NB components at the input. And by insufficient, we talk about a mismatch, that is, the filter order is less than twice the number of sinusoids or NB components at the input.

4.1.1 Sufficient Model

Double Notch Case

Some insight into the behaviour of the multiple notch error surface can be obtained by examining the behaviour of the double notch case. In order to keep this explanation mathematically simple as well as to be able to graphically visualize, it will

be necessary to use the filter containing the minimal number of parameters (i.e mirror symmetry of the parameters) as described in the previous chapter. This by no means reduces the generality of the analysis. The filter expression is then written as

$$H(Z) = \frac{A(Z)}{B(Z)} = \frac{1+\theta_1 Z^{-1}+\theta_2 Z^{-2}+\theta_1 Z^{-3}+Z^{-4}}{1+\alpha\theta_1 Z^{-1}+\alpha^2\theta_2 Z^{-2}+\alpha^3\theta_1 Z^{-3}+\alpha^4 Z^{-4}} \tag{4.3}$$

Now, dropping the argument Z, the derivative of the square of the magnitude is

$$\frac{\partial H(Z)|^2}{\partial\theta_1} = \frac{BB^*A^*(Z^{-1}+Z^{-3})+BB^*A(Z+Z^3)-AA^*B^*(\alpha Z^{-1}+\alpha^3 Z^{-3})-AA^*B(\alpha Z+\alpha^3}{(BB^*)^2} \tag{4.4a}$$

$$\frac{\partial H(Z)|^2}{\partial\theta_2} = \frac{BB^*A^*Z^{-2}+BB^*AZ^2-AA^*B^*\alpha^2 Z^{-2}-AA^*B\alpha^2 Z^2}{(BB^*)^2}, \tag{4.4b}$$

where * represents complex conjugate and A and B are as in Eq. 4.3. Replacing Eq. 4.4 in Eq 4.1, and performing some manipulations, yields a set of non linear equations: [5]

$$\oint\frac{[Z^{-1}A^*(1+Z^{-2})+ZA(1+Z^2)](S_{xx}+S_{nx}+S_{nn})}{BB^*}\frac{dZ}{Z} = \oint\frac{\alpha AA^*[BZ(1+\alpha^2Z^2)+B^*Z^{-1}(1+\alpha^2Z^{-2})](S_{xx}+S_{nx}+S_{nn})}{(BB^*)^2}\frac{dZ}{Z} \tag{4.5a}$$

$$\oint\frac{(AZ^2+A^*Z^{-2})(S_{xx}+S_{nx}+S_{nn})}{BB*}\frac{dZ}{Z} = \oint\frac{\alpha^2AA^*(BZ^2+B^*Z^{-2}(S_{xx}+S_{nx}+S_{nn})}{BBB^*B^*}\frac{dZ}{Z} \tag{4.5b}$$

The non linearity of Eqs. 4.5a and 4.5b indicates the multimodality of the error surface for the two notch application. Thus, it is expected that the error surface may contain local

minima. Extension of the above explanation to the multi-notch application is straight forward.

The expected squared error in Eq. 4.1 can also be rewritten as [1]

$$E\{e^2\} = \sum_{i=1}^{m} \frac{A_i^2}{2} |H(e^{j\omega_i})|^2 + \oint |H(Z,\theta)|^2 S_{nn} \frac{dZ}{Z} \tag{4.6}$$

for the case of m sinusoids buried in uncorrelated noise, where A_i represents the amplitude of the i'th sinusoid. Now, consider a noise free case of m sinusoids, the above Eq. 4.6 is written as

$$E\{e^2\} = \sum_{i=1}^{2} \frac{A_i^2}{2} |H(e^{j\omega_i},\theta)|^2 \tag{4.7}$$

where $H(e^{j\omega_i},\theta)$ is obtained by substituting $e^{j\omega_i}$ for Z in the filter transfer function.

Fig. 4.1 illustrates the error surface of a fourth order CANF evaluated as in Eq. 4.7. It should be noted that the possible range of θ_1 is in the interval [-4,4] and for θ_2 it is in the interval [-2,6], however, the digital filter is not stable in this full range [5]. Fig. 4.1 shows the error surface in the neighbourhood of the global minima. The multimodality (or the existence of local minima) of the multiple notch direct form implementation is evident from Fig. 4.1. There exists many possibilities for the parameter, θ, to which an adaptive algorithm may converge , yielding erroneous results. From Fig. 4.1, it can be said that for the direct form implementation of the multiple notch CANF, a judicious choice of initial parameter value in the adaptive algorithm is required in order to converge to the global minimum. That is, we need to know the filter coefficients in the neighbourhood of the global minimum.

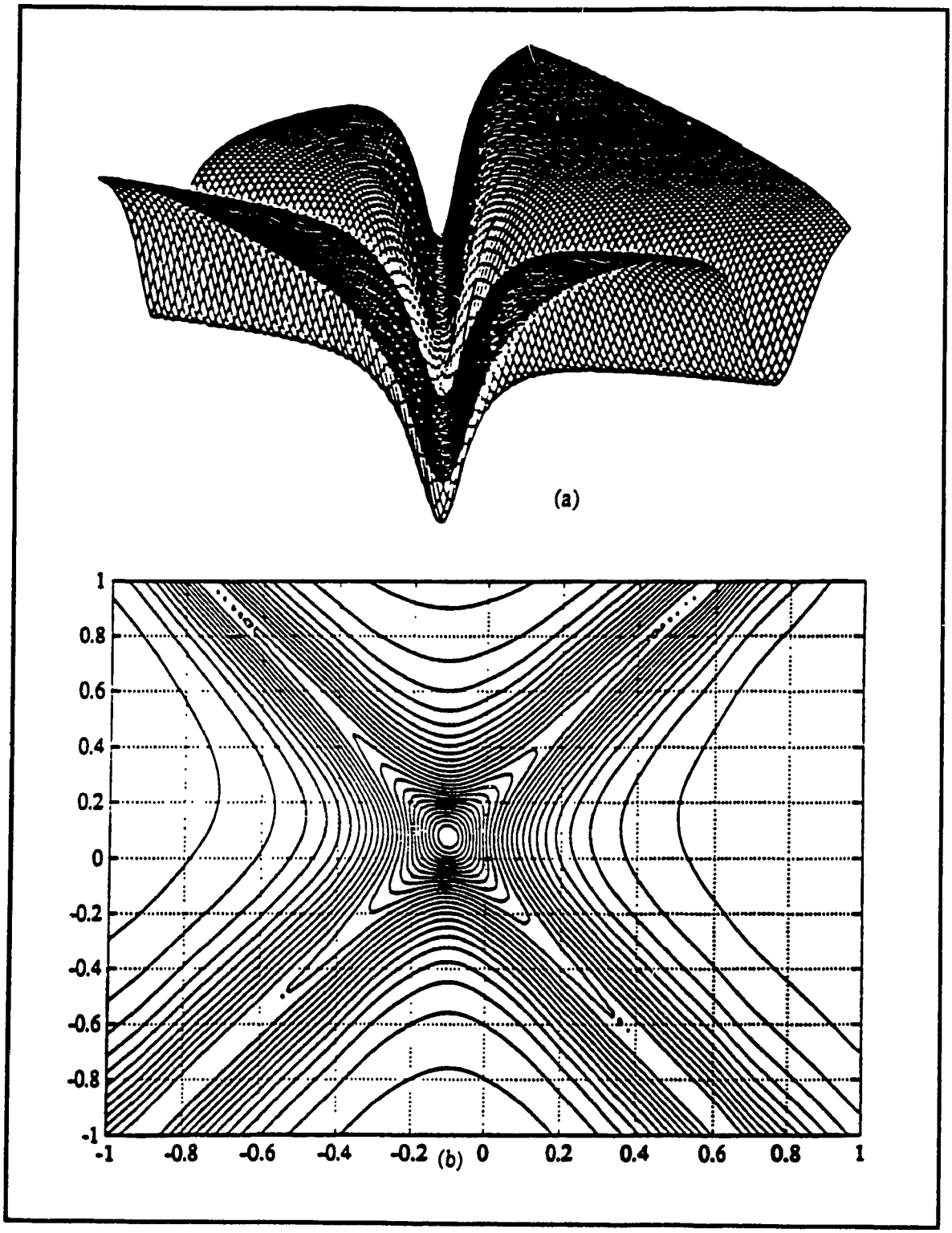

Figure 4.1. (a) Output error surface of Direct Form fourth order CANF as a function of the filter coefficients; (b) Equivalent contour plot.

Single Notch Case

Having considered the modality of the error surface associated with the double notch case (thereby, the multiple notch), we now investigate such behaviour for the single notch modules. The expression for this filter is written as

$$H(Z,\theta) = \frac{A(Z)}{B(Z)} = \frac{1+\theta_1 Z^{-1}+Z^{-2}}{1+\alpha\theta_1 Z^{-1}+\alpha^2 Z^{-2}} \tag{4.8}$$

Now, as earlier, dropping the argument Z for convenience, the derivative of the square of the magnitude can be written be as

$$\frac{\partial |H(Z,\theta)|^2}{\partial \theta_1} = \frac{2ZA}{BB^*} - \frac{Z\alpha AA^*[B+B^*Z^{-2}]}{BBB^*B^*}. \tag{4.9}$$

By replacing Eq. 4.9, in Eq. 4.1 and after some manipulations a linear equation is obtained

$$\oint B^* dZ = \oint \alpha A^* dZ. \tag{4.10}$$

This linearity of Eq. 4.10 indicates the unimodality of the error surface associated with the second order notch filter module. Unlike the multiple notch case, it is clear that in this situation no local minima will exist.

Fig. 4.2 illustrates the error function associated with the transfer function of a second order CANF evaluated for the noise free case as in Eq. 4.7. It is clear that the error function in Fig. 4.2 is unimodal, as is suggested by Eq. 4.10. In the case of second order modules it is also possible to examine the 2P parameter (see previous chapter for explanation) transfer function. For this case, Fig. 4.3 illustrates the error surface.

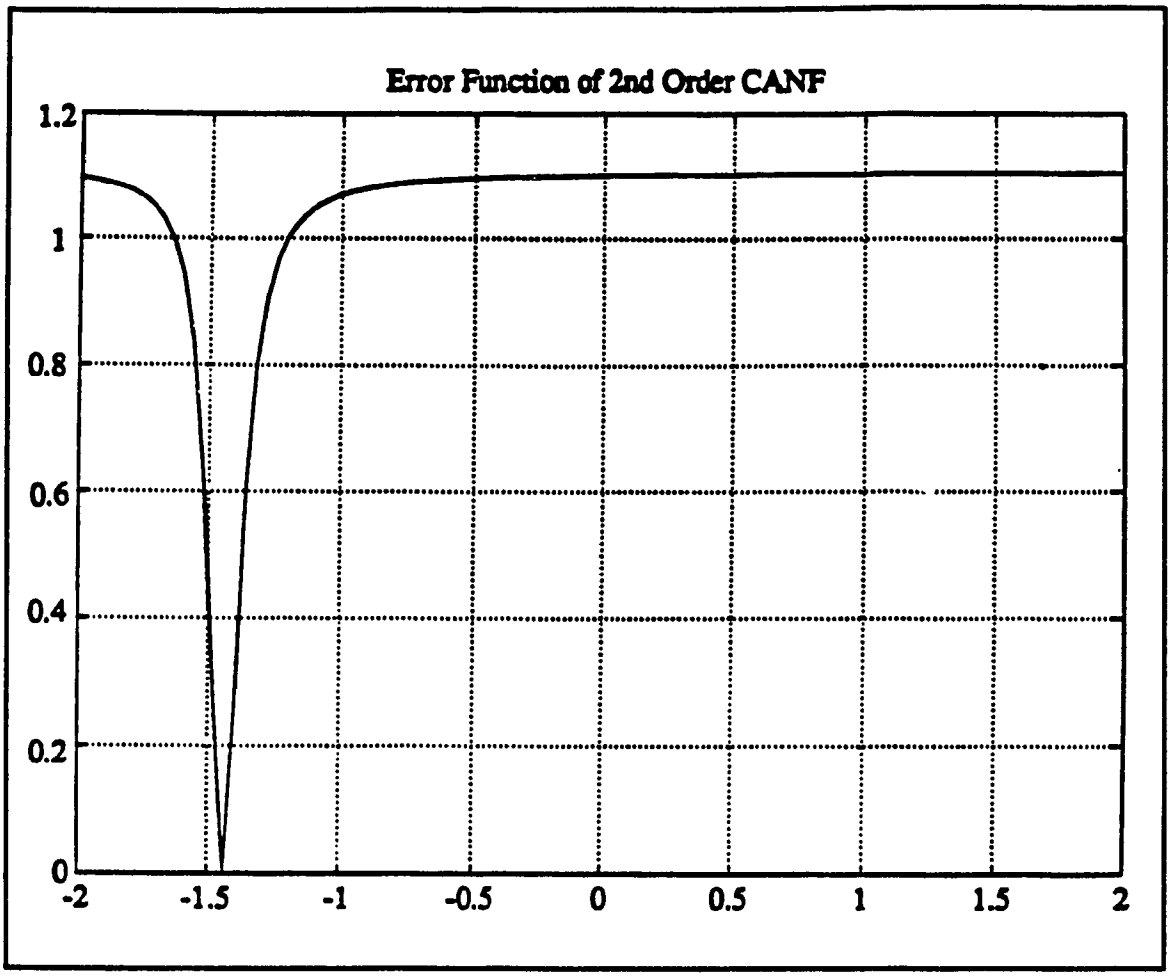

Figure 4.2. Output error surface of second order CANF with minimal parameter formulation and one normalized sinusoid the input.

4.1.2 Insufficient Filter Model

Having considered the sufficient filter model, it is observed that the error surface associated with the single notch case (i.e. second order CANF) is unimodal. This is an interesting property of the CANF which can be used advantageously in the adaptive process.

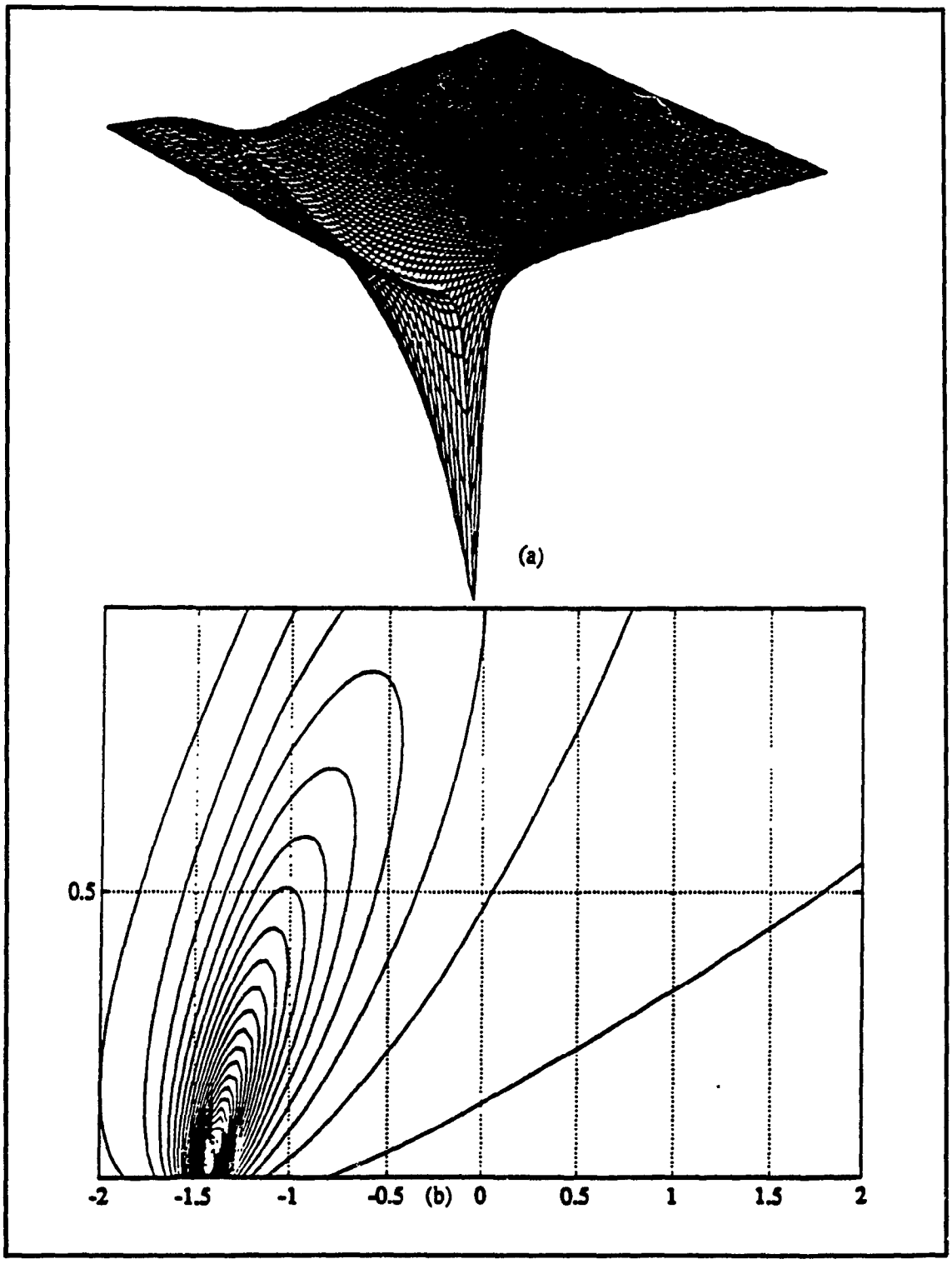

Figure 4.3. (a) Output error surface of second order CANF with 2P parameter formulation as a function of the filter coefficients; (b) Eqivalent contour plot.

Now, consider a second order CANF as an insufficient model. That is, the input mixture contains more than one sinewave, say it contains P sinewaves. The expected squared error as a function of the parameters is again given by Eq. 4.6. Considering the noiseless case of Eq. 4.7, the expected squared error is a sum of unimodal surfaces described by Eq. 4.11.

$$E\{e^2\} = \sum_{i=1}^{P} \frac{A_i^2}{2} |H(e^{j\omega_i},\theta)|^2 \qquad (4.11)$$

Since each term corresponds to an unimodal error function associated with a particular sinusoid, the sum of these terms will then yield an error function that has P minima, each corresponding to a specific sinusoid. Fig. 4.4 illustrates the error function for the P parameter second order CANF expression with two sinusoids at the input and Fig. 4.5 illustrates the error surface of the 2P parameter filter expression. The two minima associated with the two input sinusoids are clearly evident. This property of the second order CANF, having the number of minima equal to the number of sinewaves at the input, leads to an implementation that eliminates the problem of convergence to a local minima associated with higher order filters.

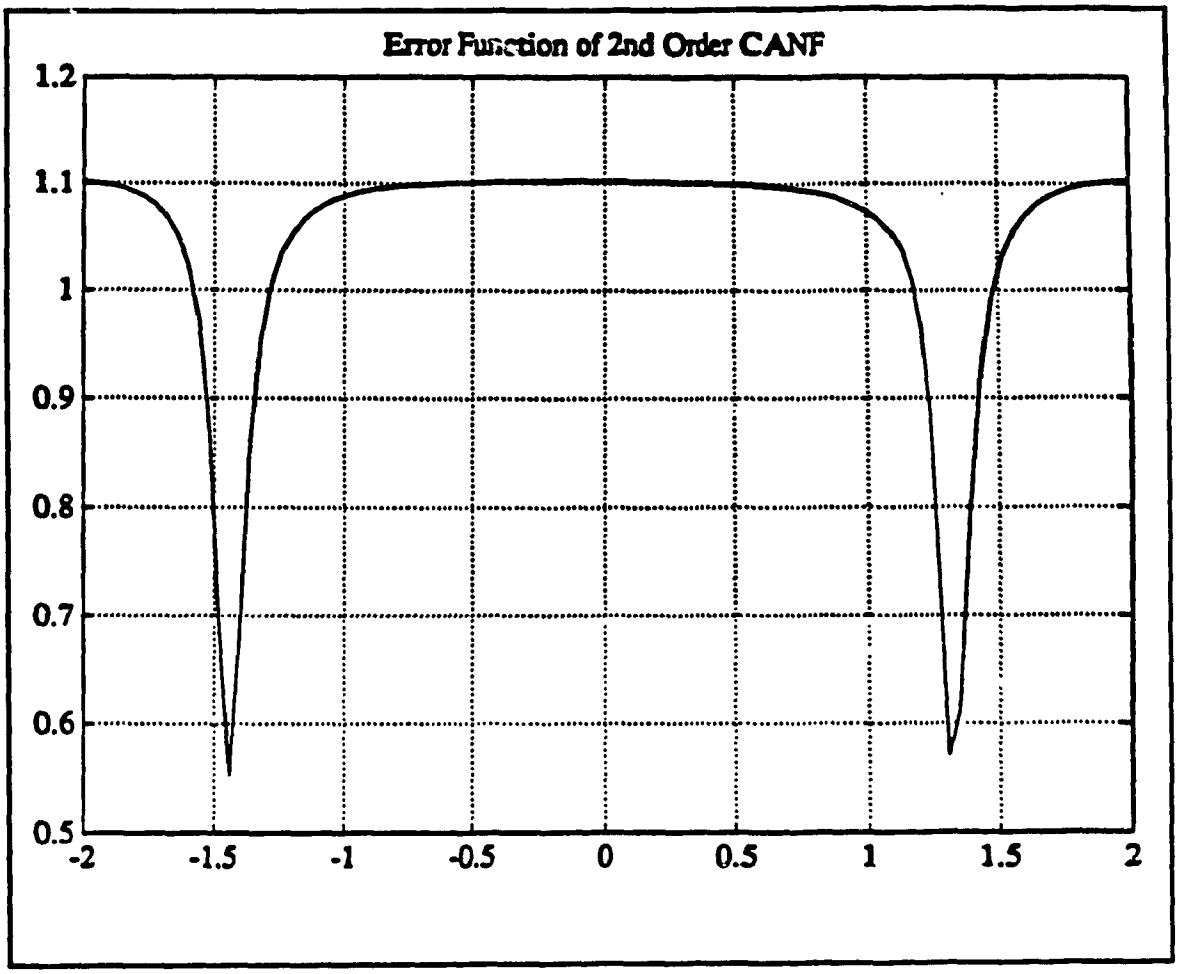

Figure 4.4. Output error function of second order CANF with minimal parameter formulation with 2 normalized sinusoids at the input.

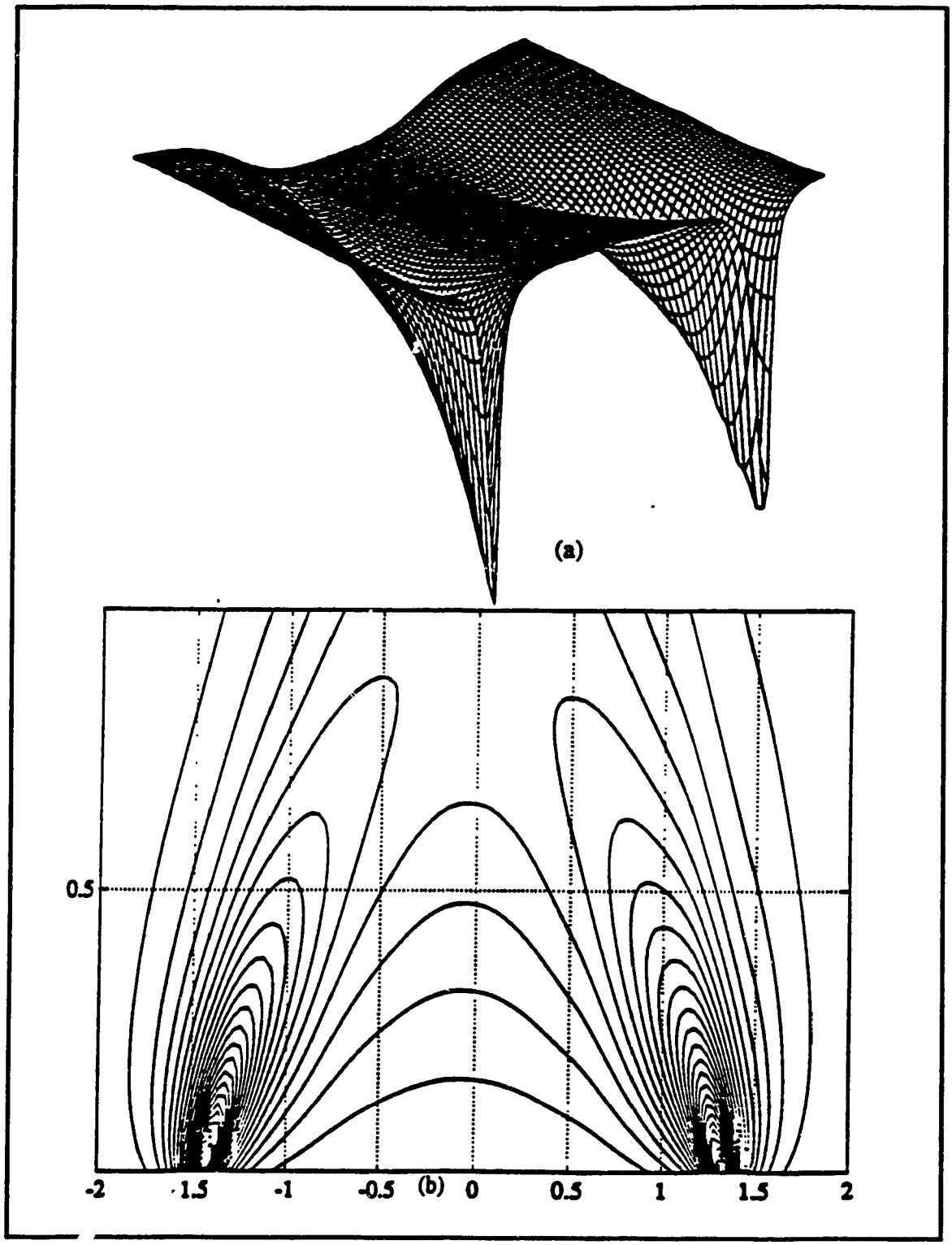

Figure 4.5 (a) Output error surface of a second order CANF with 2P parameter formulation and 2 sinewaves at the input; (b) Equivalent contour plot.

4.2 Implementation of CANFs

4.2.1 Cascade Implementation

In general, CANFs of higher order may be described in two forms, the direct and cascade, based on the following analytical expressions of the transfer functions:

$$H(Z,\alpha,\theta) = \frac{1+\theta_1 Z^{-1}+\cdots+\theta_{2P}Z^{-2P}}{1+\alpha\theta_1 Z^{-1}+\cdots+\alpha^{2P}\theta_{2P}Z^{-2P}} \qquad (4.12)$$

$$H(Z,\alpha,\theta)=\frac{(1+\theta_{1,1}Z^{-1}+\theta_{1,2}Z^{-2})\cdots(1+\theta_{P,1}Z^{-1}+\theta_{P,2}Z^{-2})}{(1+\alpha\theta_{1,1}Z^{-1}+\alpha^2\theta_{1,2}Z^{-2})\cdots(1+\alpha\theta_{P,1}Z^{-1}+\alpha^2\theta_{P,2}Z^{-2})} \qquad (4.13)$$

Eq. 4.12 represents a direct form implementation. In the last section, an example of a double notch (i.e. 4th order) CANF was used to illustrate the existence of local minima in direct form implementations. Thus, one may expect the adaptive algorithm to converge to an erroneous parameter vector (i.e local minima). This is particularly true in the case of short data, since it may require time to overcome some of the hurdles of local minima, if it is possible. To avoid such a problem, the property of second order CANF having an unimodal error surface, as described in the last section, may be used advantageously. It was said, that the error surface of a second order CANF, as an insufficient model, will yield a number of minima equal to the number of sinusoids at the input. This unique property of the CANF leads to the cascade implementation of Eq. 4.13, where the number of second order stages equal the number of sinusoids or NB signals in the input mixture. Fig. 4.6 illustrates such implementation. It should be emphasized that the parameters are estimated independently for each stage of Fig 4.6.

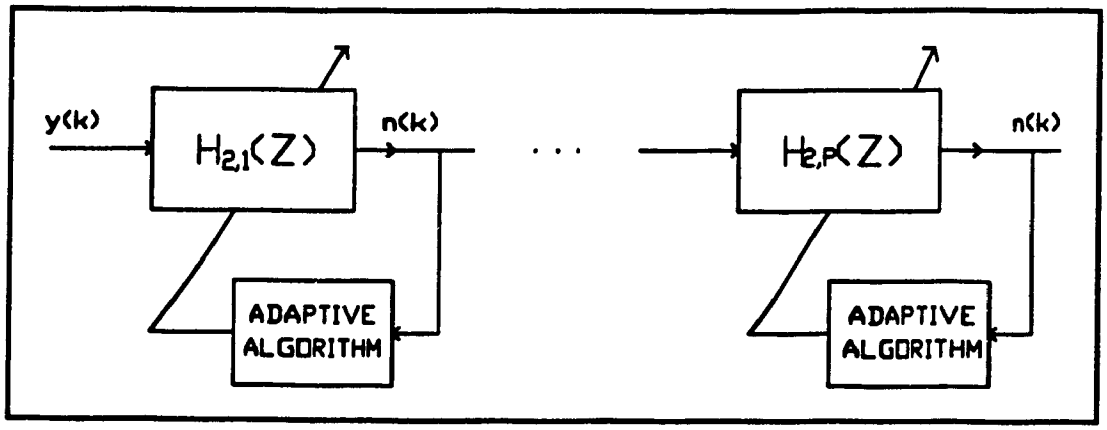

Figure 4.6. Cascade connection of P second order stages.

Consider a case in which P sinusoids are immersed in noise. Now, P second order CANF stages are implemented in conventional cascade connection. The first stage in the cascade connection will converge to one of the P minima, suppressing one sinewave. Exactly which sinewave will be suppressed first, is of no concern as the goal is to suppress all sinewaves. The output of this stage will contain P-1 remaining sinewaves. Now, the second stage will converge to one of the P-1 minima and eliminate one sinewave. Continuous propagation of the signal through the P cascade stages will eventually suppress all the input sinewaves. That is, the output of stage 1 is the input of stage 2 and so on, until at the output, only the approximate of the noise component remains. In this manner, we can guarantee convergence. Not only do we have guaranteed convergence, we need not worry about initial conditions of the parameters in the adaptive procedure, as it is not of concern which sinewave is eliminated first (each stage guarantees the suppression of one sinewave).

Another added convenience of cascade implementation is the direct usage of any available *a priori* knowledge of the frequency values of the sinusoids. This information

may be used as the starting guess of the parameters in each section. The use of this information may be considered as dividing the frequency domain in P segments, one for each of the P stages. In this manner, problems of convergence of one large filter (CANF of order 2P, where the parameter space is larger) to a unique parameter vector is reduced to P smaller filters, each having a parameter space that is minimal.

4.2.2 Cascade Implementation with *a posteriori* Filtering

The preceding ideas of notch filtering in the form of CANF, whether in the direct form or as second order modules, has been studied by various people, Rao and Kung [1], Nehorai [9], Friedlander [10] and Ng [5,11]. All these existing studies have dealt with the behaviour under the assumption of long data availability. This section will propose an algorithm, that uses the CANF as the building block, to deal with short data.

Consider P sinewaves embedded in white noise. If P second order CANFs with "very sharp notches" (α very close to one) are used to eliminate the sinewaves, then by the virtue of the algorithm, large amounts of data will be required for the convergence of each stage of cascade implementation to the desired parameters. The purpose of using narrow notches is that we wish to suppress the sinusoids, thus, only the sinewave should be eliminated and not any of the noise.

Now, consider the above situation, in which the data consists of M points, while N points ($M \leq N$) are required to form a notch point in the neighbourhood of the true notch point. Due to short data ($M \leq N$), the estimated notch point will be poor and the filter will not be able to completely cancel the sinusoid corresponding to f_1 (for the

purpose of illustration we are assuming that the first stage will indeed converge to the sinusoid corresponding to f_1, when in reality this may or may not be true). The output of stage 1 will still have some component of f_1. That is, the output of stage 1 (assumed to be free of f_1) will be corrupted by some residual power of f_1. This means that stage 2 will not have P-1 sinewaves, but will instead have P sinewaves at the input and may converge to f_1 again, deflating the argument of guaranteed convergence. These M points can be considered to be the transient period of the CANF during which no cancellation of f_1 will take place and thus, the idea of separation of sinusoids in noise using cascade implementation of second order CANFs may not be appropriate for short data.

In what follows, we propose a new configuration of the cascade connection which utilizes the information contained in the M-duration transient period and provides a reliable separation of sinewave components over the short data observation interval.

The proposed second order constrained notch filter, which serves as a building block for multistage cascade connection, is shown in Fig 4.7. The idea here is to use the CANF to estimate a notch point in the neighbourhood of the true notch point and then use these estimated parameters in an *a posteriori* constrained notch filter (CNF), a fixed filter. Motivation for this procedure is that during these M points, steady state cannot be attained and the filter output is influenced by the transient. Thus, by *a posteriori* filtering of the original data, using the parameters estimated at the Mth point, the influence of the transient (which is dependent on the initialization of the parameters θ in the adaptive procedure) is minimized. Now, provided that the notch bandwidth is large enough (controlled by the debiasing coefficient, α), an estimate near the true notch point will sufficiently suppress f_1. Henceforth, this configuration will be referred to as MC1.

Note, that the first part of the stage considered is used for the estimation of the parameter vector, θ, while the second part - *a posteriori* filtering - provides the filtering of all M points of the original signal under analysis. The lower branch, M, is used as a memory block to delay the input mixture by M points, before applying to the *a posteriori* fixed CNF, $H_1'(Z,\theta)$. The decoupling of the two functions - parameter estimation and then filtering - significantly improves (see the next section) the performance of CNF over the short data interval.

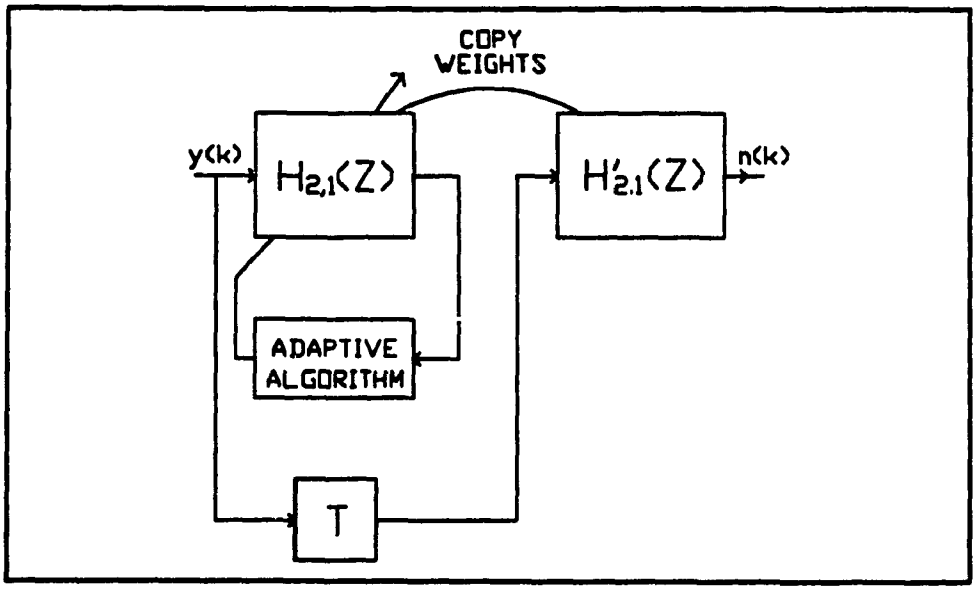

Figure 4.7. CANF with *a posteriori* fixed filtering.

4.2.3 Cascade Implementation With *a priori* Knowledge of the filter parameters

In the steady-state region (i.e., after convergence is completed) the misadjustment level is not influenced by the initial conditions. On the other hand, if M << N (i.e. convergence is not completed due to the short data), this level (thus, the performance of filtering) becomes a function of the initial conditions. In such a case, a further improvement on CANF behaviour over the short data may be achieved by using the configuration as illustrated in Fig 4.8 (basic building block for multistage connection), in which each stage is split into 2 parts. The first part of each stage is used to estimate the filter coefficients (θ) corresponding to the neighbourhood of one of the input sinusoids, which are then used as the starting guess for the second part. The second part is the MC1 implementation as described earlier. In this way, the first part in each stage is the parameter estimator that is to be used as the initial starting guess for an MC1 implementation. In the second part by repeating the process with an *a priori* knowledge of the neighbourhood of the notch point, it is possible to further minimize the output and fine tune the parameter values in the minimum mean square error, MMSE, sense. This configuration henceforth, will be referred to as MC2.

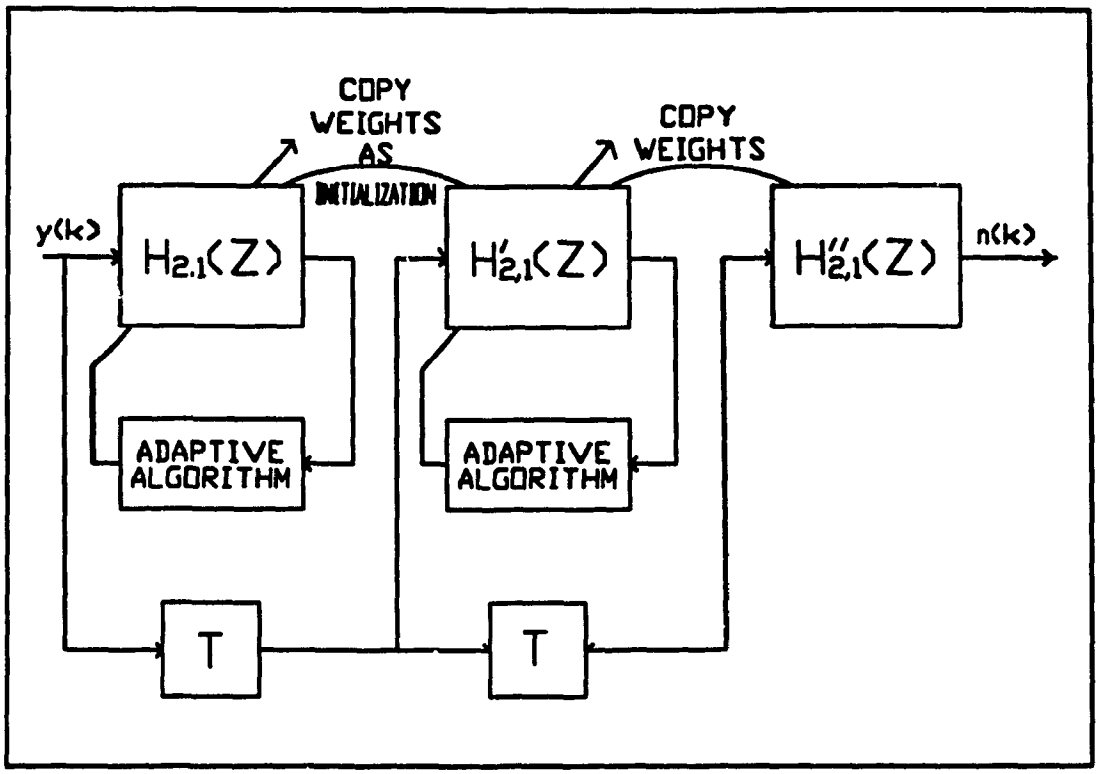

Figure 4.8. CANF with *a priori* knowledge of the coefficients.

From analysis of the error surface of the CANF, it is apparent that the cascade implementation of the second order blocks alleviates the problem associated with the convergence to a non-global minimum of higher order implementations. The number of minima associated with the second order CANF error surface is equal to the number sinusoids at the input. This fact has been advantageously used in the development of the MC1 and MC2 implementation to improve the separation of sinusoids for short data situations.

4.3 References

1. D.V.B. Rao and S.Y. Kung, " Adaptive Notch Filtering for the Retrieval of Sinusoids in Noise," IEEE Trans. Acoustics, Speech and Signal Processing, vol. ASSP-32, August 1984.

2. M. Nayeri and W.K. Jenkins, "Alternate Realizations to Adaptive IIR Filters and Properties of their Performance Surfaces," IEEE Trans. on Circuits and Systems, vol. CAS-36, April 1989.

3. B. Widrow and S.D. Stearns, Adaptive Signal Processing, Englewood Cliffs, NJ: Prentice-Hall, 1986.

4. J.J. Shynk, " Adaptive IIR Filtering," IEEE ASSP Magazine, April 1989.

5 J.F. Chicharo and T.S. Ng, "Gradient-Based Adaptive IIR Notch Filtering for Frequency Estimation," IEEE Trans. on Acoustics, Speech and Signal Processing, vol. ASSP-38, May 1990.

6. S.D. Stearns, "Error Surfaces of Recursive Adaptive Filters," IEEE Trans. on Circuits and Systems, vol. CAS-28, June 1981.

7. T. Soderstrom, L. Ljung and I Gustavsson, "A Theoretical Analysis of Recursive Identification Methods," Automatica, vol. 14, 1978.

8. J.A. Cadzow. "Recursive Digital Filter Synthesis via Gradient Based Algorithms," IEEE Trans. on Acoustics, Speech and Signal Processing, vol. ASSP-24, October 1976.

9. A. Nehorai, "A Minimal Parameter Adaptive Notch Filter with Constrained Poles and Zeros," IEEE Trans on Acoustics, Speech and Signal Processing, vol. /assp-26, Dec. 1978.

10. B. Friedlander and J.O. Smith, "Analysis and Performance Evaluation of an Adaptive Notch Filter," IEEE Trans. on Information Theory, vol. IT-30, March 1984.

11. J.F. Chicharo and T.S. Ng, "Implementation of Adaptive Notch Filtering for the Retrieval of roll Eccentricity Signal in a Steel Mill," Proc.. Int. Symp. Signal Processing Application, vol. 2, Brisbane, Australia, August 1987.

Chapter 5

Simulation Study of Time Delay Estimation and the Constrained Adaptive Notch Filter (CANF)

In chapter 2, a theoretical viewpoint of estimating the time delay between a received signal at two spatially removed sensors was presented. The underlying technique used in estimating delays was the cross correlation. Various Generalized Cross Correlators (GCCs) were presented, with emphasis on the Maximum Likelihood cross correlator. Chapters 3 and 4 presented the idea of using Constrained Adaptive Notch Filters (CANFs) to suppress/enhance sinusoids embedded in noise. In chapter 4, it was shown how the performance of CANF can be improved for short data. To handle the problem of short data, two modified cascade structures were presented.

Having described the theoretical concepts behind estimating time delays using cross correlators and the suppression/enhancement of sinusoids using CANFs, we now turn to evaluating their performance via simulations. The first section will compare the performance of the different GCCs for various signal to noise ratios (SNRs) and data lengths. The second section will attempt to compare the performance of different implementations of the CANF structures. In both of these sections, the prime objective is to show the performance for the assumption of short data availability.

5.1 Time Delay Estimation

Using computer generated data, it is possible to compare the variances achieved by the different GCCs. This simulation study is modeled after that of [1,2]. In practice,

the auto and the cross spectra have to be estimated, and as such only an Approximate Maximum Likelihood (AML) can be implemented [1]. The question then arises as to how close does the AML perform as compared to the optimal receiver (OR), which can be considered to be the true ML receiver [2]. We define the OR using *a priori* knowledge of the input signal spectrum. It is optimum in the sense that it is matched to the known source signal spectra of the input. In this section, we will compare the Smooth Coherence Transform (SCOT), the basic Cross Correlator (CC), the Approximate Maximum Likelihood (AML) and the Optimal Receiver (OR).

Two sensor models of Eq 2.1 were realized with a sampling frequency, f_s, of 2048 Hz. The sensor outputs were obtained by using three zero-mean pseudorandom Gaussian signals. Two of the three signals were used as $n_1(t)$ and $n_2(t)$. The third was passed through a 10th order butterworth lowpass digital filter with a cutoff frequency of 100 Hz to yield the input source signal, s(t). Signal, s(t), was delayed by D to yield s(t-D). The delay, D, used in all simulations was 6 samples.

The signals $n_1(t)$ and $n_2(t)$ were scaled to yield the desired signal to noise ratios. Assuming $S_{n_1n_1}(f) = S_{n_2n_2}(f) = S_{nn}(f)$, the SNR is defined as

$$SNR = \frac{\int_0^{100} S_{ss}(f)\,df}{\int_0^{100} S_{nn}(f)\,df}. \qquad (5.1)$$

where $S_{ss}(f)$ is the Power Spectral Density (PSD) of s(t) and $S_{nn}(f)$ is the PSD of the noise signal. It should be noted that the SNR of Eq 5.1 is the SNR in the signal band which is approximately 10% of the total frequency band (100/1024). As such, the total noise power at each sensor is approximately 10 times greater than indicated. For each case, 500 independent trials were conducted to obtain reliable statistics. Time delay estimates were obtained by detecting the largest peak of the GCC within the gated region of D ± 10 samples. The gate width was chosen to be the correlation time of the source signal, which is consistent with the assumption of the knowledge of the neighbourhood of true delay time.

The three GCCs - AML, SCOT and CC - were implemented as in Fig. 5.1, where the postcorrelator filter and the cross power spectrum are estimated directly from the observed data (estimates are indicated by ^). All pertinent spectra were estimated using an FFT based periodogram approach with n disjoint segments of length 64 points each. The value of n is dependent on the observation length, such that n×64 = observation length. In the case of the OR, using *a priori* knowledge of the source signal, it was selected to be

$$W_{OR} = \begin{cases} 1 & 0 \le f \le 100 \\ 0 & \textit{elsewhere} \end{cases} \qquad (5.2)$$

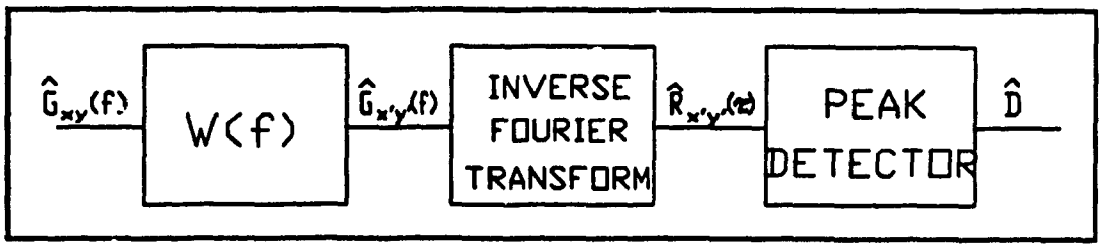

Figure 5.1. Implementation of the GCC.

Before talking about the short data situation, we will evaluate the performance of these receivers for the ideal case, that is, for a large observation interval, T. We evaluate each of the four receivers by calculating the variance of the delay estimate and comparing it against the CRLB. We expect the AML and OR to approach the bound, as the CRLB is the Maximum Likelihood estimate for large T. It is also expected that the SCOT should perform better than the CC, since it, unlike the CC, places additional weighting depending on the SNR. For the case of T = 4s, the root mean square error of the delay estimates of the four GCC receivers and the CRLB is plotted for various SNRs in Fig 5.2. The results are as expected and consistent with the published results of [1,2]. From Fig 5.2, it is apparent that all four receivers are consistent estimators of the time delay.

It was said earlier, the goal of this simulation study is to compare the performance of the above mentioned processors under the assumption of short data availability. But, before we can proceed to evaluate these processors for short data, more needs to be said about the underlying assumptions in the derivation of the ML processor. The expression for the ML processor of Eq 2.26 is obtained under the assumption that the observation time, T, is much larger than the delay, D, plus the correlation time of the source signal. The questions to ask then are the validity of this receiver for short data. Will the ML receiver maintain its efficiency for short data? and how will the other receivers perform?

To attempt to answer the above two questions, we compare the performance of the AML (AML because in practice ML can only be approximated), CC, SCOT and the OR against CRLB for short data. We have seen in Fig 5.2 that for large observation time, the performance of the AML and OR receivers approaches the CRLB. SCOT and the CC receivers are consistent estimators. For the purpose of this study of short data, we

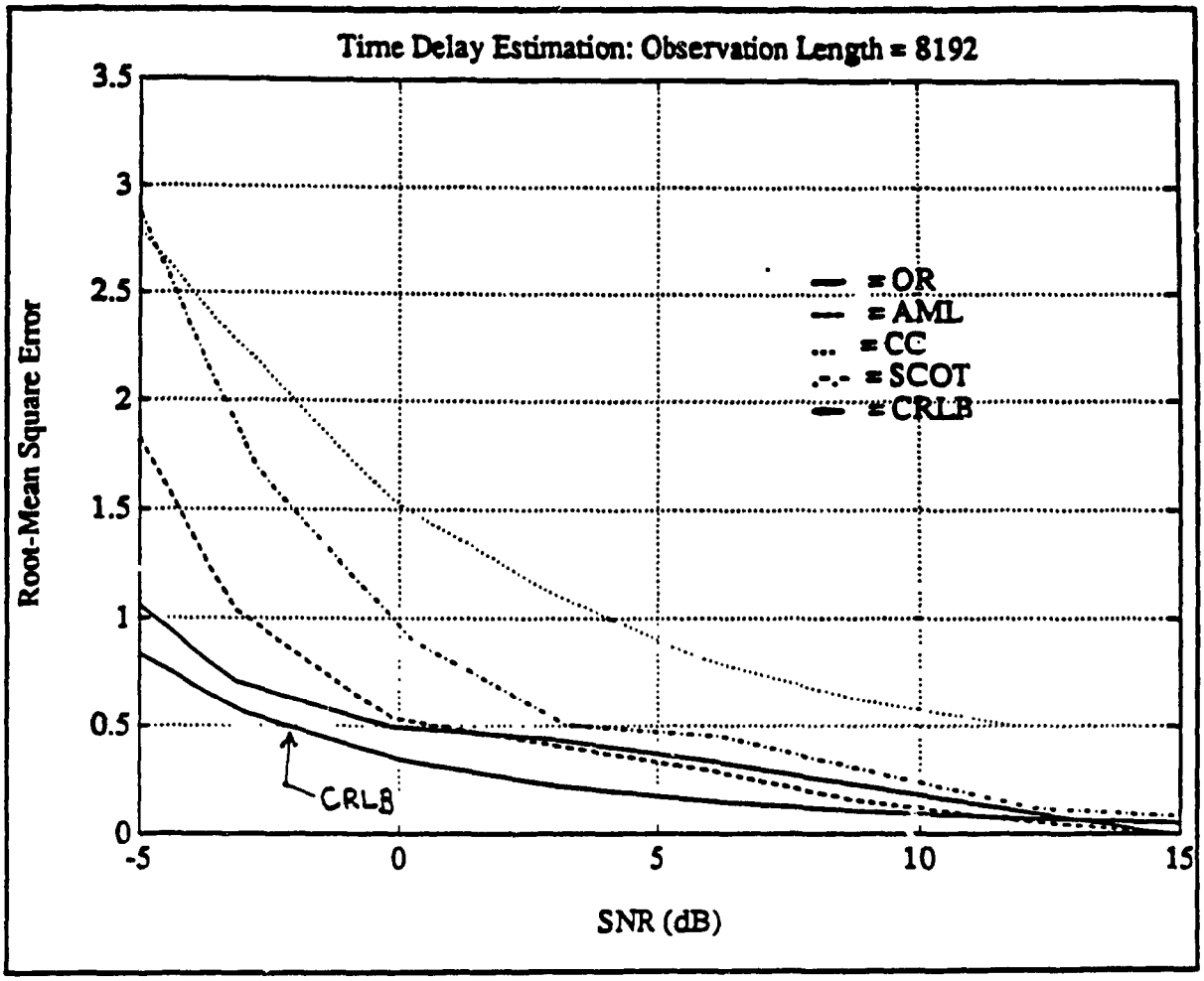

Figure 5.2. For T = 4s, Root-mean square error of delay estimates against the CRLB.

evaluate the CRLB for various short observation times and compare the root mean square error (standard deviation) achieved by the four receivers against it. In this manner, we are able to see the deviation in the tightness of the bound from the earlier case of long observation time.

The simulations for the four previously mentioned GCC processors were performed for the cases of SNR equal to approximately -6, -3, 0, 3, 6, 9, 12, 15, 18 and 21 dBs for various lengths of observation time. The results of the simulations are tabulated in Tables 5.1 through 5.4. The number in brackets represents the number of

outliers that existed for each scenario. For various SNR and observation length combinations there existed many outliers (in some cases over 50%). Since little can be said about the statistics of some of these cases; they have been eliminated from the study.

Figures 5.3 through 5.6 show the plots of the performance of the four receivers against the CRLB for the observation lengths of 1/16, 1/8, 1/4 and 1 second, respectively. From these performance plots, it can be said that all four receivers are consistent, however, they may not be valid for low SNR as in the case of T = 4s. For example, at T = 1/16 seconds, the performance of the receivers cannot be evaluated for regions of SNR lower than 5 dBs. Comparing this to the T = 4s case, it can be said that as the data length is decreased, the range of validity of the receivers must be shifted to higher SNR. In Figures 5.3 to 5.6, it is seen that as the observation length is decreased, it can be seen that the working range of the receivers decreases for lower SNR. From Figures 5.3 to 5.6, one can conclude that for short data the four receivers are consistent estimators if the range of SNR is appropriately adjusted. Fig 5.7 shows a slice of Figures 5.2 through 5.6 for SNR = 10 dbs. From this it is seen that the four receivers are indeed consistent for SNR = 10 db.

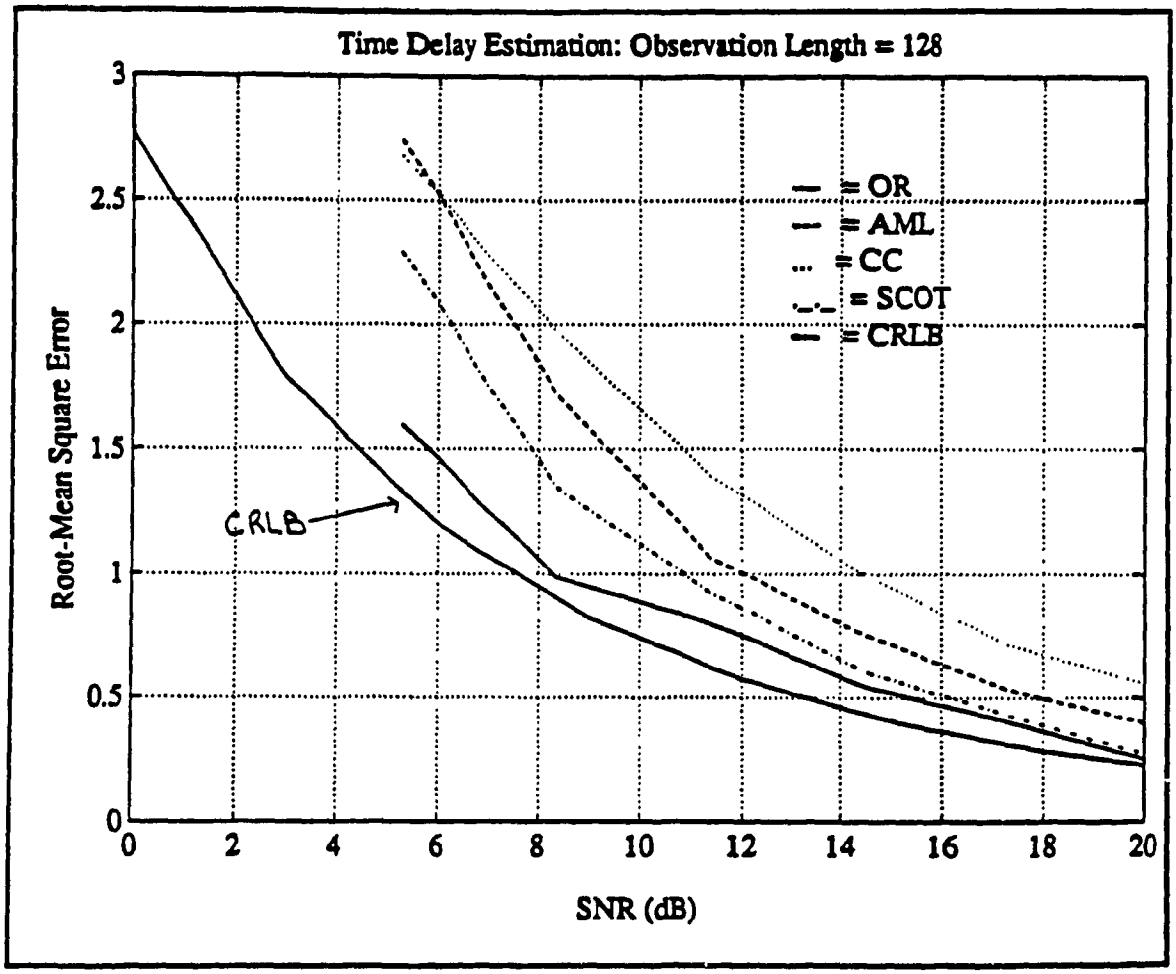

Figure 5.3. For T = 1/16, Root-mean square error of delay estimates against CRLB.

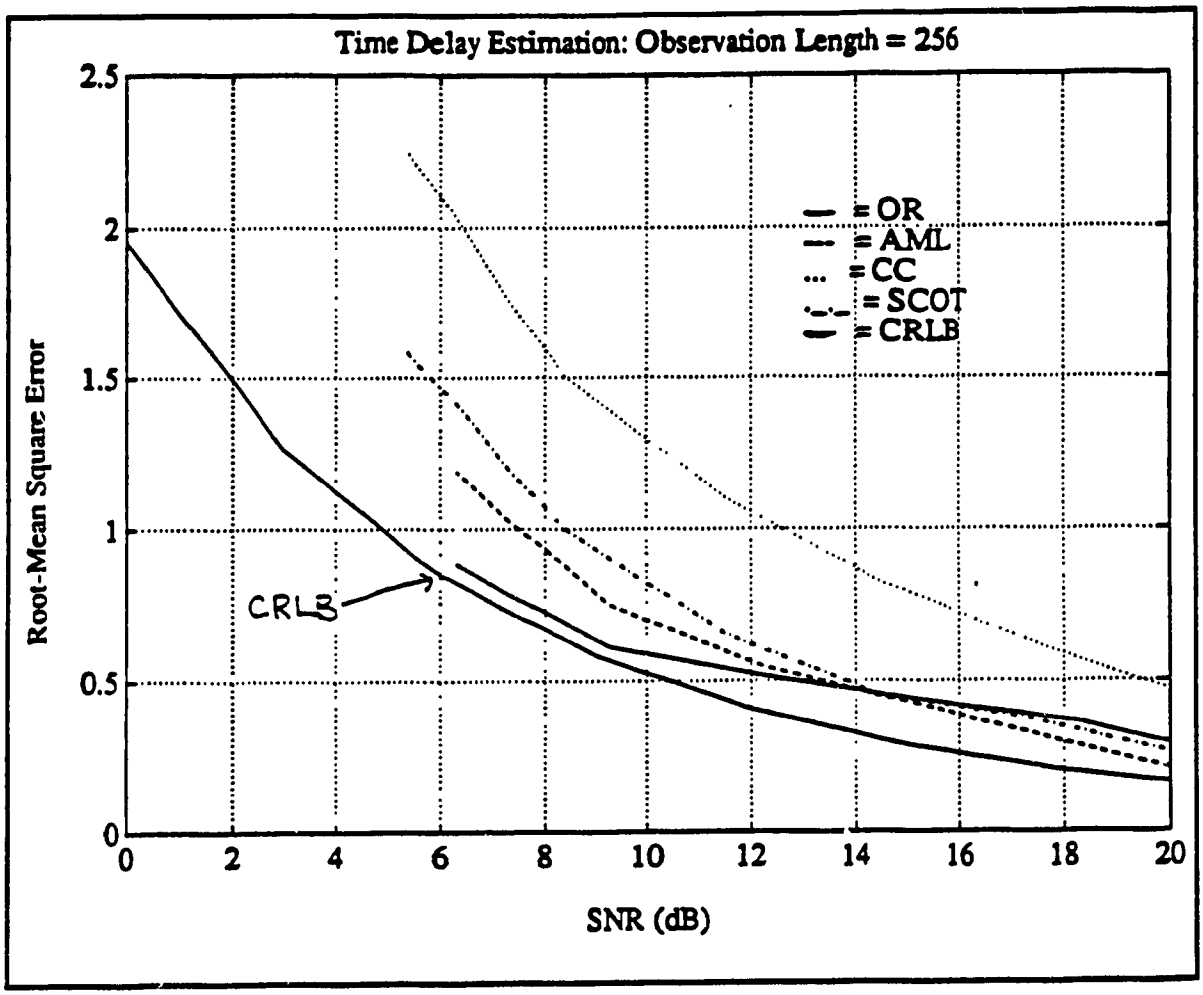

Figure 5.4. For T = 1/8, Root-mean square error of the delay estimates against CRLB.

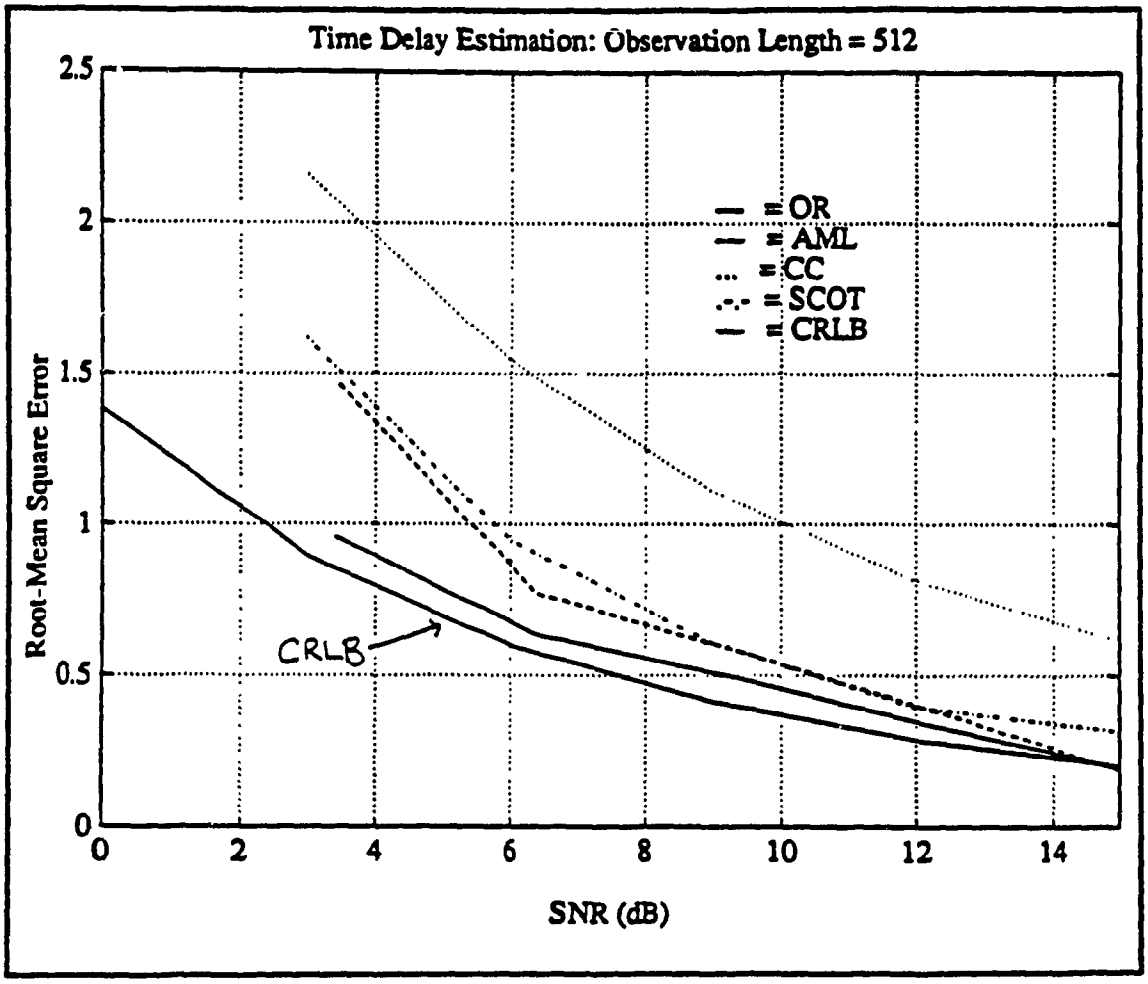

Figure 5.5. For T = 1/4, Root-mean square error of the delay estimates aagainst CRLB.

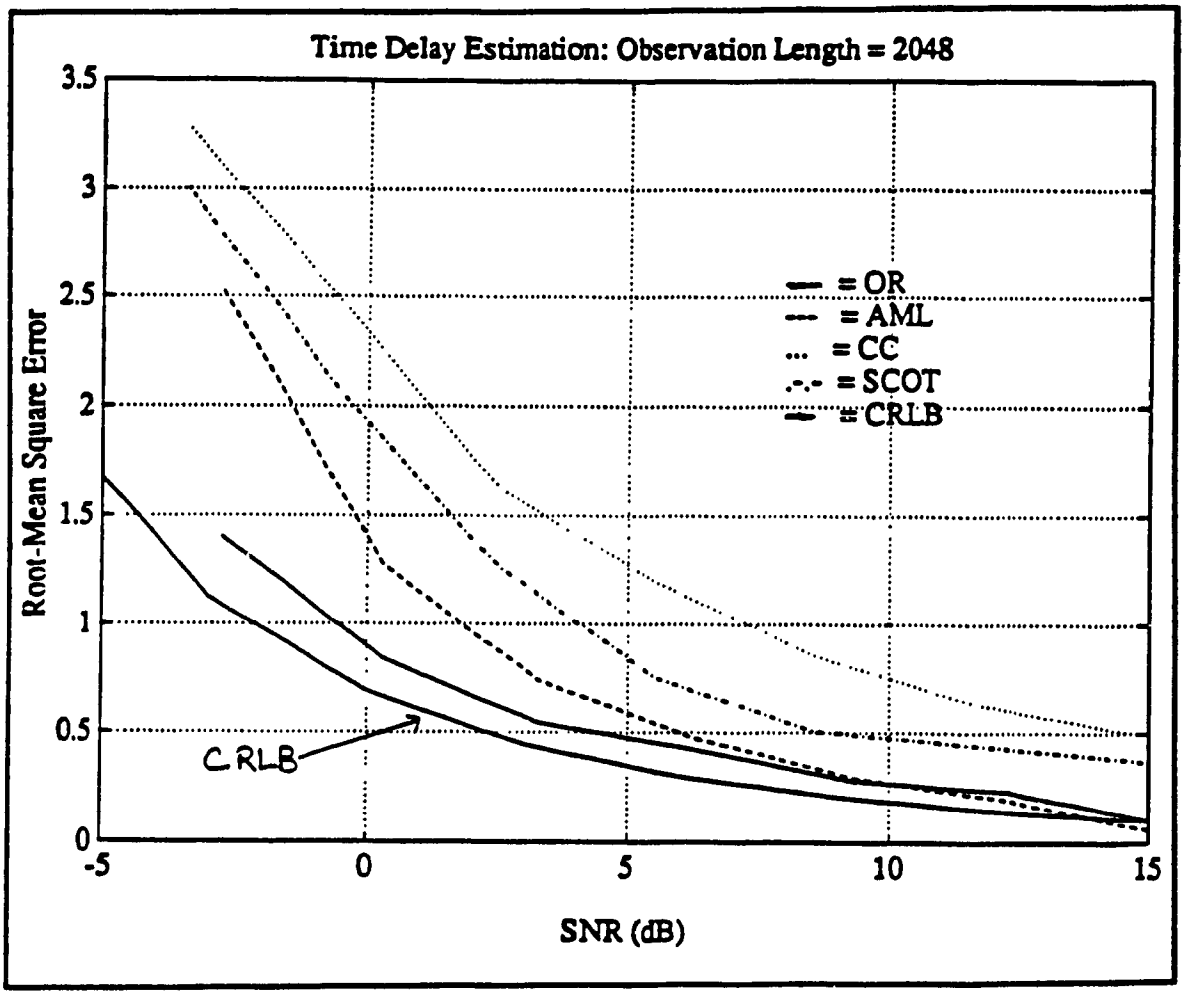

Figure 5.6. For T = 1s, Root-mean square error of the delay estimates against CRLB.

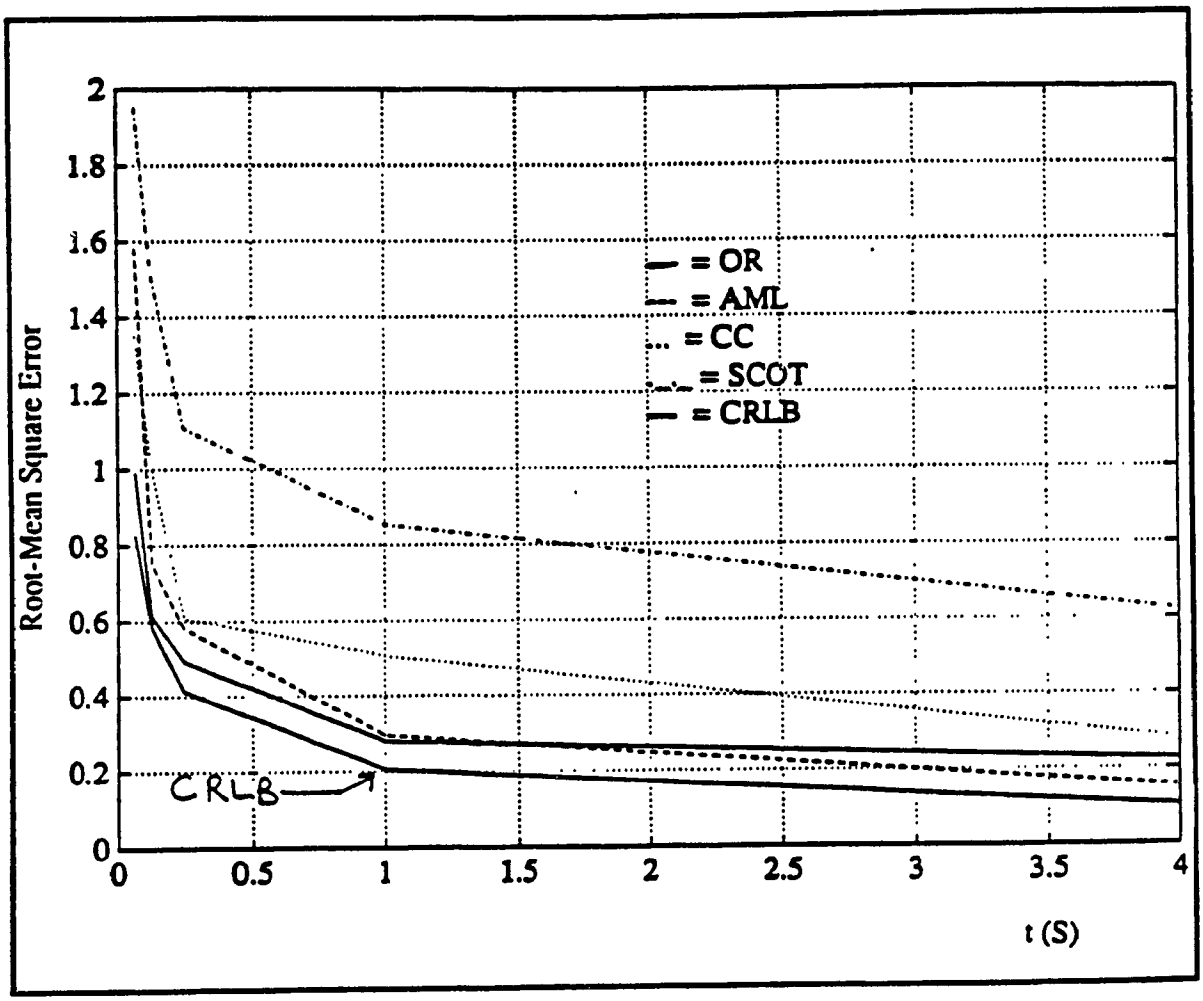

Figure 5.7. Slice of Figures 5.3 to 5.6 for SNR 9 dB showing the consistency of the estimators assymtotically.

Table 5.1 Root-Mean Square (RMS) Error of Time Delay Estimation Using the Basic Cross Correlator (CC)

N	Cross Correlator (CC)									
	SNR = -6 dBs	SNR = -3 dBs	SNR = 0 dBs	SNR = 3 dBs	SNR = 6 dBs	SNR = 9 dBs	SNR = 12 dBs	SNR = 15 dBs	SNR = 18 dBs	SNR = 21 dBs
2048	(196) 3.8839	(68) 3.2940	(3) 2.4718	1.6181	1.1972	.8537	.6362	.5033	.4669	.4238
512	(272) 4.6651	(189) 4.1416	(78) 2.9557	(4) 2.1591	1.5511	1.1094	.8154	.6128	.4538	.3030
256	(278) 5.0887	(259) 4.6592	(204) 3.8191	(97) 3.0025	(10) 2.2560	1.5086	1.1075	.8317	.6206	.4439
128	(307) 3.5540	(276) 3.5772	(226) 2.9374	(112) 2.5973	(22) 2.4269	(1) 1.9596	1.3932	.9918	.7170	.5250

Table 5.2 Root-Mean Square (RMS) Error of Time Delay Estimation Using Smooth Coherence Transform (SCOT) Processor

N	Smooth Coherence Transform (SCOT)									
	SNR = -6 dBs	SNR = -3 dBs	SNR = 0 dBs	SNR = 3 dBs	SNR = 6 dBs	SNR = 9 dBs	SNR = 12 dBs	SNR = 15 dBs	SNR = 18 dBs	SNR = 21 dBs
2048	(180) 3.7511	(37) 3.0138	2.0538	1.2573	.7522	.5064	.4444	.3795	.2620	.1090
512	(265) 4.4947	(184) 3.9593	(51) 2.5032	(1) 1.6189	.9520	.6078	.3984	.3157	.1331	.0773
256	(286) 5.0983	(250) 4.7025	(185) 3.6621	(48) 2.7589	(4) 1.5876	1.0041	.6604	.4577	.3687	.2450
128	(316) 5.0522	(279) 4.8183	(195) 4.2189	(104) 3.1417	(13) 2.2918	1.3508	.9251	.6038	.4322	.2419

Table 5.3 Root-Mean Square (RMS) of Time Delay Estimation Using the Approximate Maximum LIkelihood (AML) Processor

N	Approximate Maximum Likelihood (AML)							
	SNR = -6 dBs	SNR = -3 dBs	SNR = 0 dBs	SNR = 3 dBs	SNR = 6 dBs	SNR = 9 dBs	SNR = 12 dBs	SNR = 15 dBs
2048	(96) 3.8488	(5) 2.5248	1.2806	.7428	.4760	.2959	.1921	.0447
512	(255) 4.2713	(174) 3.7128	(61) 2.5119	(1) 1.4762	.7715	.5802	.3768	.1614
256	(315) 4.9396	(284) 4.3496	(217) 3.0751	(94) 2.1923	(25) 1.1892	.7520	.5463	.4156
128	(323) 5.0472	(278) 5.1126	(267) 4.3505	(172) 3.6601	(64) 2.7446	(12) 1.7277	(1) 1.0661	.7530

Table 5.4 Root-Mean Square (RMS) Error of Time Delay Estimation Using the Optimal Receiver Processor

N	Optimal Receiver (OR)							
	SNR = -6 dBs	SNR = -3 dBs	SNR = 0 dBs	SNR = 3 dBs	SNR = 6 dBs	SNR = 9 dBs	SNR = 12 dBs	SNR = 15 dBs
2048	(21) 2.6205	1.4040	.8437	.5469	.4242	.2806	.2302	.0892
512	(161) 3.5865	(76) 2.5076	(11) 1.4253	.9578	.6351	.4919	.3280	.1769
256	(282) 3.8874	(223) 3.2005	(121) 2.1923	(37) 1.3462	.8843	.6147	.5172	.4344
128	(289) 4.5568	(224) 4.2564	(131) 3.3839	(46) 2.2565	(3) 1.5977	.9905	.8007	.5441

5.2 Constrained Adaptive Notch Filters

Performance of the various implementations of the Constrained Adaptive Notch Filters (CANFs), described in chapter 4, is evaluated via simulations. In this section, we will attempt to verify some of the ideas developed in the previous chapters. The previously published results of [3,4,5] are all based on large data. The objective, here, is to show the improved performance of the proposed cascade structures for the case of finite observation interval (i.e. short data lengths as defined in chapter 4) as compared to the conventional cascade structure.

The performance comparisons will be based on two different basis. The first will deal with the sample statistics of the frequency estimate of the notch point, calculated from Monte Carlo simulations. The notch point frequencies are calculated by solving for the roots of the equation $A(q^{-1},\theta)$, the moving average (MA) part of the CANF. Each of the statistical results is computed from 40 independent trials. Any estimate not in the range of $\pm$ .0125 Hz of the true normalized frequency is considered to be an outlier and is not used in the calculation of the statistics. The second comparison is qualitative; it is the ability of the algorithm to separate the sinusoids.

Each algorithm is tested under various signal-to-noise ratios (SNRs) and data lengths. The SNR in the following simulation study is defined as the total signal power (i.e sinusoid power) divided by the power of the noise,

$$SNR = 10\log\left\{\sum_{i=1}^{N}\frac{A_i^2}{2\sigma_n^2}\right\}, \tag{5.3}$$

where A_i's are the amplitudes of the sinusoids. The variance of the noise is adjusted to achieve different SNRs. All simulations will use the following as the input

$$y(t) = A_1 \sin(2\pi f_1 t + \phi_1) + A_2 \sin(2\pi f_2 t + \phi_2) + n(t) \qquad t=0,1,2,\ldots,N-1 \quad (5.4)$$

where n(t) is the zero mean gaussian noise, f_1 and f_2 are the normalized frequencies in the interval [0,.5], ϕ_1 and ϕ_2 are randomly chosen phase in the interval $[-\pi, \pi]$ and the input data is sampled at a rate of 1 Hz.

Upon examination of Eq 5.4, it is evident that there exist many possible combinations of the amplitudes, A_i's, and frequencies, f_i's. Thus, for the purpose of the simulation study, we will restrict to a few cases. That is, the performance of the direct form (DF), conventional cascade (CC), modified cascade one (MC1), and modified cascade two (MC2) will be studied for the cases: 1) $A_1 = A_2$, $f_1 = .0625$, $f_2 = .1875$; 2) $A_1 \neq A_2$, $f_1 = .0625$, $f_2 = .1875$; and 3) $A_1 = A_2$, Δf ($= |f_1 - f_2|$) varying in the interval [0,.5] for various values of α, the debiasing coefficient.

CASE 1: $A_1 = 1$, $A_2 = 1$, $f_1 = .0625$, $f_2 = .1875$

EXAMPLE 1: In this example, it is desired to compare the performance of the cascade connection of two stages of second order filters (Fig 5.8) with the direct form implementation of a fourth order filter. Both of these configurations were implemented using the 2P parameter algorithm as described in the previous chapter. For all runs, the debiasing coefficient α, was chosen to be .9 and the initial value of θ (the filter coefficients) is set to zero.

First, we will compare the sample statistics of the notch point, computed from the Monte Carlo simulations. And secondly, we will compare the ability of the two implementation of the CANFs to suppress the sinusoids.

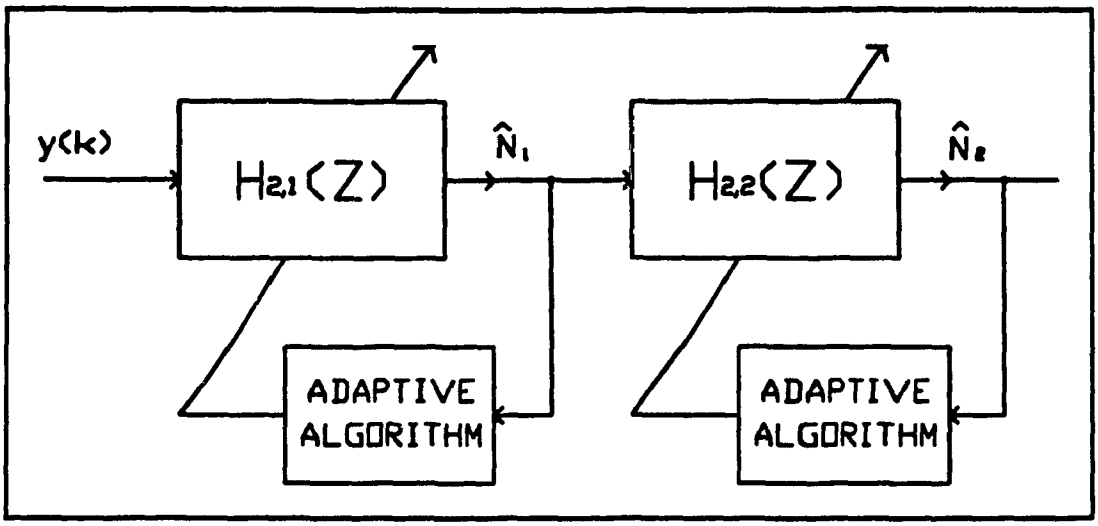

Figure 5.8. Implementation two stages of CC.

Table 5.5 and 5.6 show the bias and variance of DF and the CC of the CANF, respectively. The number in brackets represents the number of trials that were not included in the statistical calculations because the estimate was not in the arbitrarily chosen neighbourhood of the true frequency. See note below.

Note: In the case of the cascade connection, the number of outliers also include the case where both stages converged to the same notch point. This indicates that the first stage, at times, is not capable of completely suppressing one sinusoid and as a result a portion of this sinusoid propagates to the output of the first stage. Sometimes the power of this remaining component of the sinusoid is comparable to the second sinusoid, especially at the start of adaptation of the second stage. This can cause the second stage, in the cascade connection, to converge to the same frequency as the first stage. In such situations, one of the two frequency estimates has been rejected in the statistical calculations.

There are several interesting things to observe in these two Tables. First, it is evident that for every case the lower frequency (f_1) yields a lower bias and a lower variance. This is due to the fact that these constrained IIR structures are computationally efficient at frequencies much lower than the sampling frequency. For this reason the lower frequency does indeed yield better results [6].

Second, with sufficiently large data one would expect both configurations to yield similar results, since the 2P parameter algorithm is an asymptotically unbiased estimator [5]. As N approaches 2048, total error of the notch points in the two configurations approach equality. Now, as we approach N = 128, for all SNR's in the Tables, the cascade connection is the superior one. It has lower biased estimates of the notch points, as well as lower variances. This may be due to the fact, that, in the direct form implementation the recursive algorithm searches for the global minima. Thus, in order to minimize the error criterion or to approach the global minima, the algorithm must simultaneously adapt four parameters, thereby increasing the number of degrees of freedom. But, in the cascade connection, the algorithm searches for one of the two local minima corresponding to the two sinusoids at the input. As stated in chapter 4, there is no global minima [7]. In this case the number of degrees of freedom is only two for each section and thus yielding better results. This is further exemplified by the greater number of outliers in the DF, especially at N = 128. In this case, there exist local minima as well as the global minima [7], it is then reasonable to say that, the adaptive algorithm requires more time to overcome the hurdles of the local minima in order to converge to a global minimum, if it is possible.

We will now compare the two algorithms to check their ability to suppress the sinusoidal component of the input data. That is, at the output, only the noise component of the input signal should remain. Since the performance of the two configurations is similar for large N, for this comparison we will use a small value of N and illustrate the difference between the two configurations. Since we are talking about the ability to suppress a sinusoid, the following evaluation or comparison is done in the frequency domain.

Fig 5.9 shows the result of the direct form fourth order filtering with N = 128 at SNR = 4 dB. The fourth order filter will have two notch points as seen in Fig 5.9c. The notch points in this example are at frequencies in the neighbourhood of the correct frequencies. However, due to their lack of depth and sharpness, the suppression of the sinusoid is negligible as seen in Fig 5.9b. Now, as we increase the number of points (N) to 256, the notches become sharper and closer to the true frequency values (see Fig 5.10). In this case, the notches are sharper with a better estimate of the notch point and still there is only slight improvement in the cancellation of the sinusoids. The following may be used for the explanation of this phenomenon. In the Short Data Analysis chapter (chapter 4) it was pointed out that in certain situations the transient time for the CANF to converge can be large. During this transient time, the notches at the correct frequency do not exist and hence no cancellation will take place, leading to an output corrupted by the sinusoids that should have been suppressed. It should be noted that the filter transfer functions of Fig 5.9 and Fig 5.10 are valid only at the end of the adaptation process, at time N. During the adaptation period, the transfer function cannot be considered.

The outputs of each of the two stages, labelled N_1 and N_2 in the CC configuration of Fig 5.8, are of interest for the purpose of sinusoidal cancellation. Fig 5.11 shows the results of the simulations for the case of N= 128 and the SNR = 4 dBs. The figure also includes the transfer functions for each of the two stages of the cascade connection, H_1 and H_2 (subscript indicates the respective stages).

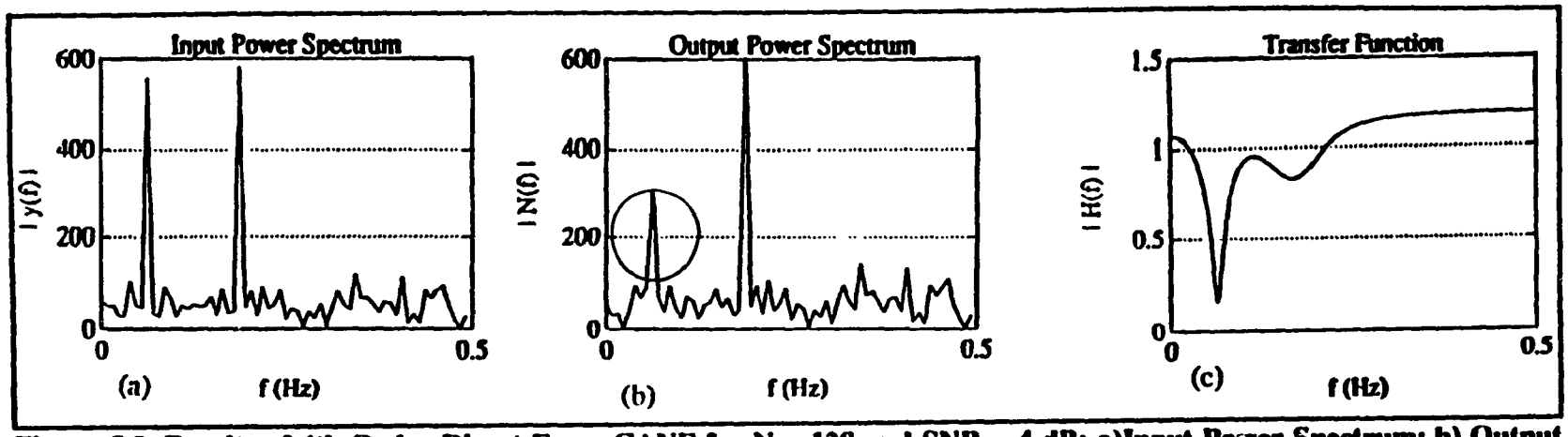

Figure 5.9. Results of 4th Order Direct Form CANF for N = 128 and SNR = 4 dB: a)Input Power Spectrum; b) Output Power Spectrum; c) Transfer Function.

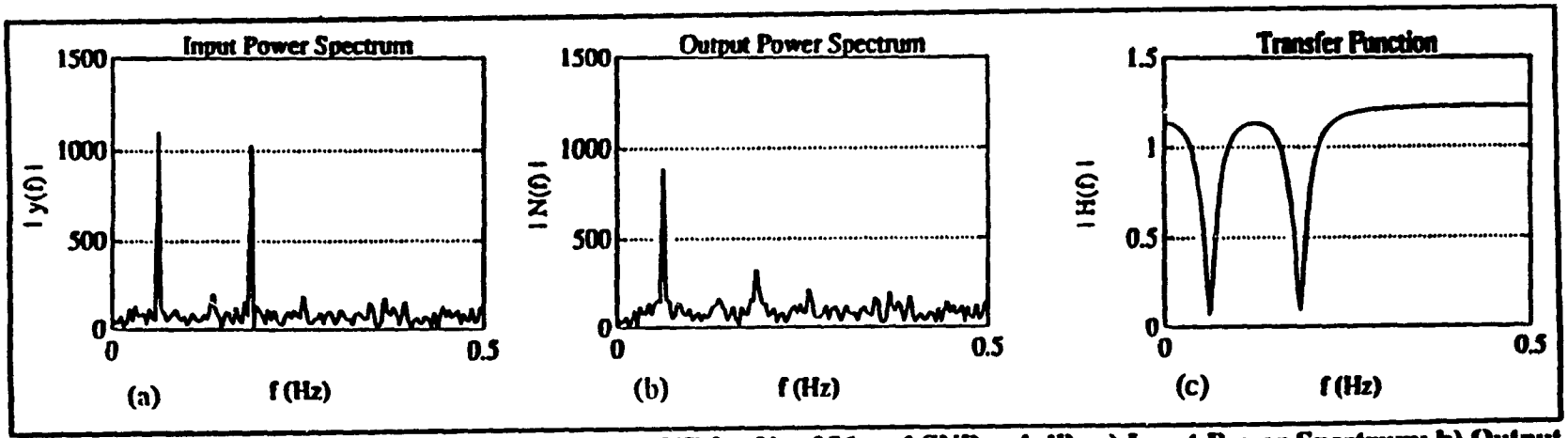

Figure 5.10. Results of 4th Order Direct Form CANF for N = 256 and SNR = 4 dB: a) Input Power Spectrum; b) Output Power Spectrum; c) Transfer Function.

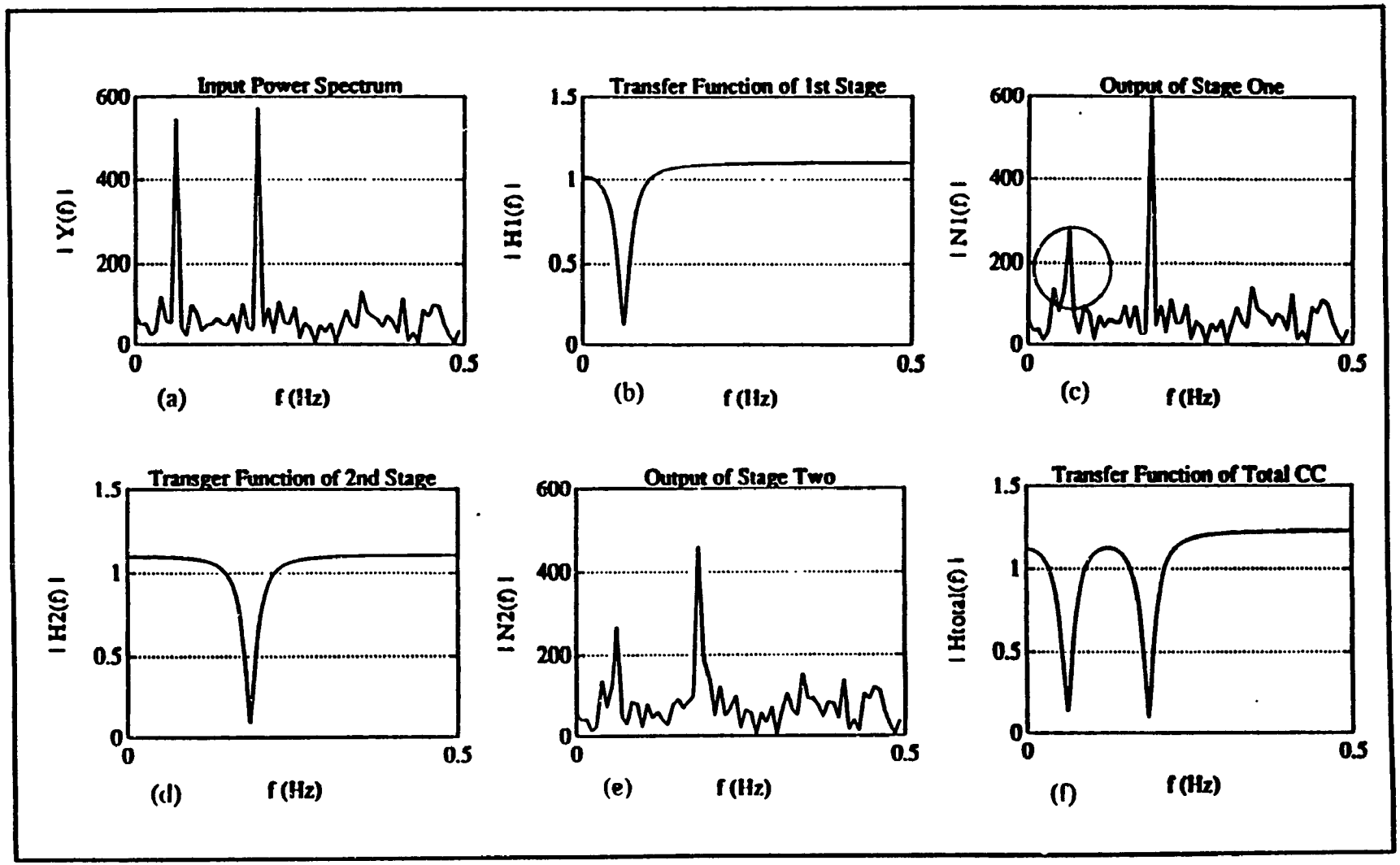

Figure 5.11. Results of Two Stages of 2nd Order CC CANF for N=128 and SNR=4 dB.

Looking at the transfer functions of the first stage, H_1 of Fig 5.11b, it is seen that the notch point is very close to f_1, and yet there is not complete cancellation in Fig 5.11c. There may be two reasons for this behaviour: (1) as in the direct form implementation, the transient may be long, and thus the residual component is due to the transient and (2) the transfer function is not zero at the notch point. Since the first stage was not capable of removing the first sinusoid completely, there exists a small component of this sinusoid at the input of the second stage, the circled part in Fig 5.11c. In the adaptation process for the second stage, the transient time will be longer because the algorithm will first try to track the same sinusoid as the first section and then try to track the correct frequency, that of the second sinusoid. In the adaptation of the second stage, tl. . starting condition are exactly the same, and thus residual component of the first sinusoid is the strongest at the start of the adaptation of this stage. That is, during the transient time of the first stage, no cancellation takes place. In this manner, some time may be wasted. For short data, it is possible that the second section may not even converge to this second frequency, this of course depends on the amount of interference by the residual sinusoid of the first stage.

It can be seen from Table 5.6, the higher frequency sinusoid has a higher bias of the notch point. Therefore, by not being as close to the actual frequency as the first stage, the sinusoid suppression is further inhibited. As we increase the number of points tc N = 256, the notches (the transfers functions in Fig 5.12b,d) become sharper and deeper resulting in cancellation to be more effective (see Fig 5.12c,e). It is then obvious, that, for short data we do not expect signal suppression from the second section to be as effective as the first section.

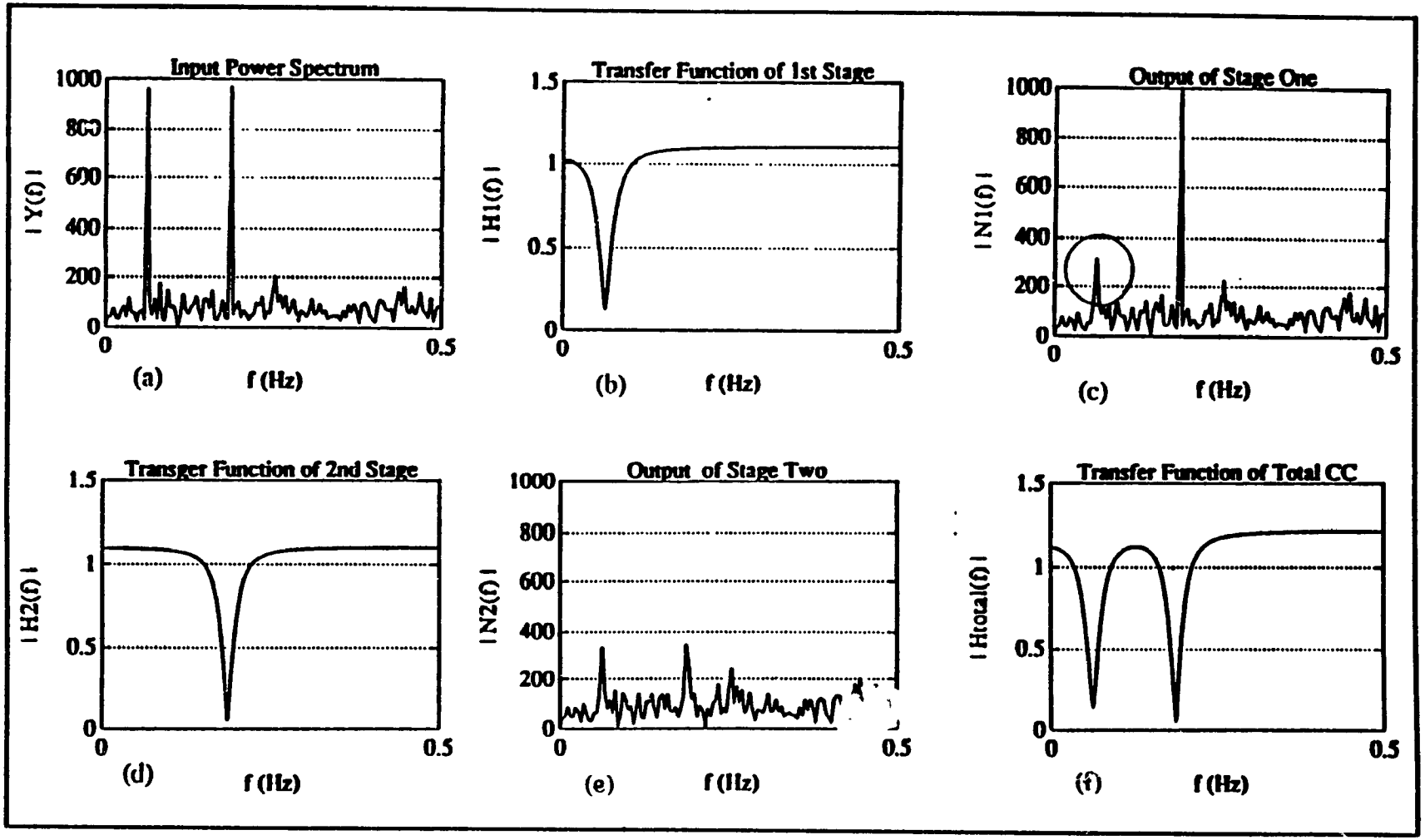

Figure 5.12. Results of Two Stages of 2nd Order CC CANF for N = 256 and SNR = 4 dB.

Comparing the results of sinusoid suppression , it is evident that for short data and equal starting conditions, the cascade connection is the superior one. It estimates the notch points with a lower error and creates sharper notches resulting in the suppression of the sinusoids to a greater extent. The various reasons for such behaviour are apparent from the above discussion. The greater degrees of freedom (estimating 2P parameters simultaneously as opposed to 2 parameters at a time) in the direct form implementation of the CANF results in inferior performance, as is shown by this simulation. We may conclude that if one is able to minimize the effect of the transient, then the above results may be further improved. This is discussed in the next example.

Table 5.5 Statistical Results of Frequency Estimates Using Direct Form (DF) Implementation

N	SNR = 10 dB				SNR = 04 dB				SNR = 0 dB			
	BIAS		VAR		BIAS		VAR		BIAS		VAR	
	f_1	f_2	f_1	f_2	f_1	f_2	f_1	f_2	f_1	f_2	f_1	f_2
	x 10^{-4}	x 10^{-4}	x 10^{-8}	x 10^{-8}	x 10^{-4}	x 10^{-4}	x 10^{-8}	x 10^{-8}	x 10^{-4}	x 10^{-4}	x 10^{-8}	x 10^{-8}
2048	.8089	7.0673	.8538	6.7168	1.0658	3.5784	1.4557	6.6456	1.0841	-2.3475	2.6203	8.8347
1024	.9593	11	1.2738	15.881	1.7101	9.2700	5.1993	19.181	2.4807	6.4572	13.295	222.22
512	1.7190	18	7.4999	30.559	(1) 2.8622	(1) 16	13.603	34.876	4.8887	17	67.474	318.02
256	(2) 4.5178	(14) 35	225.52	328.29	(2) 4.5470	(2) 33	496.55	189.19	(2) 11	(3) 32	797.19	596.82
128	(35) 16	(18) 29	395.83	201.65	(28) 25	(20) 41	867.51	2351	(13) 22	(18) 63	4174	185[a]

Table 5.6 Statistical Results of Frequency Estimates Using Conventional Cascade (CC) Implementation

N	SNR = 10 dB				SNR = 04 dB				SNR = 0 dB			
	BIAS		VAR		BIAS		VAR		BIAS		VAR	
	f_1	f_2	f_1	f_2	f_1	f_2	f_1	f_2	f_1	f_2	f_1	f_2
	x 10^{-4}	x 10^{-4}	x 10^{-8}	x 10^{-8}	x 10^{-4}	x 10^{-4}	x 10^{-8}	x 10^{-8}	x 10^{-4}	x 10^{-4}	x 10^{-8}	x 10^{-8}
2048	(1) -1.2386	(1) 7.4734	.0412	14.587	-.9205	(1) 2.8648	1.1193	13.236	(4) -1.5339	(6) -8.8550	2.1104	733.75
1024	(1) -.8733	(2) 8.3957	.2144	10.873	(2) -.3144	(2) 6.3386	2.8989	41.030	(4) .4579	(7) 3.5274	3.3959	1274.4
512	(1) -.5460	(4) 12	.9502	44.668	(1) .1745	(4) 12	3.0613	94.939	(2) 1.1081	(10) 12	17.463	821.05
256	2.6283	(6) 16	7.3342	66.053	(1) 2.4243	(8) 19	65.076	169.53	(10) 10	(8) 21	454.58	454.71
128	(2) 7.7763	(16) 32	83.508	288.92	(8) 14	(13) 33	497.43	392.31	(7) 12	(18) 37	1951.1	2115

EXAMPLE 2: In this experiment, we will compare the performance of the MC1 of Fig 5.13, against the performance of the CC of Example 1. MC1 is tested under identical conditions as the CC of the last example (i.e. same input, SNR, α, and so on). As in the CC, each stage of Fig 5.13 is adapted independently. The output of the first stage is the input to the second stage. However, in the MC1 each stage is split into two sections. The first section is used to estimate the filter coefficients using a conventional 2nd order CANF, while the second section is an *a posteriori* fixed Constrained Notch Filter (CNF), that filters the original input used to estimate its coefficients. In Fig 5.13, these *a posteriori* IIR filters are denoted by a prime. The coefficients for these fixed CNFs are obtained by copying the weights from the adaptive section after the adaptation is complete. The time delay, T, is chosen to be the amount of time required for adaptation to be completed.

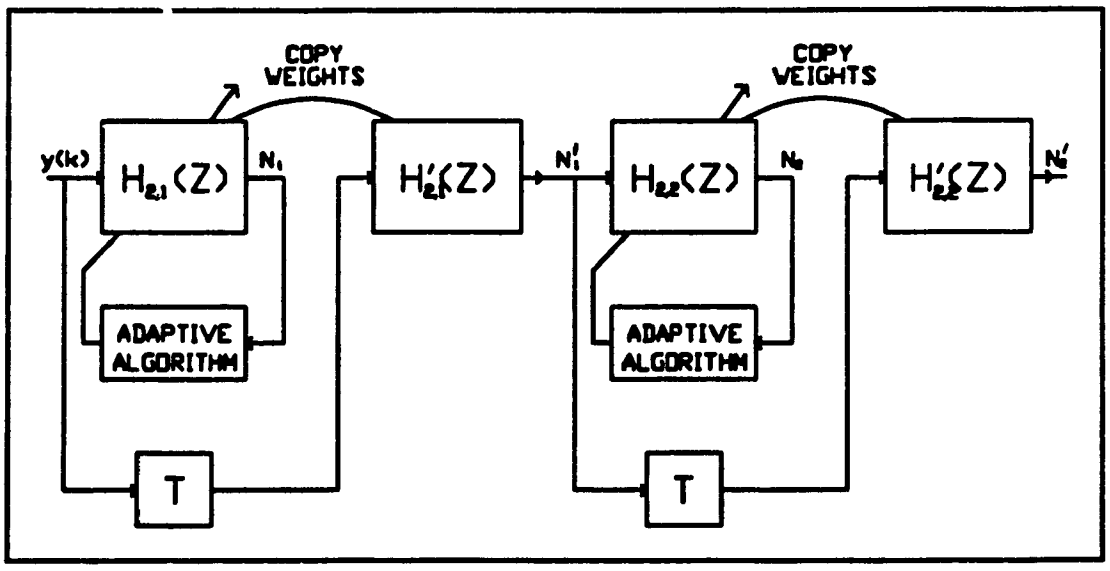

Figure 5.13. Implementations of two stage of MC1.

In the previous example, it was observed that the CC showed a significant improvement over the DF. Better statistics of the notch point frequency were obtained (see Table 5.5 and 5.6) and there was a slight improvement in the suppression of the sinusoids for the short data case. However, both of these algorithms displayed a large number of outliers, particularly for short data and low SNR. The MC1 implementation of this example results in a minimal number of outliers, while maintaining or enhancing the performance as compared to the CC.

The notch point statistics of the MC1 are tabulated in Table 5.7. Comparing these statistics with that of the CC (Table 5.6), we notice that the number of outliers have been reduced considerably. Even in this worst case, there is a significant reduction in the number of outliers.

The reasons for the large number of outliers in the CC were discussed in the first example. It was reasoned that if the effect of the transient can be minimized, then the subsequent stage would provide faster and better estimate of the next notch point. That is, if the input to the second stage contains only one sinusoid (the residual component or the transient of the 1st stage has been minimized by *a posteriori* filtering) then it is forced to converge to the correct frequency. There is no possibility of converging to another frequency value as the input to the second stage contains only one sinusoid, hence minimizing number of outliers.

In this modified structure, one may notice from the Tables 5.6 and 5.7, that there is a slight improvement in the variance of the notch point of the second frequency, f_2. This, of course is due to the minimized interference by the residual component of the sinusoid that has been eliminated by *a posteriori* filtering in the first stage.

The minimization of the transient effect can be further clarified by looking at the ability of this modified structure to suppress sinusoids. Fig 5.14 contains the results of a simulation example of this MC1 implementation with SNR = 4 dB and N = 128. Fig 5.14c shows the output of the adaptive section of the first stage. It can be seen, that the power of the sinusoid that is supposed to have been eliminated is quite large, the circled part. By *a posteriori*, filtering, we eliminate or minimize the effect of the transient. Fig 5.14d shows the results of this process, the transient is virtually removed. Now, if we use this *a posteriori* filtered sequence as the input for the second stage, we may expect it to converge to the correct frequency, as can be seen by the results of Fig 5.14f,g, the outputs of the adaptive and the *a posteriori* fixed filter, respectively. In Fig 5.14g, the two sinusoids have been almost completely removed. This is a considerable improvement over the CC (the final output in Fig 5.11e) and a drastic improvement over the DF, Fig 5.9b. This simulation example is again implemented with N = 256, and comparing the overall output of the three types of filters (Fig 5.10b, 5.12e and 5.15g), it is evident that the MC1 configuration displays the suppression of the sinusoids to be the most effective.

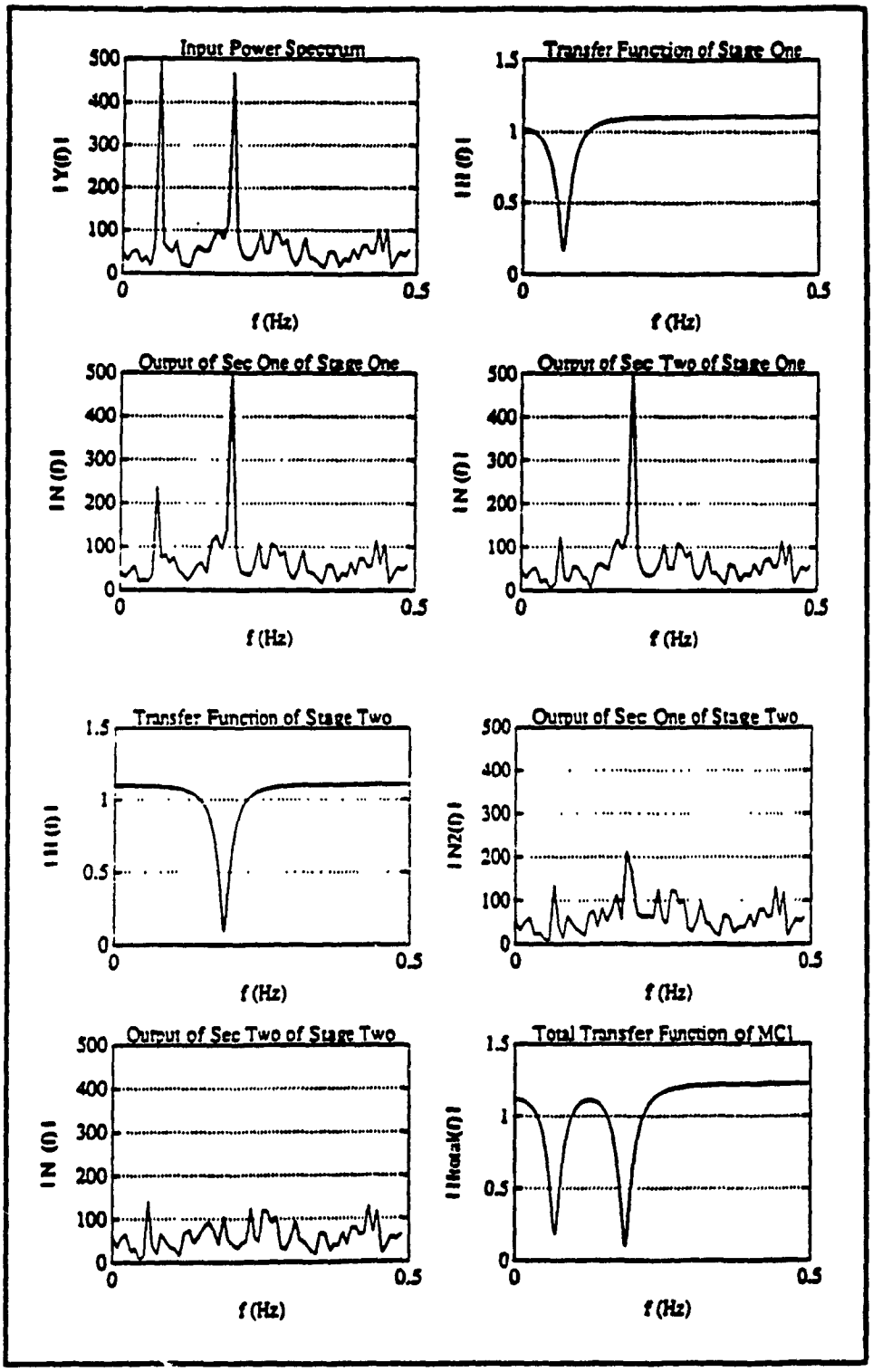

Figure 5.14. Results of Two Stages of MC1 CANF for N = 128 and SNR = 4 dB.

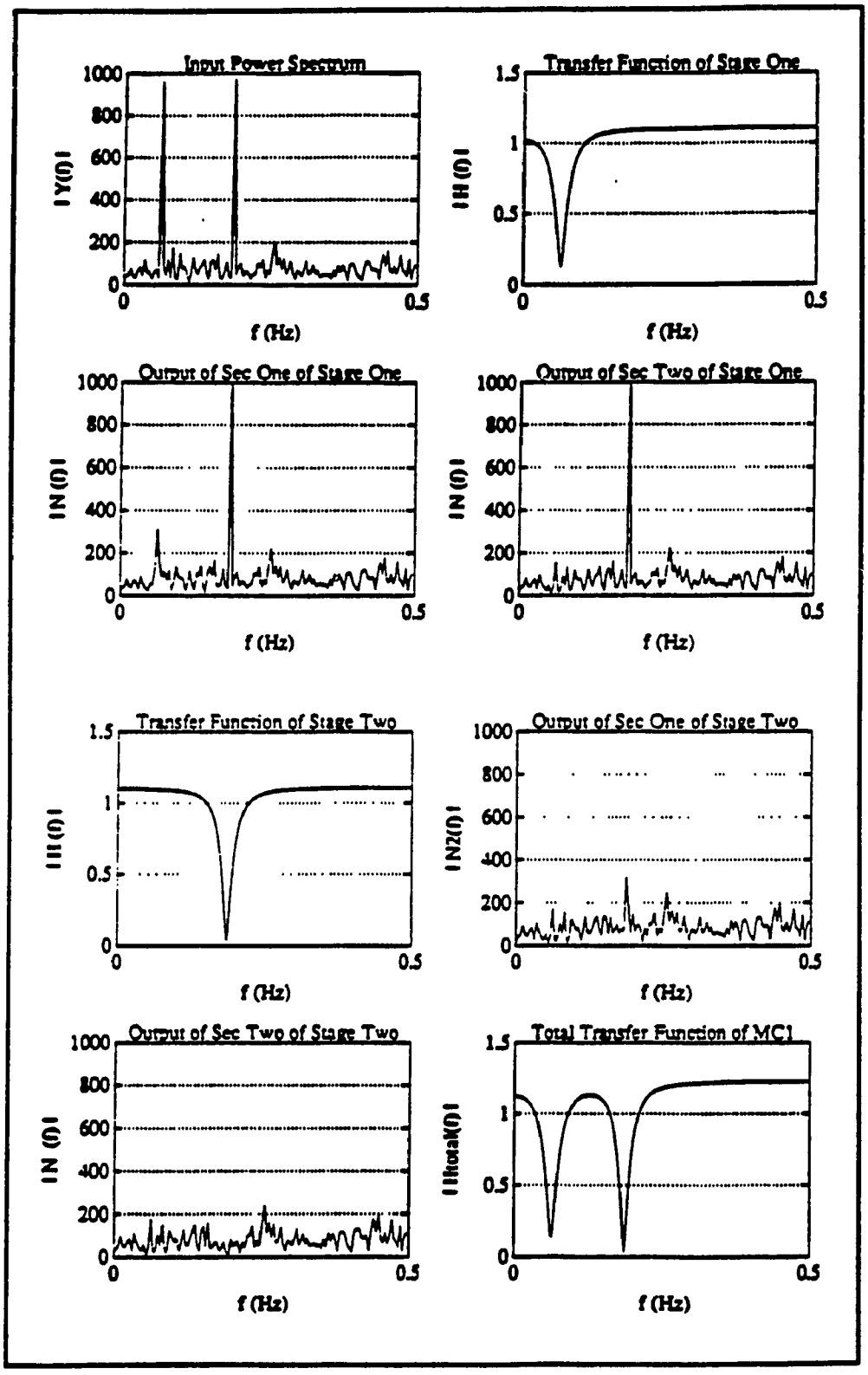

Figure 5.15. Results of Two Stages of MC1 CANF for N = 256 and SNR = 4 dB.

Table 5.7 Statistical Results of Frequency Estimates Using Modified Cascade One (MC1) Implementation

N	SNR = 10 dB				SNR = 04 dB				SNR = 0 dB			
	BIAS		VAR		BIAS		VAR		BIAS		VAR	
	f_1	f_2	f_1	f_2	f_1	f_2	f_1	f_2	f_1	f_2	f_1	f_2
	x 10^{-4}	x 10^{-4}	x 10^{-8}	x 10^{-8}	x 10^{-4}	x 10^{-4}	x 10^{-8}	x 10^{-8}	x 10^{-4}	x 10^{-4}	x 10^{-8}	x 10^{-8}
2048	-1.1717	4.9449	.1962	4.44953	-.8781	2.3693	.7292	5.7066	-.9426	-2.1543	3.0851	11.109
1024	.7445	7.2583	.2285	9.8967	-.7154	5.4911	.8770	8.9244	-.6783	1.9602	4.9792	12.395
512	.2836	10	1.0661	21.759	.7641	10	2.6339	11.743	(1) .6373	(1) 12	18.353	103.35
256	1.8745	15	9.9335	41.187	3.0527	17	13.279	79.769	1.9859	21	180.48	219.95
128	9.3325	22	131.41	142.39	(2) 9.0231	31	370.64	527.73	(6) 6.5515	(7) 39	1148.1	1140

Example 3: From the previous two examples, it was realized that the MC1 implementation performs better than the DF implementation of the CANF. In this example, similar to the previous two examples, we will compare the performance of the MC2 against the results of MC1 of the last example, and see if it does indeed perform better. Each stage of this configuration can be considered to be divided into two sections, first being the CANF parameter estimator, which is then used as the starting guess of the parameters in the adaptation process of the second section. The second section is the MC1 implementation of the last example. Thus, with the extra operation of the first section we can expect improved results.

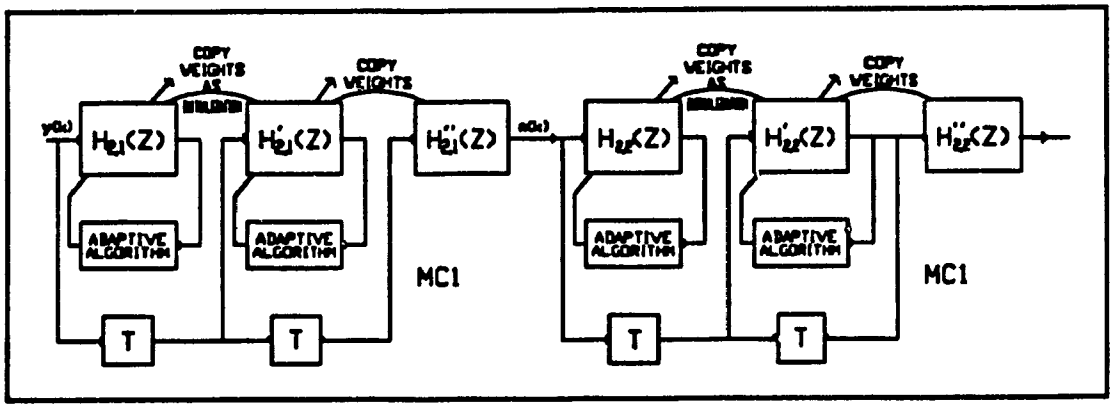

Figure 5.16. Implementation of two stages of MC2.

The notch point statistics of the MC2 are tabulated in Table 5.8. Comparing these results with that of the MC1 (Table 5.7), it is seen that for all cases, there is negligible improvement (if any) in the bias of the notch point. However, two important observations are made with respect to the variance. First, for the smaller observation times (N = 128, 256 and 512), the variance in general has been reduced considerably. If we talk about the total error (Bias squared plus the variance), then it appears that the MC2 does indeed

perform better than MC1. From this result, we can safely say that the knowledge of the neighbourhood of the notch point, however little it may be, can lead to improved results. In effect, this can be considered to be using *a priori* knowledge of the parameter values to move nonlinearly along the convergence path on the error surface, described in chapter 4, towards the optimal or minimal output energy solution. This is particularly useful for short data, as little time is available for adaptation. Thus, starting closer to the optimal solution along the convergence path, one can see the reduction in the number of iterations required in the adaptive procedure. Second observation of importance is the increase of variance for the larger observation intervals at low SNR. For N =1024 and 2048 at SNR = 0 dB, the MC2 implementation yielded an increase in the variance. The explanation for this phenomenon is twofold, which is described in the next paragraph.

First, by the procedure of first estimating the filter parameters values in the first section of MC2, the region of operation on the error surface has moved closer to the minimal energy point. In the second section of MC2, as the filter parameters converge to the optimal value, the output energy will be minimized in the mean. Now, if we continue the adaptation procedure, small perturbations in the filter parameters, due to noise, may lead to a large changes in energy, This is particularly true, if we are operating with energy surfaces that have very sharp minima, which is the objective in the sinusoid suppression using CANFs. Now, examining the parameter update equation in chapter 3, we see that it is a function of output energy. It is then understood that above perturbation in the parameter values, after convergence, may lead to oscillations within the neighbourhood of the optimal solution for a fixed convergence rate. Problems of this type

of instability or more appropriately oscillating misadjustment are inherent in adaptive algorithms where such energy surfaces exist. The second reason for the increase in the variance is due to the finite precision of the computing facilities.

As in the previous two examples, the second basis of evaluation is the ability of MC2 configuration to suppress the sinusoids at the input. Fig 5.1 shows the output results for the case of SNR = 4 db and N = 128. From Fig 5.17d, it can be seen that the sinusoid suppression is similar to that of the MC1 of Example 2 and is considerably better than that of the CC of Example 1. Comparing sinusoids suppression for the case of SNR = 4 and N = 256, the final output in Fig 5.18d clearly reveals an enhanced performance. MC2 is able to suppress the sinusoids to a greater extent than MC1 for N = 256.

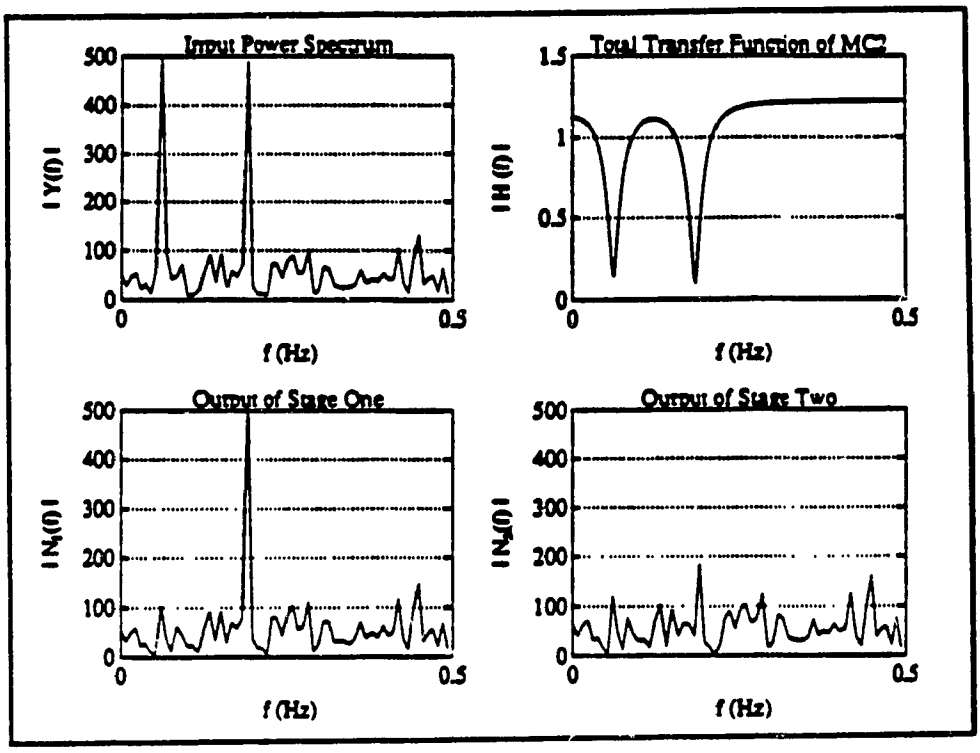

Figure 5.17. Results of Two Stages of MC2 CANF for N = 128 and SNR = 4 dB.

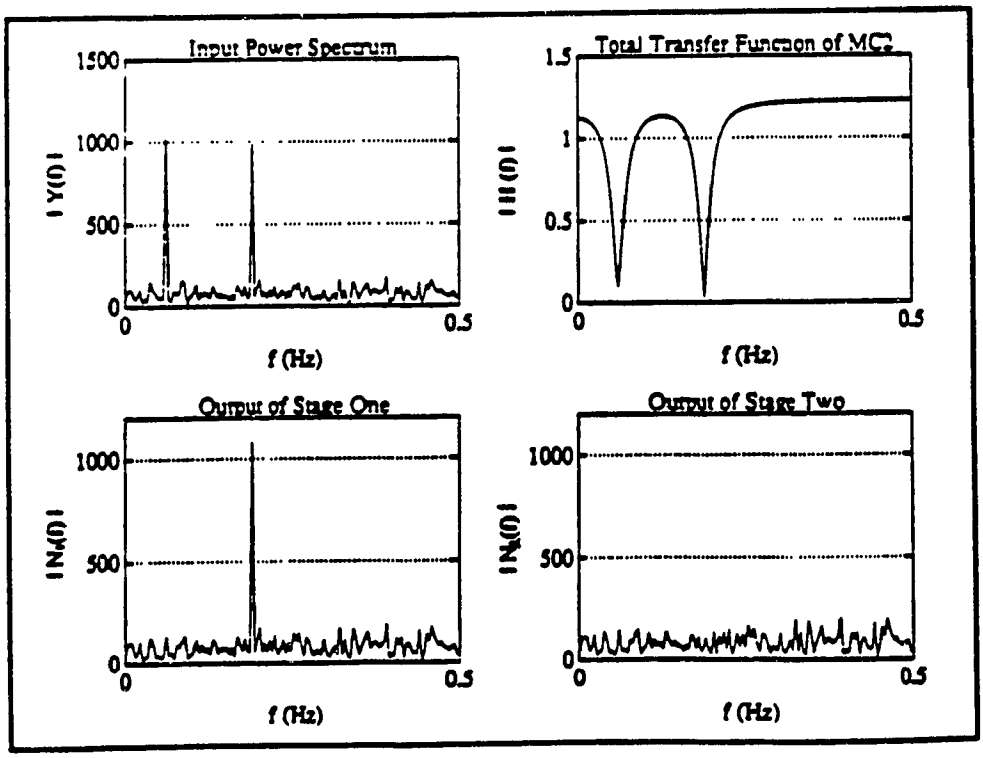

Figure 5.18. Results of Two stages of MC2 for N = 256 and SNR = 4 dB.

Table 5.8 Statistical Results of Frequency Estimate Using Modified Cascade Two (MC2) Implementation

N	SNR = 10 dB				SNR = 04 dB				SNR = 0 dB			
	BIAS		VAR		BIAS		VAR		BIAS		VAR	
	f_1	f_2	f_1	f_2	f_1	f_2	f_1	f_2	f_1	f_2	f_1	f_2
	x 10^{-4}	x 10^{-4}	x 10^{-8}	x 10^{-8}	x 10^{-4}	x 10^{-4}	x 10^{-8}	x 10^{-8}	x 10^{-4}	x 10^{-4}	x 10^{-8}	x 10^{-8}
2048	-1.2490	5.7271	.0916	7.1303	-1.1886	1.6915	.1858	4.7011	-1.0924	-.8453	3.0995	70.558
1024	-.5272	7.8146	1.6513	10.278	-.9005	6.1716	.6234	10.015	.5352	4.2667	16.572	150.18
512	.524	10	1.5710	13.523	(1) .5734	(1) 11	4.0706	28.04	(4) .5775	8.4537	7.3598	77.672
256	1.2557	16	3.2708	30.673	2.7492	21	14.356	208.08	(1) 6.6384	(4) 25	168.99	365.23
128	(1) 7.2441	23	99.349	144.27	3.4852	(2) 26	199.81	241.26	(2) 5.44656	(9) 38	575.39	1130.6

Example 4. In this example we will consider the performance of the CC, MC1 and the MC2 configuration on the basis of total error. The total error is defined as the sum of the variance and the squared bias of the estimated notch point frequency. Fig. 5.19 illustrates the total error performance as a function of the data length for SNR = 10dB. It can be seen that as the data length decreases, the total error in the MC2 implementation is reduced significantly. It is noted that in all three configurations the performance is approximately the same for large data lengths, as it is expected (i.e. asymptotically all three configuration are equivalent).

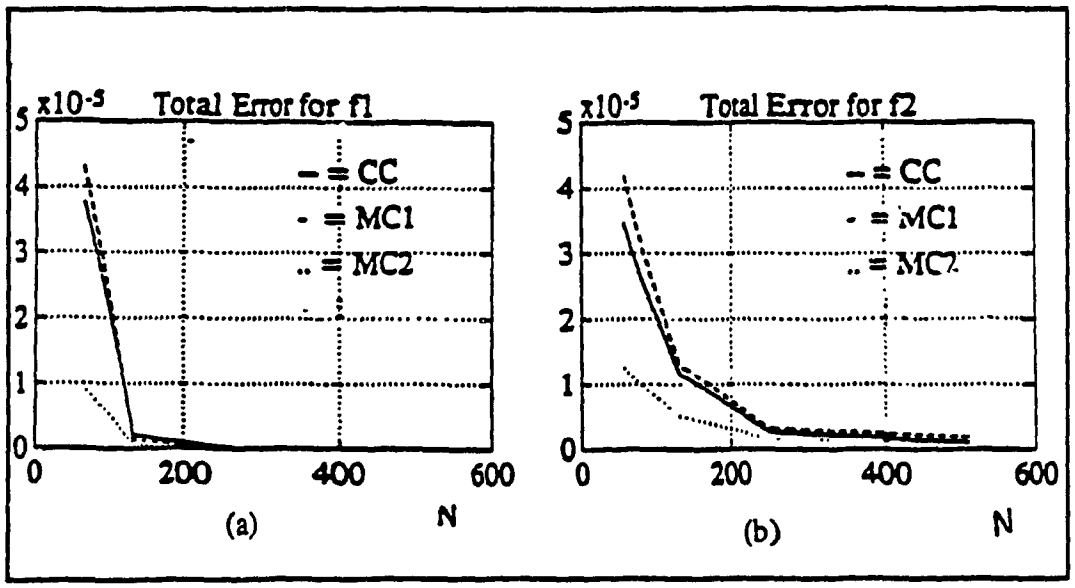

Figure 5.19. Total error performance of CC, MC1 and MC2: (a) total error for f_1; (b) total error for f_2.

CASE: 2 $A_1 \neq A_2$, $f_1 = .0625$, $f_2 = .1875$

Example 5: Having considered the performance of the various CANF implementations for the input sinusoids of equal amplitude, we now examine the performance of unequal sinusoid amplitudes. The experiments in this example are conducted under identical conditions as in the previous case, except for the Sinusoid Amplitude Ratio (SAR = A_1/A_2). In the previous examples, this ratio was equal to unity. We now consider the case of SAR equal to two and four. The objective here is to determine performance of the proposed structure, MC2, for the situation of unequal sinusoid amplitudes. Earlier it was shown that the performance of the MC2 is superior to the other configuration considered, thus, in this example, the performance of MC2 will be compared against that of CC. This example is intended to show the drastic difference in the convergence to the notch points for SAR $\neq$ 1, using CC and MC2.

Tables 5.9 and 5.10 show respectively the notch point frequency statistics achieved by the CC and MC2 implementations. As before, the number in brackets shows the number of outliers that existed in each situation. For SAR = 2 and f_1, it is seen that both configuration yield similar results . However, for f_2, CC is clearly seen to not perform well. In almost all cases, the number of outliers using CC approach the number of trials. This indicates the inability to converge to f_2 and thus the limitations of CC. Using MC2 these outliers do not exist.

Now increasing SAR to 4, it is seen that for f_1, both implementations yield similar results, with the variance of MC2 being slightly lower. For the second notch point (f_2), it is seen that CC implementation does not converge at all. For all data lengths, the

number of outliers is equivalent to the number of trials. In the MC2 implementation, for short data there are large number of outliers, however, as the data length increases to 2048, the number of outliers approach zero. This behaviour of the two implementations further fortifies the improved performance of the MC2 configuration. Thus, it can be concluded that for a mismatch in the input sinusoid amplitudes, the preferred configuration is the Modified Cascade Two.

Table 5.9 Statistical Results of Frequency Estimates Using the Conventional Cascade (CC) Implementation for Unequal Amplitudes

N	SNR_1 = 16 SNR_2 = 10 dB SAR = 02				SNR_1 = 22 SNR_2 = 10 dB SAR = 4			
	BIAS		VAR		BIAS		VAR	
	f_1	f_2	f_1	f_2	f_1	f_2	f_1	f_2
	x 10^{-6}	x 10^{-4}	x 10^{-8}	x 10^{-8}	x 10^{-4}	x 10^{-4}	x 10^{-8}	x 10^{-8}
2048	-.075	26	.09	261.4	(2) .1526	-	.1772	-
1024	.1861	(29) 38	.3006	560.89	2.1221	-	6.8988	-
512	1.1106	(34) 32	1.9664	547.8	1.3398	-	17.57	-
256	1.3472	(37) 31	2.1160	14.663	(1) 3.2628	-	139.41	-
128	4.6652	-	30.992	-	4.8998	-	171.51	-

Table 5.10 Statistical Results of Frequency Estimates Using the Modified Cascade Two (MC2) Implementation for Unequal Amplitudes

N	SNR_1 = 16 SNR_2 = 10 dB SAR = 02				SNR_1 = 22 SNR_2 = 10 dB SAR = 4			
	BIAS		VAR		BIAS		VAR	
	f_1	f_2	f_1	f_2	f_1	f_2	f_1	f_2
	x 10^{-4}	x 10^{-4}	x 10^{-8}	x 10^{-8}	x 10^{-4}	x 10^{-4}	x 10^{-8}	x 10^{-8}
2048	-.1390	6.6857	.0156	7.7418	.1702	11	.1224	65.255
1024	.1408	9.1717	.2984	12.555	.4178	(5) 21	.5858	272.9
512	.6683	9.7435	1.0838	11.174	(1) .5103	(12) 23	.4996	379.48
256	2.4185	14	5.3137	37.033	1.2977	(20) 20	1.8831	147.05
128	3.9937	(2) 21	16.988	141.33	4.5837	(35) 43	155.03	149.98

CASE 3: $A_1 = A_2$, $\Delta f = |f_1 - f_2|$

Example 6: When the problem of separation of multiple frequencies embedded in noise is of concern, it is of interest to quantify the closeness of the frequencies. Since we are dealing with two frequencies embedded in noise, this can be done by the proximity difference, Δf. We define this as the difference between the two normalized frequencies. Multiplying this factor by the number of data samples, the well encountered time-bandwidth product can be obtained. In this experiment we evaluate the performance of the MC2 configuration, it being the best of the configurations considered. To be more specific, it is intended to show the total error achieved by the MC2 configuration as a function of Δf.

For each value of the proximity difference, the total error is evaluated from 40 independent trials. SNR for every trial is 5 dB as defined in the previous cases. The lower frequency is kept constant, f_1 = .0625 Hz and f_2 is decreased from .1875 Hz to .075 Hz in steps of .00625. In order to avoid any spurious estimates degrading the statistics, the resulting estimates were gated within ± 10% of the true frequency values. In this way total error can be evaluated as a function of Δf.

In chapter 3, it was said that the debiasing coefficient, α, has two conflicting functions. It controls the bandwidth and the sharpness of the notches for the suppression of the sinusoids and it also controls the speed of convergence. That is, if we desire very sharp and narrow bandwidth notches (which is necessary if the proximity difference is small) then the value of α must be chosen close to unity. In such a case, the length of

the input data must be increased appropriately to achieve a required error. Thus, one can see the tradeoff that exists in the closeness of frequencies and the data length for a fixed value of the debiasing coefficient.

We perform this experiment for three different values of the debiasing coefficient, α, and fixed data length of 256 samples. In this way, we may see how the total error behaves as a function of the proximity or the time-bandwidth factor for a specific value of the debiasing coefficient and data length.

Figures 5.20 and 5.21 show this result for α = .9 and .95, respectively. We will leave the third experiment for later. It is observed that for α = .95 (the higher value of the debiasing coefficient or effectively the narrower notch bandwidth), the two frequencies are resolved with a lower total error and the error curves of Fig 5.21 are smoother. By increasing the debiasing coefficient to .95 from .90, it is possible to resolve frequencies for a lower Δf.

The explanation for lower error are inherently due to the choice of the debiasing coefficient. As α approaches unity, the algorithm will converge to the notch point frequencies with greater accuracy. This is due to the pole-zero placement nature of the CANFs (see chapter 3). Of course, the earlier mentioned tradeoff limits the value of α that can be used in practical situations.

In order to resolve the two frequencies as Δf gets smaller, it is necessary for the notch bandwidths to be small. They must be small such that there is no overlap in the notches. Since α is kept constant to .9 for the results of Fig 5.20, as Δf approaches the

lower end, there is more and more overlap in the notches. In such situations, both stages may converge to frequencies in between f_1 and f_2 and thus the large error. Of course, it is in this situations that there are likely to be larger number of outliers. This was observed for the case of Fig 5.20, where the lack of consistency in the error curves is due to a large number of outliers.

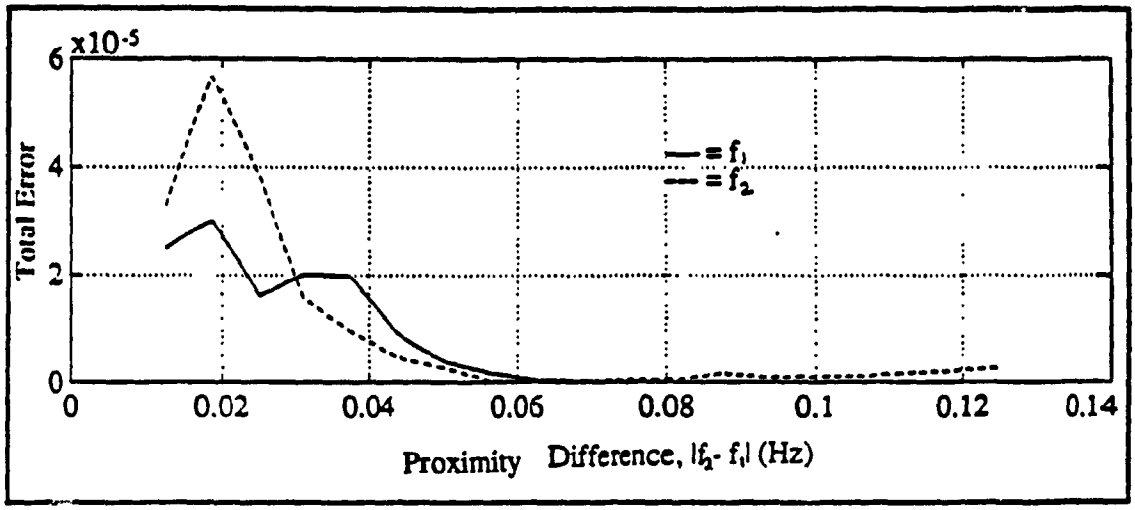

Figure 5.20. Total error of the MC2 implementation as a function of the proximity difference for α = .90.

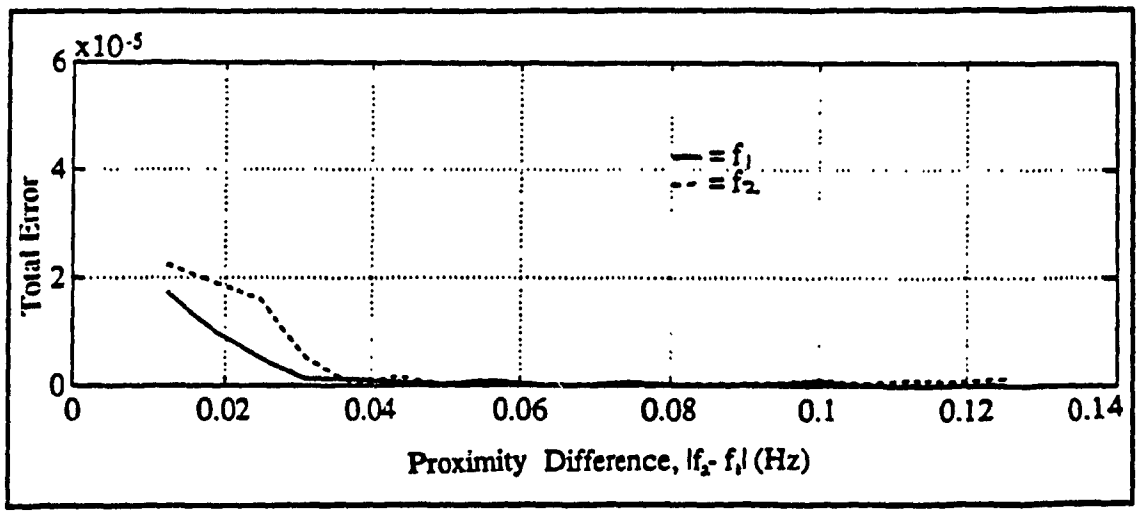

Figure 5.21. Total error for the MC2 implementation as a function of the proximity difference for α = .95.

We are considering the MC2 configuration for the separation of two closely spaced frequencies, in which it is seen that the task of sinusoid frequency estimation and the suppression has been separated. Each stage in Fig 5.16 is composed of three sections. Since the first two sections are used to estimate only the notch point, it is possible to use a value of the debiasing coefficient that is close to unity. Once the notch point frequency is determined, we may use a smaller value of α for the sinusoid cancellation in the third section of each stage, which is used for sinusoid suppression. In this manner the conflicting roles of α can be separated.

Chapter 3 described the idea of using a time varying debiasing coefficient. Since the first two sections of the MC2 configuration are adaptive, we may use an exponentially increasing α described by Eq 3.49. For the reasoning and the advantages of using a time varying debiasing coefficient refer to chapter 3. Now, this third experiment, for the case

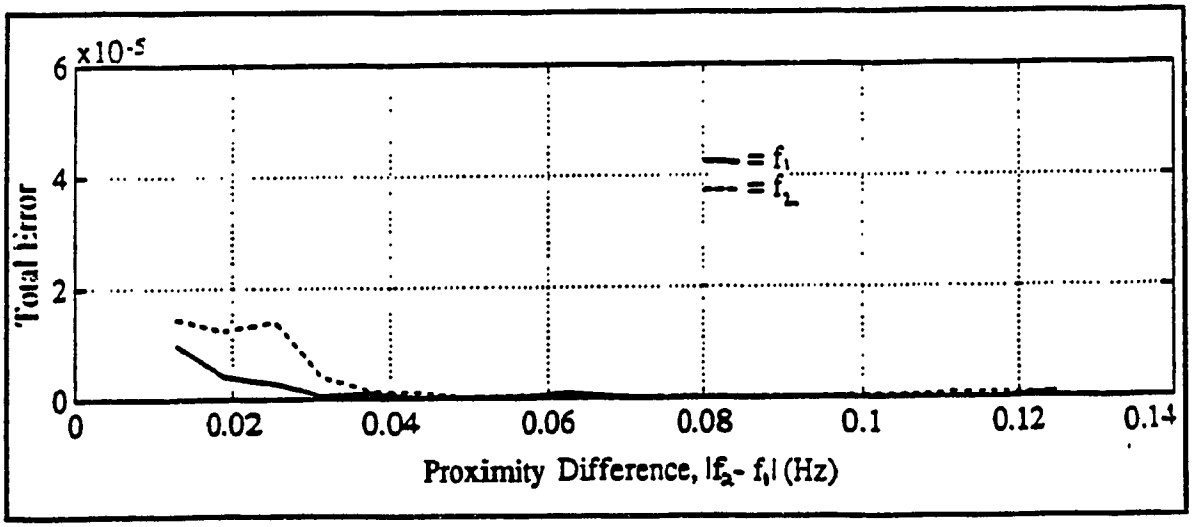

Figure 5.22. Total error for the MC2 implementation as a function of the proximity difference for the time-varying debiasing coefficient.

of total error vs. the proximity difference for the MC2 configuration is performed exactly as the last two, except we use a time varying debiasing coefficient for the second stage section of each stage. The following values were used in the update of the debiasing coefficient; $\alpha_o = .99$, $\alpha_\infty = .999$ and $\alpha_{start} = .96$. For the first section, a constant nominal value of $\alpha = .90$ is used to obtain an estimate in the neighbourhood if the notch point frequency. Fig 5.22 shows the plot of total error vs. Δf. It is seen that the two frequencies in this situation can be resolved for a smaller Δf with lower error. It is also observed that the outliers were virtually eliminated leading to the much smoother error curves. It is possible to achieve the separation of frequencies with an even lower Δf by manipulating the parameters in Eq 3.49. The choice of optimal parameters is dependant on the specific case concerned and can be evaluated empirically.

5.3 References

1. K. Scarbrough, N. Ahmed, G.C. Clifford, "On the simulation of a Class of Time Delay Estimation Algorithms," IEEE Trans. on Acoustics, Speech and Signal Processing, vol. ASSP-36, June 1981.

2. J. Krolik et al, "Time Delay Estimation of Signals with Uncertain Spectra," IEEE Trans. on Acoustics, Speech and Signal Processing, vol. ASSP-36, December 1988.

3. D.V. Rao and S.Y. Kung, "Adaptive Notch Filtering for the retrieval of Sinusoids in Noise," IEEE Trans. on Acoustics, Speech and Signal Processing, vol. ASSP-32, August 1984.

4. B. Friedlander and J.O. Smith, "Analysis and Performance Evaluation of an Adaptive Notch Filter," IEEE Trans. on Information Theory, vol. IT-30, March 1984.

5. A. Nehorai, "A Minimal Parameter Adaptive Notch Filter with Constrained Poles and Zeros," IEEE Trans. on Acoustics, Speech and Signal Processing, vol. ASSP-33, August 1985.

6. T. Kwan and K. Martin, "Adaptive Detection and Enhancement of Multiple Sinusoids Using a Cascade IIR Filter, " IEEE Trans. on Circuits and Systems, vol. CAS-38, May 1990.

7. J.F. Chicharo and T.S. Ng, "Gradient Based Adaptive IIR Notch Filtering for Frequency Estimation," IEEE Trans on Acoustics, Speech and Signal Processing, vol. ASSP-38, May 1990.

Chapter 6

Auditory evoked Brainstem Audiometry

The previous chapter consisted of simulation results of various Constrained Adaptive Notch Filters and the Time Delay Estimation. Having described the theory and studying some results via simulations, we now discuss the application to the analysis of Auditory evoked Brainstem Responses (ABR). Before discussing the results of this application, this chapter will describe the essence of ABR. Emphasis is placed on familiarizing the reader with the idea of ABR audiometry. Its drawbacks and motivation for the need of more powerful processing algorithms are presented.

Non-behavioural hearing assessment has gained widespread acceptance in Audiometry. By non behavioural, we mean that a patient need not respond to whether a particular audible stimulus was heard by him or not, thus, avoiding any patient bias or subjectivity. Evoked potentials or evoked responses (ERs) have gained an increasing acceptance as a non-behavioural tool to measure acoustical sensitivity of the auditory path in the difficult to measure patients - infants, comatose, and handicapped. From these responses, it is possible to determine various aspects of a patient's hearing capacity.

ERs are obtained by applying an acoustical stimulus and then measuring the electrical activity of the brainstem by surface electrodes placed on the scalp. Measurement of this type impose one drawback, the measured response is embedded in the background EEG activity. Of course, we may use implanted microelectrodes in the brainstem to localize the interference by the background EEG activity, however, one can understand its lack of clinical feasibility for simple hearing loss test. Thus, a major engineering task

is to separate the ER and the background EEG. This chapter will explore the idea of ERs as a tool in audiological clinical environment. The first section will define What are Evoked Responses? The second section will describe the use of these evoked potentials in Audiometry. In particular, it will explore the waveform morphology of the auditory ERs, the method of acquisition and the parameters of interest from the response. Section three will describe some of the existing algorithms for processing the acquired waveforms to reveal the ABR with emphasis on the correlation method, since this is the method most widely used. Two examples using the correlation method will be used to show the need for development of new signal processing algorithms.

6.1 What are Evoked Responses ?

Over the years many different tools for the analysis of the brain have emerged. One would think that its inner workings are well understood by now. Computer generated images of the brain anatomy fill number of medical journals, showing the results of NMR and CT scans. None of these impressive techniques offer any idea of how the brain is functioning, its ability to think, or how the information is transported back and forth. From a psychiatric point of view, a useful tool would then be a way to decipher human cognition or from neurological view how the brain controls various bodily functions [1].

There exists one primitive test, that does shed some light on how information is shuttled about the brain. By primitive we mean that since it was first introduced in 1950s by a physiologist named Dr. George Dawson, it hasn't changed much. Consider stimulating a patient's senses - a small electrical shock, a flash of light or a tone in the

ear - which results in electrical response being generated that eventually go to the brain. We are able to measure these responses at the brain by electrodes pasted to the scalp [1].

These types of responses are known as ERs. They are basically electroencephalograms (EEGs). But, unlike the ordinary EEGs (which show continuous electrical activity of the brain), these responses are elicited in a controlled manner. That is, ERs mark the brains transient response to a particular stimulus, such as sound. Another way of looking at this would be to say: How does the brain react to hearing a sound. As the sound (the stimulus) travels from the ear, through the brainstem and to the higher brain, a series of pulses are generated along the way, causing scalp potential to vary with time. From this, one can determine the reaction of a particular area along the pathway [1,2].

If the scalp electrical recording is unusual (response appears later than expected, has a different shape than normal and so on), then a physician may conclude something is wrong. In this manner neurologists are using ERs to diagnose brain tumors, multiple sclerosis and hearing loss to name a few. Psychiatrists have hopes of using ER to one day have insights into human cognition. Some researchers believe that someday, evoked potentials will help human-factor engineers design better cockpits, unbeatable lie-detectors and even allow them to design equipment operated by a mere thought [1].

From the above discussion, it seems that the hopes of using ERs are great, but the obstacles in the development of this tool are also great. The problem with ERs is that they contain too much information. The electrodes not only pick up the response to the stimulus but also pickup the ongoing EEG - random firing of billions of nerve cells. That

is, even the tiniest action of the body or the mind will result in some sort of electrical activity in the brain - for example, electrodes detect activity from a blink of an eye, swallowing or even the 60 Hz noise from the nearby lightbulbs [1].

To extract, the small ER (sometimes as low as a nanovolts) in the midst of large background noise requires considerable amount of computing power. Even though the idea of ER was developed about forty years ago, it only found its way into medical laboratories when computers grew cheap. Its full capacity can only be realized when more powerful software and processing algorithms are developed to separate various components of the EEG [1].

Most neurologists, today measure ERs with the same processing tool used forty years ago. Dr. Dawson first applied the technique of signal averaging. He reasoned that by stimulating a patients arm, he would be able to view the ER on an oscilloscope linked to the scalp electrodes, if he could find a way to enhance the 5 to 10 microvolt time-locked ER in the 50-100 microvolt EEG. He exposed his oscilloscope tracings to a photographic film. By stimulating the patient many times and superimposing the tracings on the film, the time locked evoked response began to stand out amidst the random EEG. One major assumption in his reasoning was time invariance of the evoked response. That is, each stimulus will elicit a similar response, independent of time of the stimulus. Today, instead of using a photographic plate a computer based averaging technique is used. In 1958, engineers at Massachusetts Institute of Technology in Cambridge, designed a computer to do this same job digitally - averaging a number of evoked tracings to reveal the evoked response [1].

6.2 Evoked Potential in Audiology

In the last section, the notion of ER was developed. It was said, that it is the most promising tool in psychiatry and neurology. We will now explore the idea of ER in a branch of medicine known as Audiology. The use of evoked responses in Audiology is wide, thus in this report we'll concentrate on Auditory Evoked Brainstem Response (ABR) to determine hearing threshold of individuals. We'll describe what are ABRs and their waveform morphology, how the measurements are taken, and what are some of the parameters that can be used to diagnose hearing loss.

First, the question of What is ABR? As the name implies, Auditory Evoked Brainstem Response is an ER, obtained from the brainstem, elicited via auditory stimulus. That is, by applying a auditory stimulus to the ear, the surface electrodes placed on the scalp detect the electrical response of the brainstem.

6.2.1 ABR Waveform Morphology

Dr. Dawson's reasoning in establishing the measuring technique of the ER was the repeatability of the response. That is, for every stimulus applied, the measured response will be similar. While this may or may not be true in other ER applications; it is definitely applicable in the ABRs. Irrespective of the patient's age, hearing sensitivity, or his waveform morphology, a repeated suprathreshold stimulus will produce a similar response [3,4]. By suprathreshold, we mean stimulus level above the hearing threshold (i.e. lowest intensity level of the stimulus at which the brainstem response can be detected). Thus, repeated stimulus will produce a similar response. This idea of

repeatability is of great significance as far as the measuring of these waveform is concerned, as will be seen later.

For the purpose of description of the ABR waveform, we'll use a waveform elicited at much higher stimulus level than the threshold, say 80 dBs. Fig 6.1 depicts a typical suprathreshold waveform in time domain. The waveform of this figure is marked with an impulse at the start of the response. The impulse is only a representation of the stimulus to indicate the start of the response. Thus, a fraction of a second after the person hears a click (instead of tones, a click is used as a stimulus for the ABRs), the ABR waveform is observed for the first 10 millisecond. The response consists of a series of peaks and valleys. Fig 6.2 shows the areas in the auditory tract, where each of the component is generated. The first component of the waveform is generated near the inner ear; components 2 through 5 arise somewhere in the brainstem and the location of generation of components 6 and 7 is still ambiguous. By the time the 10 milliseconds have elapsed the response has probably reached the cerebral cortex, the higher brain. In this region, the response is processed as cognition and is of no use as far as auditory evaluation is concerned. The locations of generation of these component responses in Fig 6.2 are only approximate as no one knows the exact point of generation. In all likelihood, there is more than one source for each component [1].

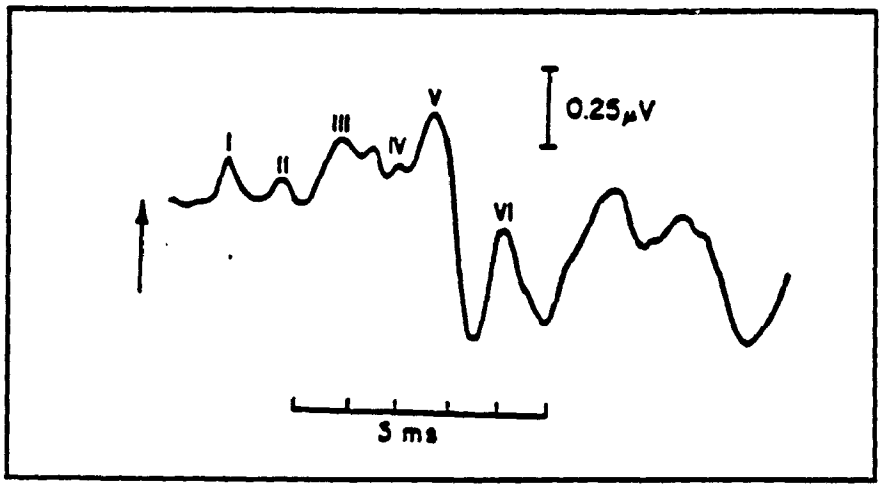

Figure 6.1. Auditory Evoked Brainstem Response.
[adapted from [3]]

Figs 6.1 and 6.2 show the ABR waveform that is generated in normal healthy individuals. There exists a small variation in response from person to person, however, the general waveform shape is preserved. There have been attempts to categorize all possible waveforms into a finite set that can be used for template processing [5].

We've looked at the ABR in time domain up to now. We'll now consider this response in frequency domain, it will prove to be useful in the analysis or in the recovery of the signal from the noise. The spectral energy of the human ABR appears to decrease up to about 2000 Hz and flattens above 2000 Hz. From the present studies available, spectral energy is limited to 1500 Hz. There are three main components: a low frequency component at around 100 Hz, in literature there appears to be some controversy about the existence of this component; a midfrequency component around 500 Hz; and a high frequency component at 1000 Hz. Fig 6.3 shows a typical spectra of an ABR. Studies suggest that the midfrequency component is associated with component 5 of Fig 6.2, and the high frequency component is associated with the early components, 1 through 4.

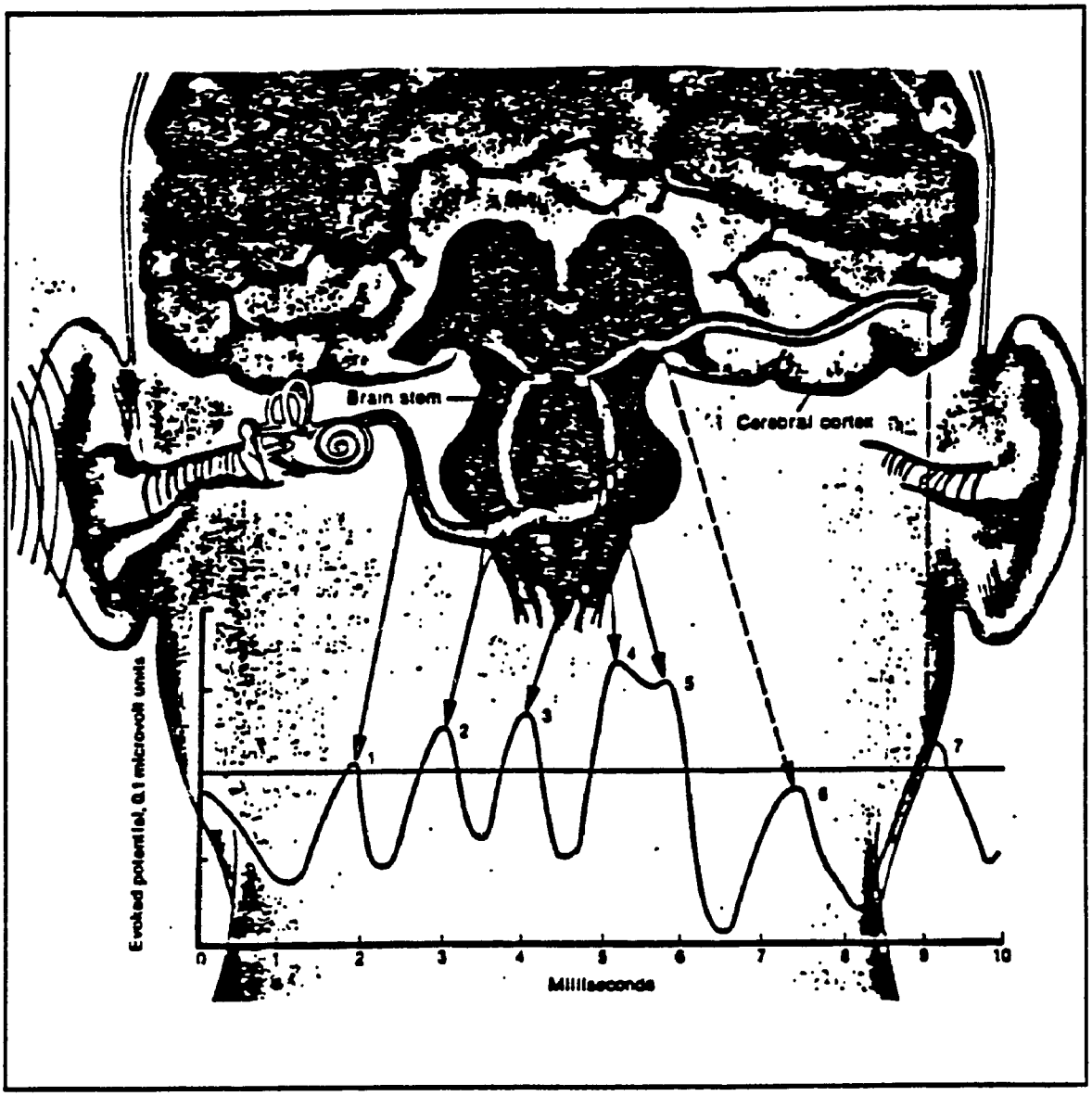

Figure 6.2. Location of generation of components of ABR. [adapted from [1]]

These two frequency components are probably the most important for hearing threshold and other diagnostics [6].

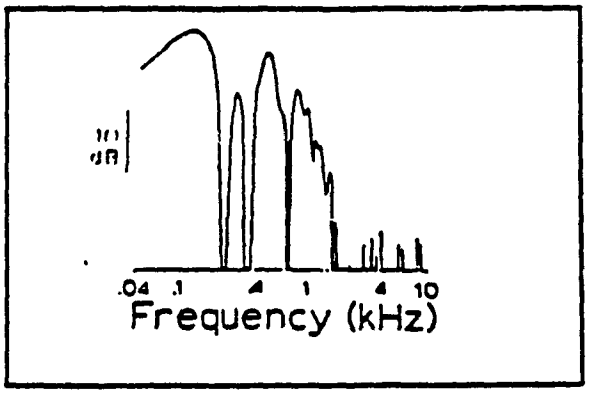

Figure 6.3. Spectra of ABR.
[adapted from [6]]

6.2.2 Measurement Techniques

ABR waveforms, unfortunately comprise about 1 percent [7] of the ongoing EEG activity and thus must be extracted by some signal processing mean. Earlier, it was said that the technique that was used forty years ago, is still in practice today - the idea of signal averaging to eliminate noise. Consider a digitized signal (k refers to discrete time and the subscript i refers to i'th run in the averaging process)

$$y_i(k) = x_i(k) + n_i(k), \tag{6.1}$$

where x(k) is the evoked response and n(k) is the background noise. Now, if the noise is assumed to be white (there is no correlation in individual samples) and is uncorrelated with x(k), then we can enhance the signal x(k) by applying signal averaging technique. Since every response y_i will have a noise component n_i that is uncorrelated from response to response (results of whiteness of the noise) then by using this random nature, it is possible to sum a large number of responses to reduce the effect of noise. In the following equation

$$\lim_{M\to\infty}\sum_{i=1}^{M} y_i(k) - \lim_{M\to\infty}\left[K\sum_{i=1}^{M} x_i(k) + K\sum_{i=1}^{M} n_i(k)\right] \approx \hat{x}(k) \qquad (6.2)$$

if M approaches a large number then the effect of the noise can be minimized, revealing a clean ABR component. In Eq 6.2, K is a normalizing factor, x(k) is the ABR, n(k) is the noise and $\hat{x}$(k) is the estimated ABR. If the ABR component of the signal y is not similar from run to run, then it is obvious that some of it will also be cancelled. For a further discussion of the stochastic nature of signals refer to [8,9].

There are many different ways to record these responses, one such system is shown in Fig 6.4. Standard EEG disk electrodes are attached to the vertex, the mastoid and the forehead which is used as a ground reference. The preamplifier can have a variable gain which is usually around 100 within the cutoff frequencies of about 100 Hz and 3000 Hz. The output of this preamplifier is connected to an amplifier of a gain around 1000, which is also adjustable. The amplifier is connected to a filter which is setup as a low pass with a cutoff of 5kHz. This filter is used to remove all high frequency spikes in the EEG and prevent them from being averaged in the response. Finally a microcomputer based data acquisition system is used to average and store the waveforms. The applied auditory stimulus is a very small pulse with its intensity adjusted by a decade counter. The reason for using a pulse instead of a tone is that a pulse is composed of many harmonics. Thus, it is capable of exciting the neurons in the auditory tract at all frequencies, not just a single tone. In this manner, ABRs at different stimulus intensity levels can be obtained. Through an earphone, the click or the pulse stimulus is monaurally delivered to a patient relaxed in a supine position in a sound-attenuated chamber. Once

an averaged response corresponding to a particular stimulus level is obtained then the operator can manually change the intensity level of the stimulus for the next ABR. In this manner a typical ABR testing procedure will involve a collection of a set of responses, associated with different stimulus levels, for each ear [7].

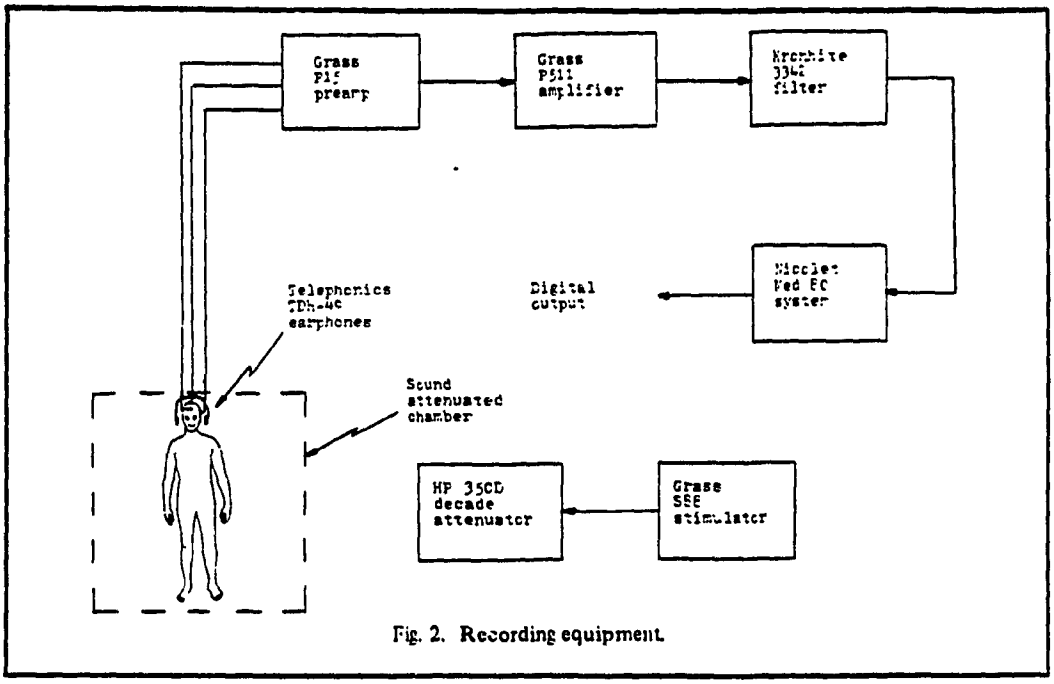

Fig. 2. Recording equipment.

Figure 6.4. ABR recording technique. [adapted from [7]]

6.2.3 Evaluation

Up to now, the idea of evoked potentials as applied to ABR and their measurement techniques have been described. This section will describe how these waveforms are used in audiological assessment. That is, the parameters of the ABRs that are of interest.

ABR waveforms can be used to diagnose various aspects of the brain and the hearing. Here, we'll concentrate on two very important deductions from ABR. They are the hearing level threshold and the detection of neurological disorders in the auditory tract [1,2,7,10,11,12].

A typical ABR recording session lasts about 45 minutes. The subject is placed in a supine position in a sound-attenuated chamber. The measurements are obtained in an ipsilateral configuration (stimulus is applied to the ear that is being tested), however, contralateral (stimulus is applied to the ear opposite to the one being tested) measurements are taken in certain cases. In a general procedure, the examiner begins by obtaining a response with a high (suprathreshold) stimulus level and then in a predetermined step size decreases the stimulus level to obtain a set of responses for each ear. He also obtains a response with no stimulus, a control. Each ear is tested independently.

In the determination of the hearing "threshold" or the lowest intensity level at which a response is detected, it is necessary to collect averaged responses at various intensity levels and then determine whether an ABR is present or only background EEG exists. Thus, in this manner the only parameter of interest is the amplitude of the ABR, that is, it is only required to distinguish the presence or absence of the ABR, rather than its shape. At intensities much higher than a person's threshold, ABRs are easily detected upon visual inspection, however, as the intensity approaches a person's threshold, the response is marred by the background EEG activity and it becomes a difficult task for the clinician to evaluate these waveforms. Fig 6.5 shows a set of ABR waveforms for left ear

of one subject. Evaluation of this set yielded a threshold of 50 db. Notice that the waveforms at suprathreshold stimulus levels are clean and easily detected, however, at levels below and close to the threshold, it is difficult to determine the presence or absence of the ABR [4,10,11].

The second diagnosis that is available from these tests is the ability to detect neurological disorders. Upon examination of the set of responses, a latency vs. stimulus intensity (LI) curve can be determined. For the purpose of developing the LI curve, overall latency of a particular response is measured as the time delay of the fifth component relative to the delay of the previous response (the response due to a stimulus one level higher). In Fig 6.5, the latency of the fifth component is shown as the thick dark curve. A large deviation of the LI curve from the normal may lead an audiologist to consider the possibility of neurological disorder. It is possible that some sort of tumor has developed. Another indication of a tumor may be the intercomponent latencies of a particular response. That is, for example, if the delay between component one and component five of a response at some stimulus level is out of the norm, then it may lead to an indication of other disorders [2,7,9].

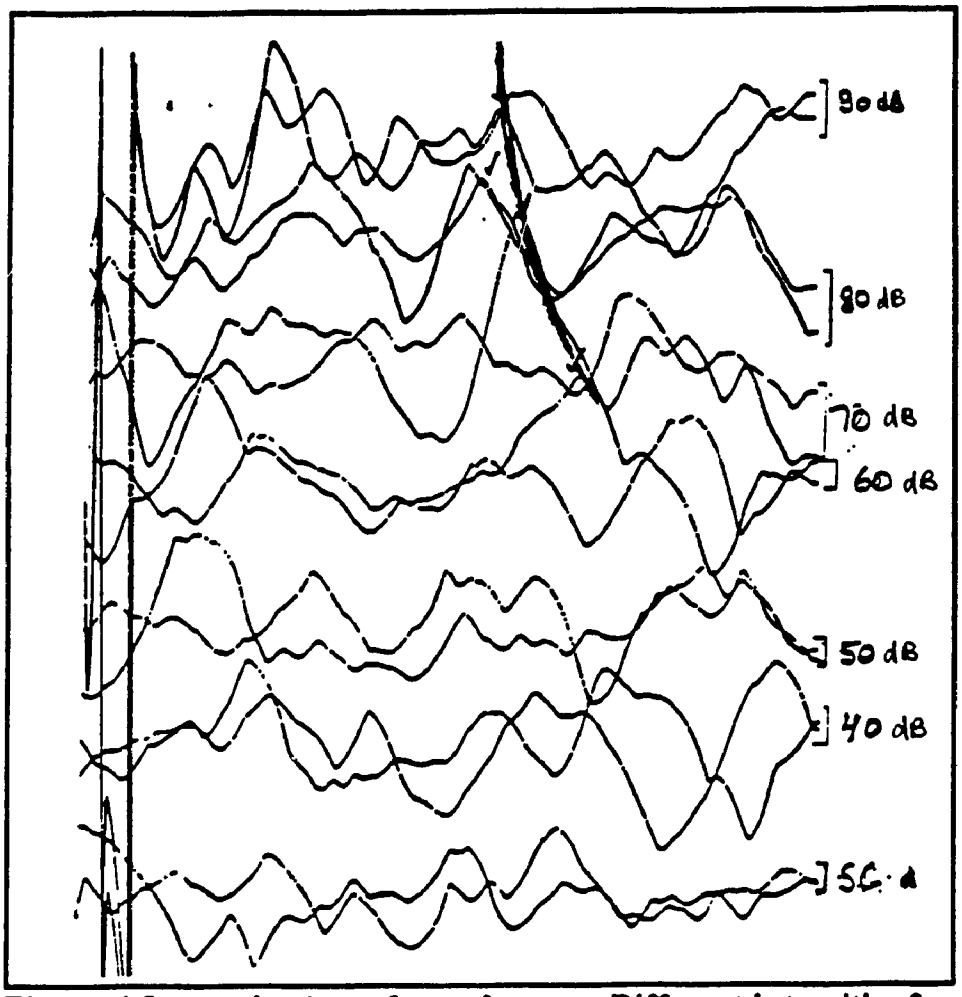

Figure 6.5. Acquired set of waveforms at Different intensities for one ear of a subject.

6.3 Present Methods of Evaluating the Measured Signals

Whether an ABR is judged as being present or absent depends on the testers experience in evaluating the waveforms. These judgments of the waveforms can become subjective since the tester may have some prior knowledge of the patient history or he may simply see the waveforms differently on different days. When the ABRs are particularly noisy, the tester bias becomes increasingly important and may lead to erroneous evaluations. Another problem that exists in this kind of visual evaluation is

intertester reliability. Different testers may adopt different criterion for accepting a waveform as containing an ABR. Thus, evaluation of the ABRs for hearing threshold is not a wholly objective procedure [11].

To eliminate the subjectivity in the evaluation process, various ideas have been proposed in literature. The goal has been to develop a mathematical tool to detect the presence of the ABR. Weber and Fletcher [4] have proposed an algorithm based on correlation between pairs of waves obtained with and without acoustic stimulus. Wong and Bickford [12] compared the power in the ABR waveforms with and without acoustic stimulus. Salamon [14] applied a non parametric analysis of the variance test. All the above algorithms were found to have same statistical validity as the visual detection method, however no extensive studies have been done to dismiss or accept the usefulness of these methods. By visual detection, we mean that a clinician scores the presence/absence of the ABR by looking at the averaged waveform. Recently, Arnold [10] compared visual detection against (1) correlation [4]; (2) variance ratio [12]; and (3) multiple pre-post Z test [15]. The results indicate that the visual detection was the most sensitive, but it was not more statistically sensitive than the correlation method. In the following, a description of the variance and the correlation method will be given with emphasis on the correlation method. Two example of hearing assessment will be shown using the correlation method.

6.3.1 Variance Ratio [12]

Using the variance ratio test, in determining the hearing threshold, a slightly modified measuring technique is used. Tester obtains non-stimulus waveform for each ear

as in the ususal measurements. However, in using the variance ratio test for detection, a nonstimulus waveform corresponding to each stimulus waveform is needed. Instead of using the standard 10 millisecond window, a 20 millisecond window is used with stimulus being applied at the 10 millisecond mark. This is shown as the pre and the post stimulus waveforms in Fig 6.6. The reasoning behind using a separate control run for each stimulus is that it is expected that the background activity may have some similarities in continuum of time and thus the influence on the pre and post section may have similarity to a certain degree.

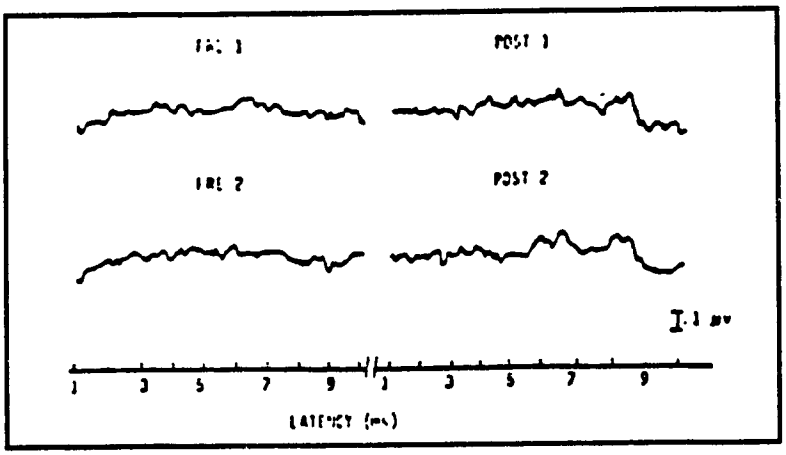

Figure 6.6. Example of variance ratio detection method for a 10 dB SL trial. [adapted from [10]]

Every waveform obtained has fluctuations about some baseline. The extent of these fluctuations can be quantified by computing the variance. The variance will be related to the size of the peaks, thus, the pre-stimulus waveform will have a lower variance (it is only noise, no ABR is present) as compared to the post-stimulus waveform. The pre and post refer to waveforms obtained without and with stimulus, respectively. The variance is computed as

$$var = \frac{1}{N}\sum_{i}^{N}(x(k) - \bar{x})^2, \tag{6.3}$$

where x(k) is the value of the waveform at time k, $\bar{x}$ is the mean value of all points along the waveform, and N is the number of points in each waveform.

To detect the presence of ABR, it is necessary to determine whether the variance of the poststimulus waveform is greater than the variance of the prestimulus waveform. The ratio, var(post)/var(pre), is used to compare the variability of the pre and poststimulus pairs of waveforms. If the ABR is not present, then the ratio will be close to unity as both pre and post waveform comprise only of noise. If the poststimulus contains an ABR, then the ratio should be higher then unity. The higher the ratio, the greater the certainty of an ABR being present. For each stimulus level, the measurements are taken twice and the presence of an ABR is judged, if ratios from the two trials are greater than some criterion value. This criterion value can be determined by an empirical study.

6.3.2 Correlation Method [9]

As in the variance ratio test, the correlation method requires a slightly modified approach in the general waveform acquisition. In this method we do not need pre and post type of measurements, however, it is required that two waveforms be acquired for each stimulus level. A typically acquired waveform set of one subject's left ear is shown in Fig 6.7.

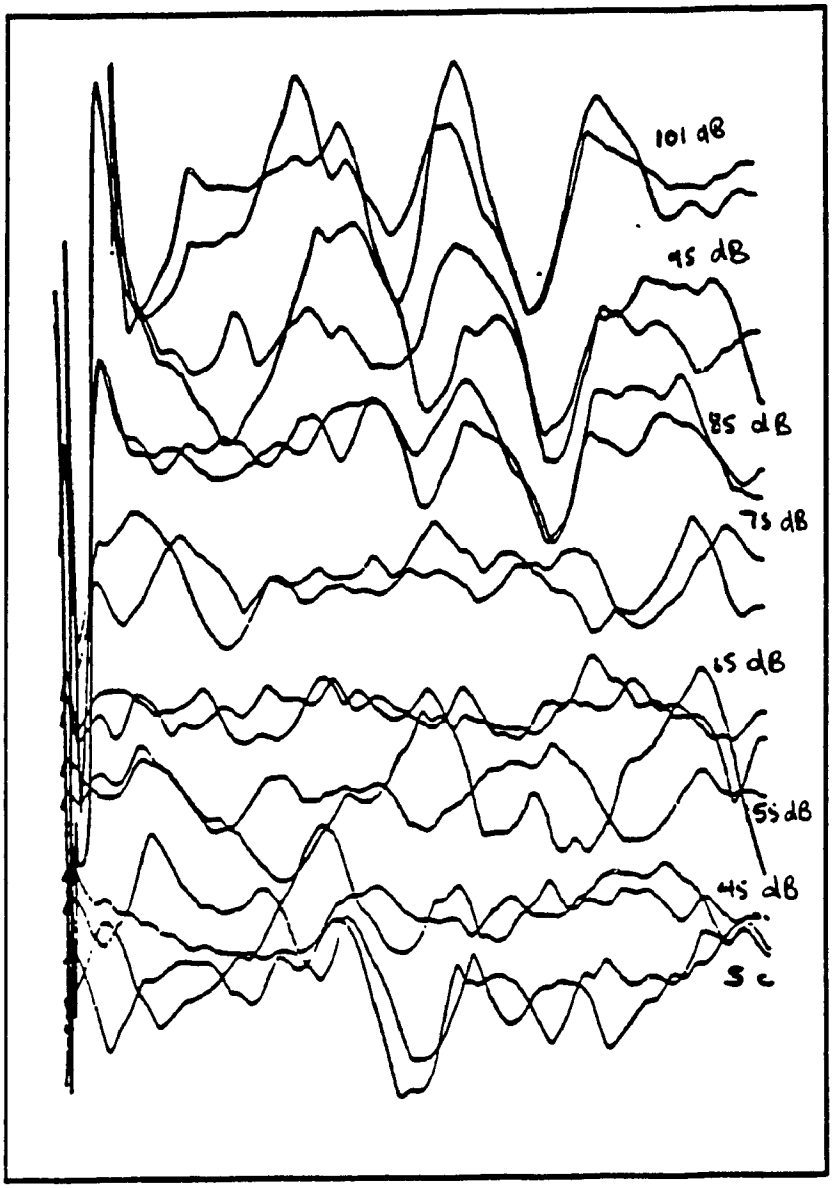

Figure 6.7. A typical set of ABR of one ear.

The rationale for using correlation analysis is based on two factors. First, it uses the idea of repeatability, two waveforms obtained from same stimulus must be correlated to some degree. Secondly, the waveforms are composed of a number of points (usually 256) representing discrete values at equally spaced time intervals and are stored as such. It is possible to use the Pearson r correlation [15] to compare any two waveforms. Thus,

a series of digital values of one waveform can be correlated to the corresponding digital values of another waveform.

Using the Pearson r correlation, it is possible to develop a mathematical tool for detection of ABR. If for example, a stimulus well above the threshold is presented to a patient, the averaged waveform will clearly reveal the ABR. A waveform obtained in a second trial with same stimulus intensity level will be similar. Then, if the two waveforms are correlated, a positive correlation coefficient will result. Conversely, if the applied stimulus intensity level is at or below the threshold, the waveforms show no consistency and hence, a much lower correlation coefficient will be seen.

The ABR correlation analysis works in the following manner. Each waveform of 10 millisecond duration contains 256 points, only the last 128 points (those occurring in the interval 5 mS to 10 ms) are used in the correlation process. By windowing out the first 128 points, we have effectively insured the removal of any stimulus artifact that may exist at the start. With this windowing, the correlation method will focus on region containing the all important component V of the ABR since it occurs after the first 5 ms. A computer is used to obtain Pearson r correlation coefficient, PRCC, by comparing the 128 pairs of digital values for each of the stimulus intensity levels.

The PRCC at or below the threshold are approximately zero, indicating little similarity between waveforms. However, above the threshold, the PRCC are greater than zero. It is important to determine whether a high correlation could have occurred by chance. This can be achieved by correlating each of the two waveforms (obtained with same stimulus intensity level) with each of the two control waveforms (nonstimulus

waveform). Thus, four additional PRCCs are obtained. Now, if the subject responded to a test stimulus, then the PRCC of the stimulus vs. stimulus runs should be higher value than the PRCCs of the four stimulus vs. silent control runs.

The remaining step in this procedure is to determine, if the PRCC of the stimulus vs. stimulus run and the four PRCCs of the stimulus vs. control runs is significantly larger. This can be done using the Fisher Z' transformation [7]. The PRCCs are converted to a Z' score, which can then be related to a gaussian distribution yielding a probability value. The presence of an ABR at any stimulus intensity level is then defined as follows: the PRCC of the stimulus vs. stimulus run must be larger (.05 level of confidence) than three of the four PRCCs of the stimulus vs. silent control runs.

6.3.3 Demonstration of the Correlation Method

Having described the correlation method, it is fitting to show its performance. Two examples will be used to show its failure to work as a useful tool. The first example will show its ability to work as a useful tool. Conversely, the second example will show its failure to detect ABR, thus, the inconsistency of the algorithm.

Example 1: Fig 6.8 contains a set of waveforms of one ear of a patient at various different intensities. Two averaged waveforms are obtained for each stimulus intensity level. The last two wave rms are the silent control waveforms (those obtained without applying stimulus). Table I shows the results of correlation method. Upon visual evaluation by an experienced clinician it ascertained that the threshold for this patient's right ear is approximately 55 dB. This is confirmed by the results in Table I, the criterion

as described above is achieved for stimulus intensity levels as low as 55dB. That is, the PRCC of the two stimulus runs is greater than at least 3 of the 4 stimulus vs. control PRCCs for each stimulus level as low as 55dB. At stimulus level of 55 dB, this criterion fails and no ABR is considered to be present. This is consistent with visual evaluation.

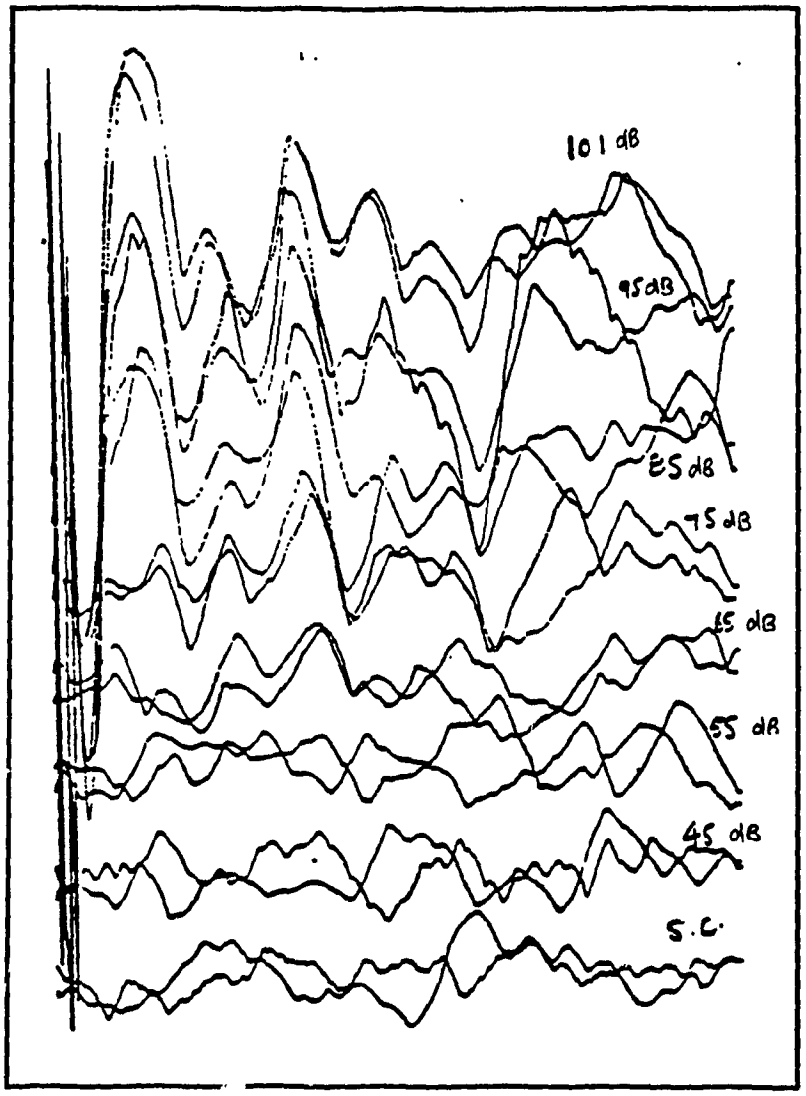

Figure 6.8. Set of ABR of one ear depicting the usefullness of Correlation technique.

Table 6.1 Results of Correlation Method Showing Its Effectivness

Hearing Sensation Level (dB)	Stimulus PRCC	Control PRCC	RESPONSE
91	.72	-.12	4/4
85	.59	-.12	4/4
75	.48	-.12	4/4
65	.31	-.12	4/4
55	.30	-.12	3/4
45	-.45	-.12	0/4
35	.06	-.12	2/4

Example 2: Fig 6.9 contains a set of waveforms of one ear of a patient at various stimulus. As above, two waveforms are obtained for each stimulus intensity level. The last two waveforms are the control runs. Table II shows the results of the correlation analysis. Visual inspection by an experienced clinician reveals the threshold to be around 55 dB. However, the correlation analysis suggests profound hearing loss. At every stimulus level, the analysis shows that the PRCC of the stimulus vs. stimulus run is not significantly larger than three of the four stimulus vs. control PRCCs. This contradiction in evaluation clearly shows the weakness of the correlation method. It evaluates a waveform as containing no ABR, when upon visual inspection the waveforms clearly contain ABRs. Thus the correlation algorithm yielded the incorrect evaluation. The lack of consistency of this method renders it a useless tool in an automated threshold tracking device, as this is the ultimate goal.

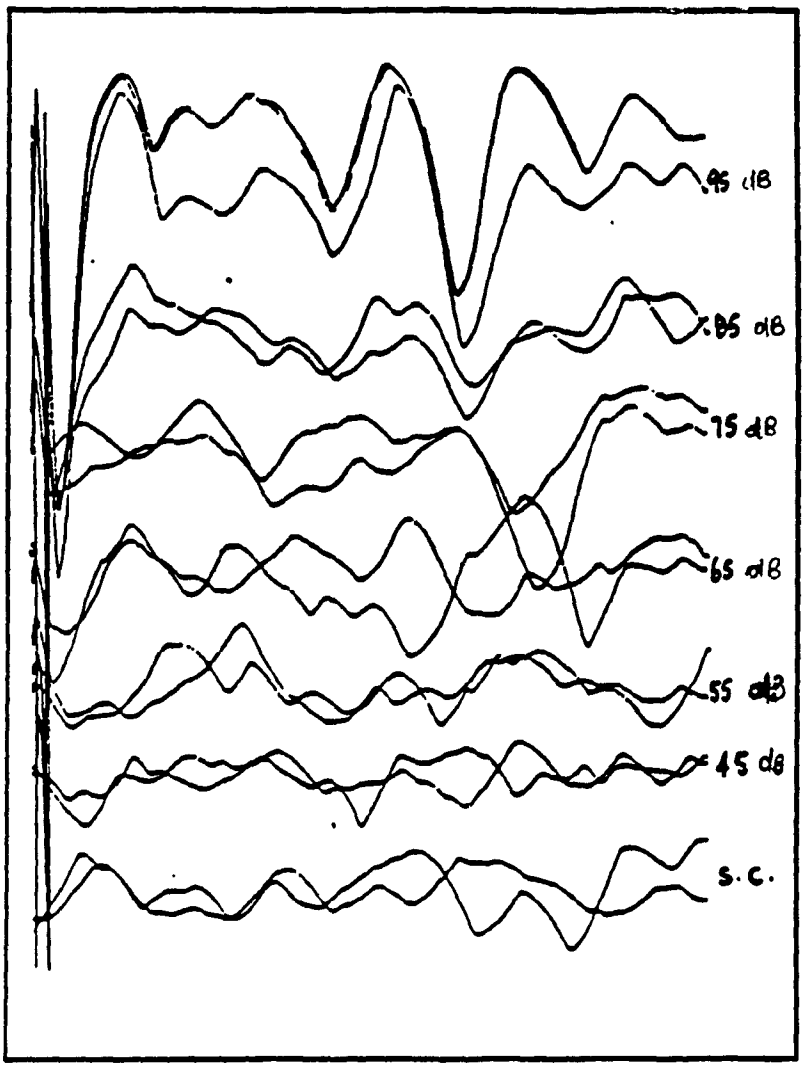

Figure 6.9. Set of ABR showing the inconsistency of the correlation method.

Table 6.2 Results of Correlation Method Showing Its Noneffectivness

Hearing Sensation Level (dB)	Stimulus PRCC	Control PRCC	RESPONSE
85	.53	-.02	0/4
75	-.39	-.02	0/4
65	.43	-.02	0/4
55	-.67	-.02	2/4
45	-.27	-.02	1/4
35	-.50	-.02	0/4

6.4 References

1. Carol Truxal, "Watching the brain at work," IEEE Spectrum March 1983.

2. Personal Communique on ABR analysis, H. Ilecki, Royal Victoria Hospital.

3. K. Hecox and R. Galambos, "Brainstem Auditory Evoked Responses in Human Infants and Adults," Arch, Otolaryngology, vol. 99, January 1974.

4. B.A. Weber and G.L. Fletcher, "A Computerized Scoring Procedure for Auditory Brainstem Response Audiometry," Ear Hearing vol 1, 1980.

5. K. H. Chiappa, K. J. Gladstone, and R. R. Young, "Brainstem Auditory Evoked Responses," Arch. Neurol., vol 36, Feb. 1979.

6. J.R. Boston, "Spectra of Auditory Brainstem Responses and Spontaneous EEG," IEEE Trans. on Biomedical Engineering, vol. BME-28, April 1981.

7. W. Woodsworth, S. Reisman and B. Fontaine, "The Detection of Auditory Evoked Responses Using a Matched Filter," IEEE Trans. on Biomedical Engineering, vol. BME-30, July 1983.

8. A. Papoulis, Probability, Random variables and Stochastic Processes, Mcgraw-Hill 1984.

9. M.B. Priestly, "Time Series Analysis," Academic Press Limited, 1988, England.

10. S.A Arnold, "Objective versus Visual Detection of the Auditroy Brainstem Response," Ear Hearing vol. 6, 1985.

11. R.J. Salvi et al, "Evoked Potentials: Computer Automated Threshold-Tracking Procedure Using an Objective Detection Criterion," Ear and Hearing vol 8, 1987.

12. P.K.H. Wong and R.G. Bickford, "Brainstem Auditory Evoked Potentials: the use of noise estimates," Electroencephlogr Clin Neurophysiol, 1980.

13. Royal Victoria Hospital, Department of Otolaryngology.

14. 13. G.E. Salamon, "Electric Response Audiometry (ERA) based on Rank Correlation," Audiology vol 13, 1974.

15. J.D. Wicke et al, "On-line Statistical analysis of averaged evoked potentials: application to evoked response audiometry (ERA)," Electroencepholgr Clin Neurophysiol vol 44, 1978.

16. A.L. Edwards, Statistical Methods for Behavioral Methods, Rinehart and Company, Inc, 1958.

17. D.M. Green and J.A. Swets, Signal Detection Theory and Psychophysics, Robert E. Krieger, 1974, New York.

Chapter 7

Results of ABR

In chapter 5, computer generated data was used to describe the relative performance of the algorithms considered, while chapter 6 described the area of ABR audiometry. This chapter presents the results of applying ABR data to these algorithms. Section one describes the ABR acquisition setup. Section two presents the results of ABR enhancement using CANF based algorithms and in section three, these enhanced ABRs are used to determine latencies of the ABRs by various cross correlation techniques.

7.1 ABR Acquisition System

The data acquisition system used for the recording of the ABRs was setup as described in Fig 6.4. Since single channel recording is done, clip electrodes are attached to the earlobes and a standard EEG disk electrode is attached to the forehead, which is used as the ground reference. The incoming EEG signal is amplified using a GRASS P15 preamplifier with a gain of 100. The output of the preamplifier is connected to a GRASS P511 amplifier with a gain of 1000. This output is then passed through a KRONHITE 3342 adjustable filter to eliminate spurious high frequency peaks from degrading the averaged evoked response. It is set up as a lowpass filter with a corner frequency of 5 kHz and a stopband roll-off of 48 dB/octave. A NICOLET MED 80 was used to average the individual responses. The data was stored on floppy disks and later transferred to an IBM PC environment. A 100 μS stimulus pulse was generated using a GRASS S88 stimulator. The intensity of this pulse was adjusted by an HP350D decade counter. The

stimulus pulse was delivered monaurally with a TELEPHONICS TDH-49 earphone with a 10 Ω input impedance.

The sampling rate for the analog-to-digital conversion was chosen to be 25 kHz and the duration of waveform is 10.24 mS. The individual ABRs were acquired at rate of 37 clicks/s. Throughout the experiment, the subject remained in a supine position in a sound attenuated chamber.

7.2 CANFs in ABR enhancement

Simulation results in section 1 of chapter 5 showed the performance difference of the three implementations considered, namely those of the CANF based CC, MC1 and MC2. We will now use these proposed configurations for the ABR signal enhancement. ABR enhancement can be achieved by subtracting the output of the proposed implementation from its input, as depicted in Fig 3.6. Each ABR used in these analysis is an average of 1024 individual responses for the same stimulus intensity. This averaging, a standard in ABR audiometry, is used to enhance the low SNR that is inherent in evoked potentials. All averaged ABRs must be preprocessed before using them in the considered algorithms.

First, the stimulus artifact must be eliminated while ensuring that the ABR stays intact. This is accomplished with an 8^{th} order digital butterworth bandpass filter with an upper and lower corner frequencies of 300 and 1500 Hz, respectively. Since the two frequencies of interest are well within the passband, this filter adequately removes the stimulus artifact while ensuring the signal integrity. Secondly, each waveform is

normalized by its maximum value. By this normalization, the maximum value of all waveforms is unity and thus, the rate of convergence (μ), in the adaptive procedure can remain constant for all waveforms. Eq 7.1, the CANF parameter update equation, can be used to explain this reasoning:

$$\theta(k) = \theta(k-1) + \mu K(n)\hat{n}(k) \tag{7.1}$$

It can be seen that the value of $\theta(k)$ is dependent on the power of the output noise, $\hat{n}(k)$. Thus, if the range of the amplitudes of the different waveforms varied, then it is obvious that the rate of convergence factor (μ) needs to be changed from ABR to ABR for CANF to convergence. By constraining the dynamic range of the amplitudes to a maximum value of unity, it is possible to use the same step size, μ, for all waveforms.

In our simulation study, it was observed that the total error in the notch point frequency was the lowest for the MC2 implementation; while the total error of CC and MC1 were approximately equivalent. However, in the spectral analysis of the transient, MC1 (and MC2) was found to be superior to CC. Analysis of real data ABR will be restricted to CC and MC1 implementations. The reason for this is as follows: since the approximate center frequencies of these two components are known, CANF coefficient corresponding to these frequencies can be used as the initial starting guess in the adaptive process. In MC2, the first section is used to obtain an estimate of the CANF parameters in the neighbourhood of the desired frequencies. Now, since we already have an *a priori* knowledge of this estimate, it is possible to eliminate this section. From this viewpoint, MC1 can be considered to be MC2. We, therefore, will use only CC and MC1 for the spectral analysis of ABR.

In the adaptive procedure of CC and MC1, after intensive experimenting, the debiasing coefficient, α, was set to .96. After, having estimated the coefficient corresponding to the notch point, the debiasing coefficient was increased to .97 in the CNF section of each stage (the fixed filter section of MC1). By this increase in the debiasing coefficient, it is possible to achieve a slightly narrower notch bandwidth, while maintaining a small transient time of CNF. Narrow notch bandwidth are required for closely spaced sinusoids of this application. The rate of convergence or the step size (μ) was set to 0.3 for all adaptations.

7.2.1 ABR Spectral Analysis

Fig 7.1a,b,c,d show the Power Spectral Densities, PSDs, of (1) ABR (before the removal of the artifact); (2) ABR (after the removal of artifact); (3) enhanced ABR using a two stage implementation of CC; 4) enhanced ABR using a two stage implementation of MC1. In the presence of noise, from Fig 7.1a,b, it is observed that the PSD based on FFT alone is not able to resolve the two components clearly. Results of MC1 and CC in Fig 7.1c,d, respectively, clearly indicate the resolution of the two components. As well, there is a marked improvement in the PSD of ABRs using the MC1 over that of CC. The point to remember is that it is inherently possible for both stages of CC implementation to converge to the same frequency. This would fail to resolve the two ABR frequency components, hence the failure of CC. Based on the ABR corresponding to Fig 7.1a, this subject was evaluated to be profoundly deaf in the ear considered at this stimulus intensity. But, the results of Fig 7.1c,d particularly Fig 7.1d, suggest clear existence of the two important components around 500 and 1000 Hz.

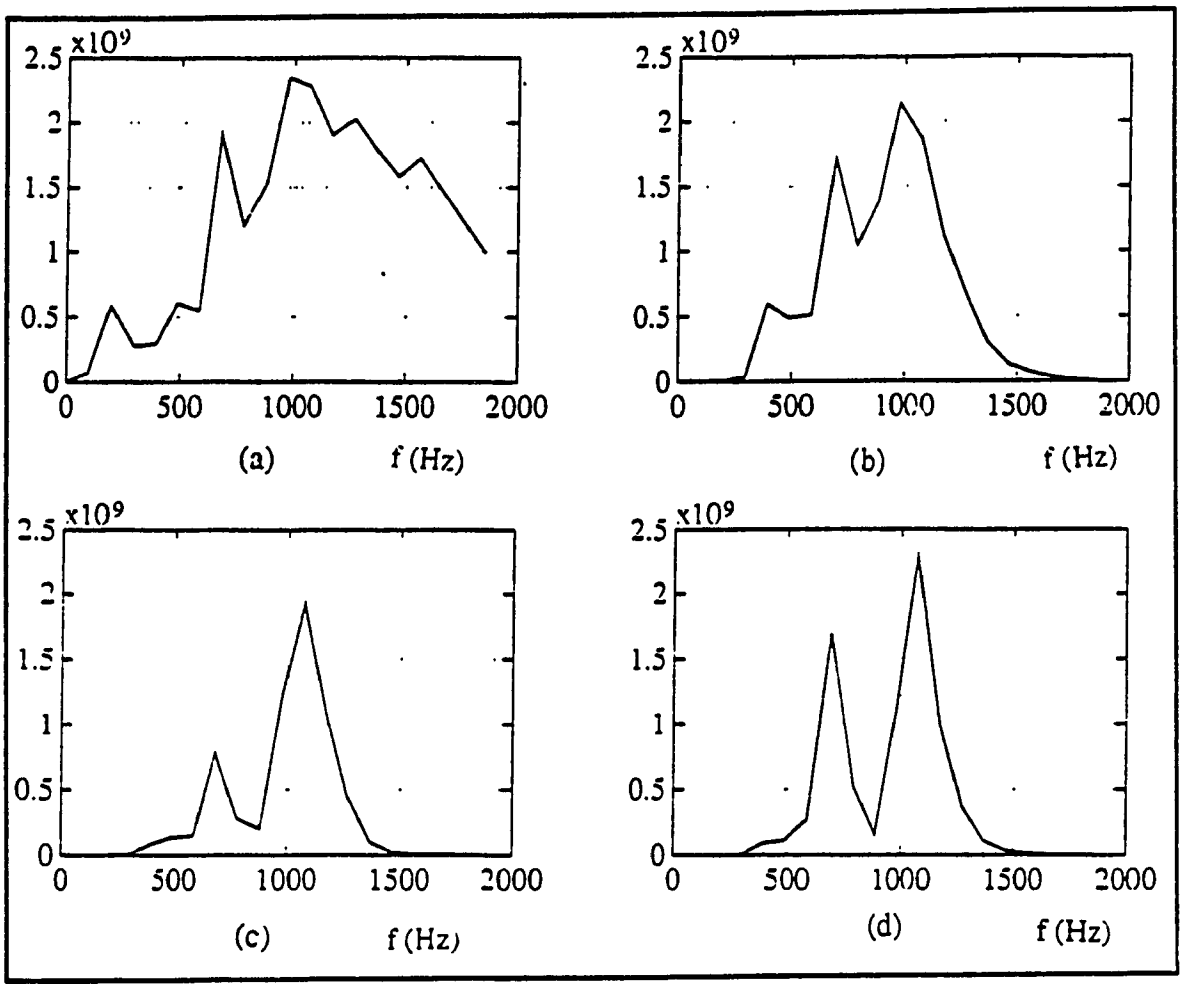

Figure 7.1. PSDs of (a) ABR (before artifact removal); (b) ABR after artifact removal; (c) ABR enhanced by CC; (d) ABR enhanced by MC1.

Observing the corresponding ABRs in Fig 7.2, it is likely that a reversal in evaluation by an audiologist is warranted. Another important observation from these results is the latency (time of occurrence) of the fifth component, labelled V in Fig 7.2. Before using MC1 and CC to enhance the ABR, it is observed that the latency of this fifth component is around 5.7 mS, while the enhanced ABR shows this to be about 5.1 mS, a more acceptable value.

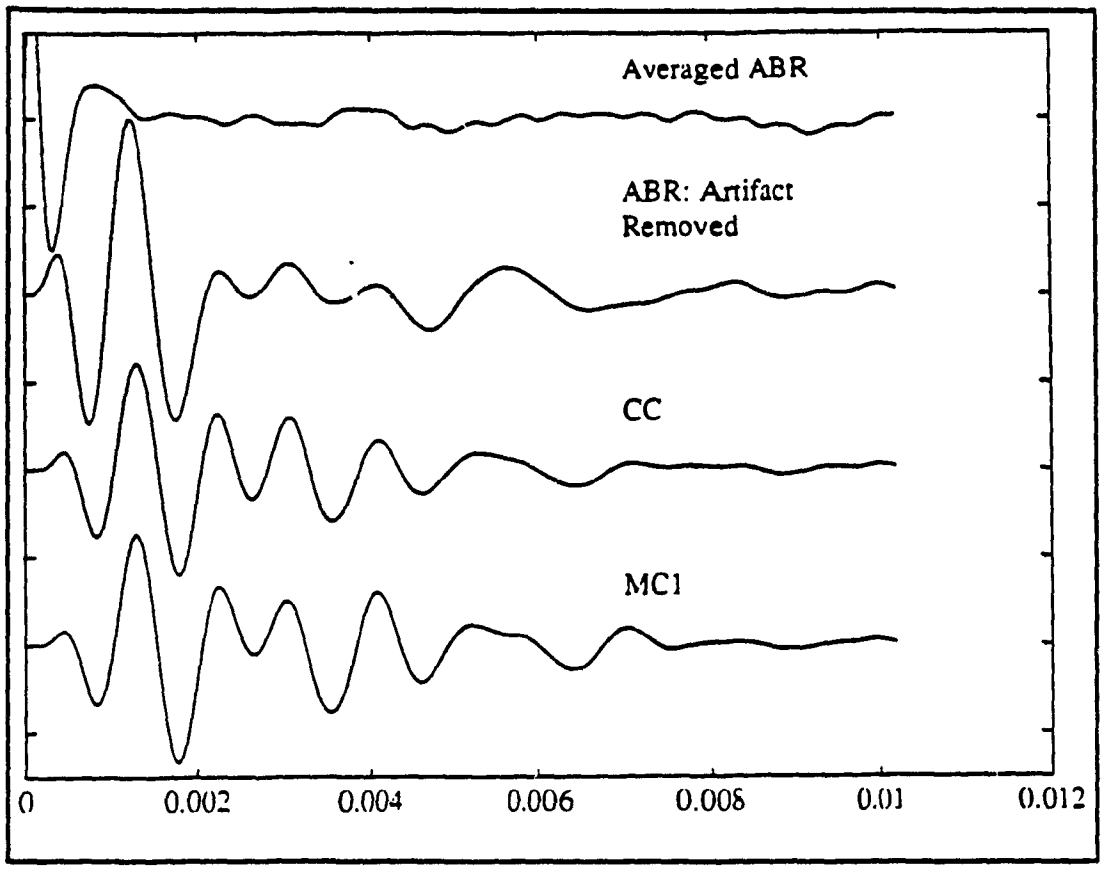

Figure 7.2. ABRs corresponding to the PSDs of Fig 7.1

7.2.2 Threshold Detection

From the above results, it is apparent that the MC1 implementation of CANF provides better results than the CC implementation. In the next two examples, we'll use two sets of ABRs that have been processed by MC1 and CC, to determine hearing thresholds. This hearing threshold is compared to the threshold determined by the existing algorithm in use, namely the correlation method described in chapter 6.

Example 1. Fig 7.3a,b,c shows the recording of ABR before filtering (i.e. as acquired), ABRs processed via MC1 and CC implementations, respectively. The numbers in parenthesis on the right indicates the stimulus intensity in dBs and the solid and dashed lines indicate two different averaged waveforms acquired for the same intensity level. For each intensity level, two ABRs are acquired, which are shown in pairs (solid-dashed line). The last pair in each set is the silent control, i.e. response obtained without applying a stimulus.

Examining the original (unfiltered) waveforms of Fig 7.3a, it was ascertained by an audiologist that they showed no response. It was concluded that this subject is severely hearing impaired, however, it may still be possible for the high levels of the physical activity to influence the outcome. Table 7.1 shows the results of the correlation method. From these results, it is impossible to determine a threshold level. The criterion for the detection of ABR (described in chapter 6) by the correlation method is not met consistently enough, to be able to make a threshold decision. Fig 7.3c contains the ABRs enhanced by the CC implementation. By evaluating these waveforms, it can be roughly seen that there exists a response for stimulus intensity as low as 85 dB, however, it is likely that the threshold is even lower. Fig 7.3b contains the ABRs enhanced by the MC1 implementation. Response for stimulus level as low as 75 dBs can be detected easily. As with the results of CC, it is likely that a trained clinician may accept the waveforms due to even lower stimulus intensities as containing a response. In this latter set, the results of MC1, various components of each ABR appear to be defined more clearly than the CC results. Thus, other parameters of the ABR audiometry (such as latency) can be much more readily detected.

Example 2. As in the last example, Fig 7.4a,b,c contain the recording of a set of ABRs at various stimulus intensities, ABRs processed by MC1 and those processed via CC, respectively. In this situation, based on the visual inspection of the waveforms of Fig 7.4a, by an audiologist, profound hearing loss of this ear is concluded. Correlation method yielded a threshold as low as 95 dBs, inconsistent with that of the visual inspection. The waveform in Fig 7.4b,c both yield a threshold as low as 85 dBs, with the individual peaks in Fig 7.4b being much more defined. Should the test have continued for lower intensities, it may be possible (even likely) that the hearing threshold becomes lower.

Table 7.1 Correlation Method Analysis of ABRs of Fig 7.3a

Hearing Sensation Level (dB)	Stimulus PRCC	Control PRCC	RESPONSE
101	.12	-.23	1/4
95	.15	-.23	1/4
85	.54	-.02	4/4
75	.43	-.23	2/4
65	.13	-.23	0/4
55	.36	-.23	4/4
45	.44	-.23	3/4

Table 7.2 Correlation Method Analysis of ABR of Fig 7.4a

Hearing Sensation Level (dB)	Stimulus PRCC	Control PRCC	RESPONSE
101	.71	.37	4/4
101	.85	.37	4/4
101	.32	.37	3/4
95	.17	.37	2/4
95	.64	.37	4/4
85	.33	.37	0/4

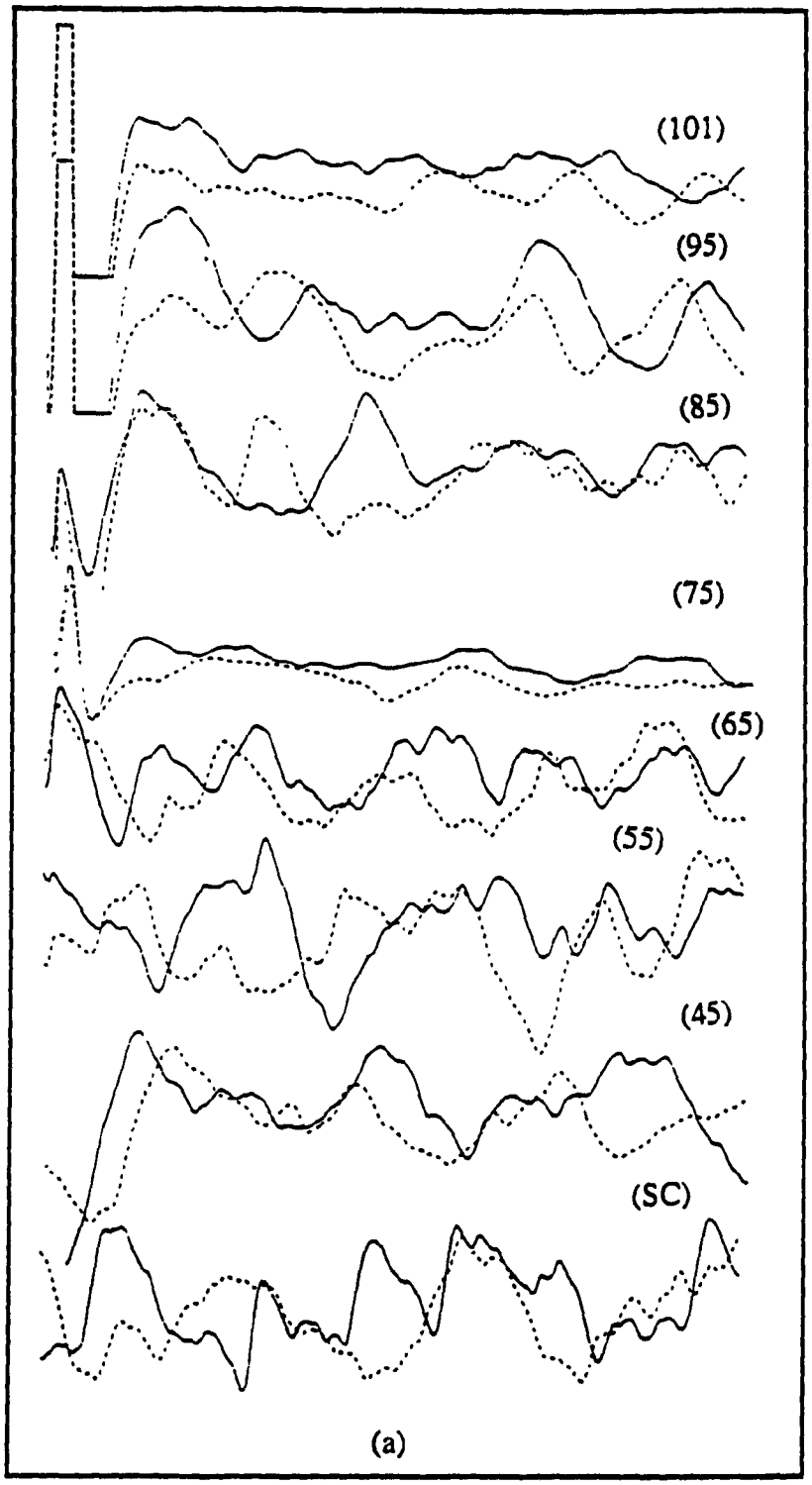

Figure 7.3. (a) Set of averaged ABRs for one ear of one subject; (b) ABRs filtered by MC1; (c) ABRs filtered by CC.

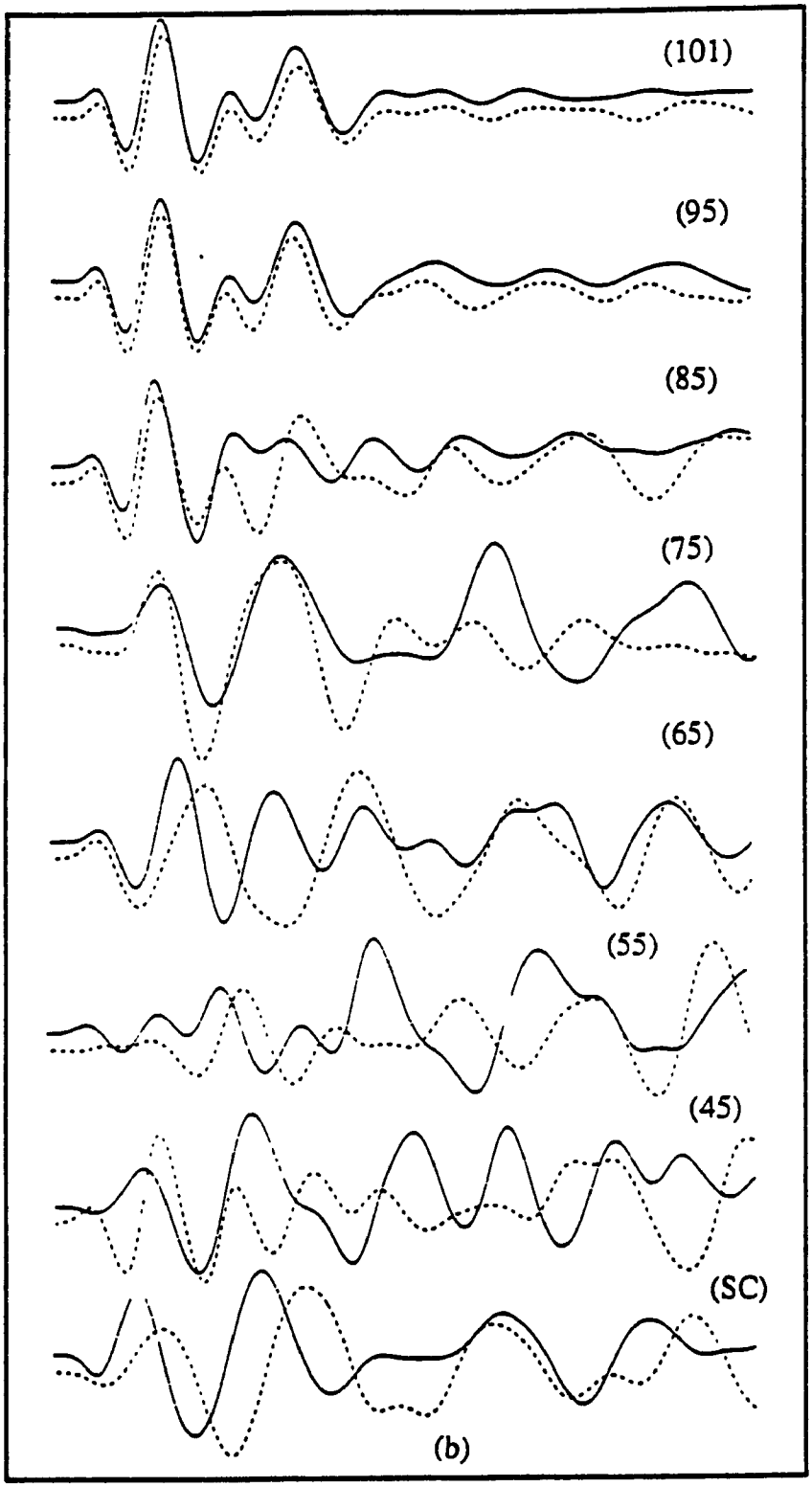

Figure 7.3 (con't)

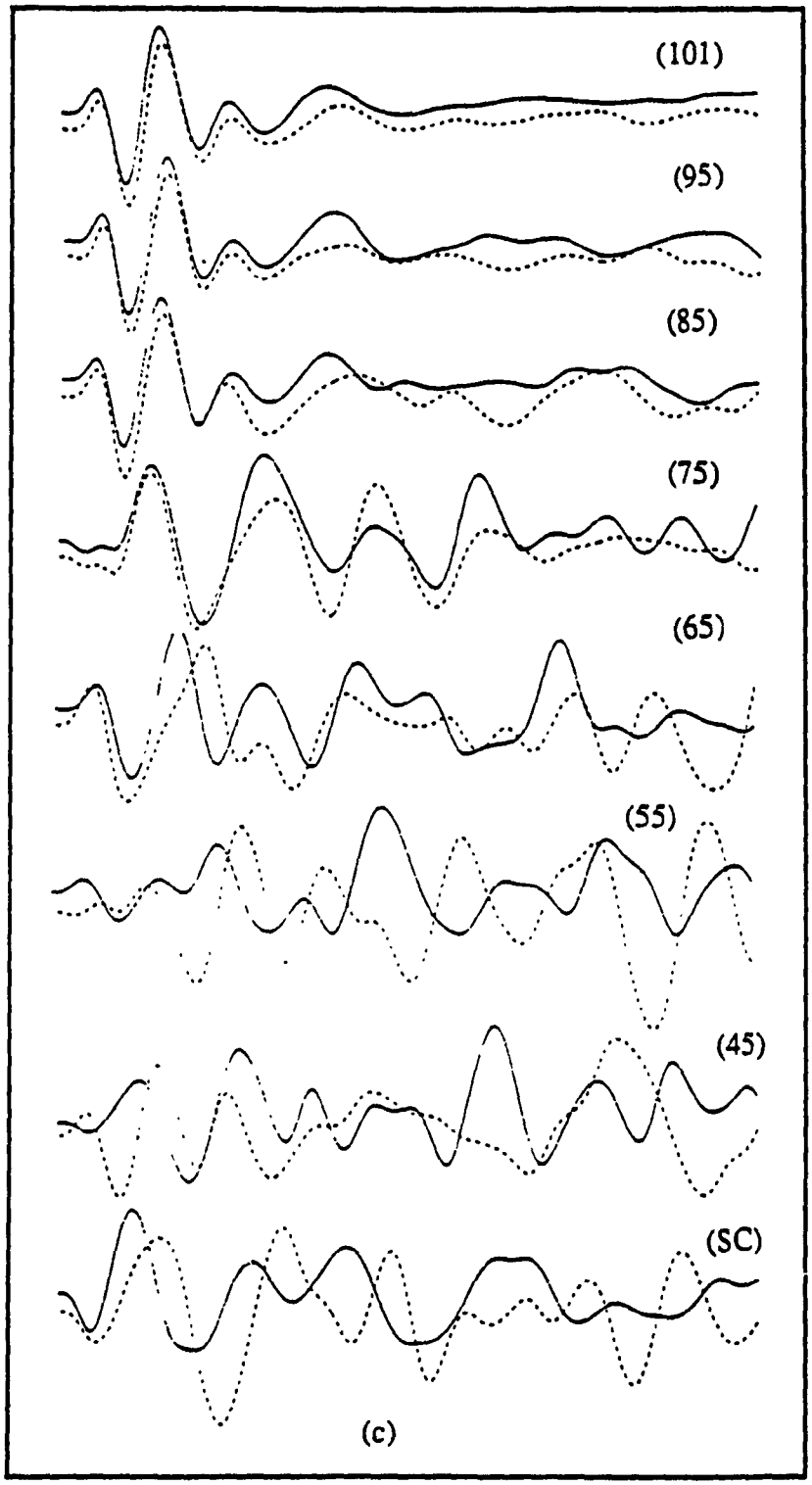

Figure 7.3 (con't)

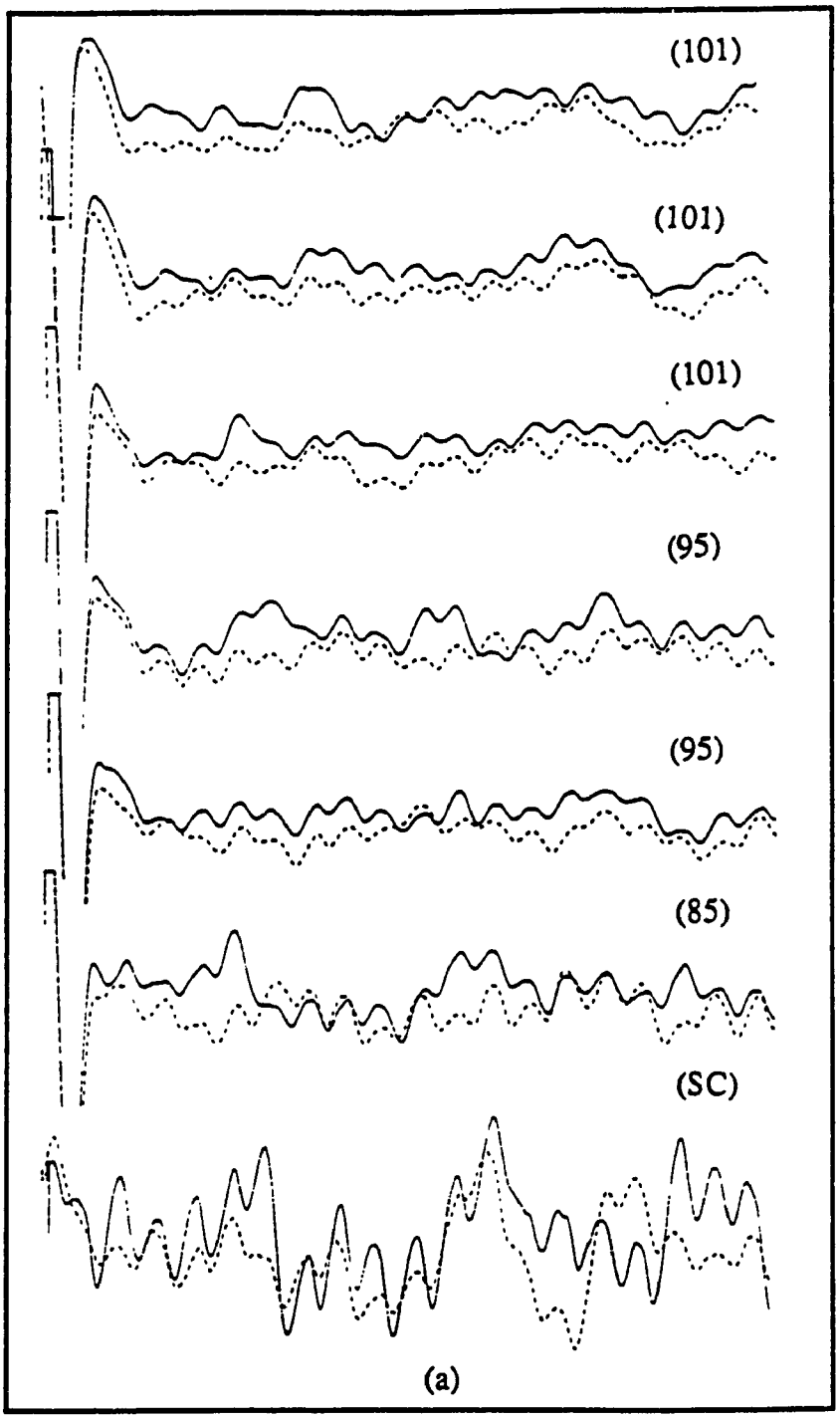

Figure 7.4. (a) Set of averaged ABRs for one ear of a subject; (b) ABRs filtered by MC1; (c) ABRs filtered by CC.

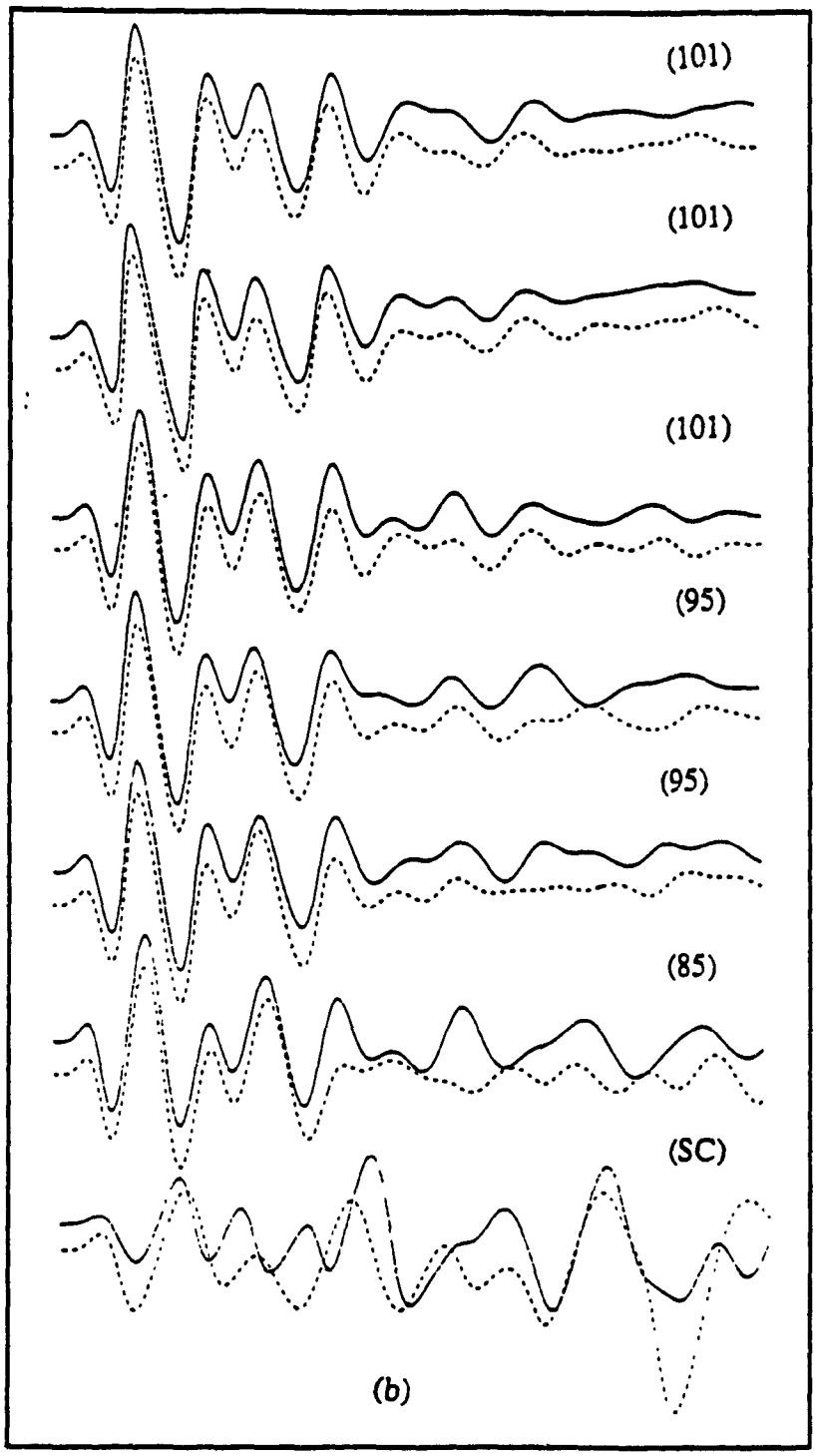

Figure 7.4 (con't)

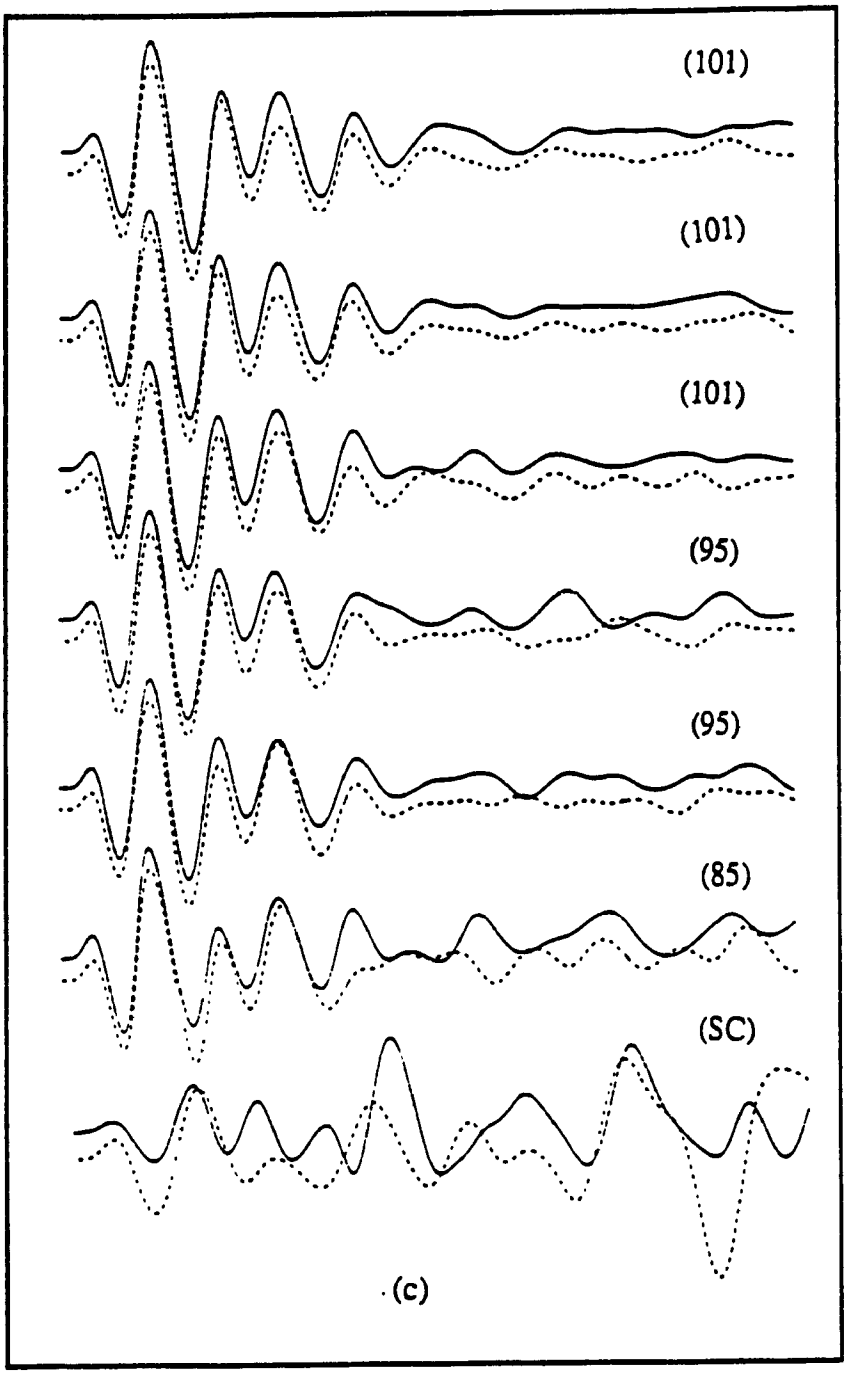

Figure 7.4 (con't)

From the above two examples, it is obvious that the correlation method and the visual evaluation of the noisy ABRs can at times lead to incorrect conclusions. Processing these ABRs with the CC and MC1 implementation leads to cleaner ABRs and thus the improved visual evaluation. *It would be interesting to see the results of the correlation method for detecting the presence of ABRs, using the ABRs processed with MC1 implemetation.* The reliability of visual inspection is increased. Parameters, such as latencies, can be elicited much more readily from the CANF filtered ABRs than otherwise. MC1 and CC implementations, therefore, can form a useful tool in the ABR audiometry.

7.3 Cross Correlation for ABR Latency Detection

In this section, results of using generalized cross correlation techniques to estimate the group delay of the successive waveforms in the series of waveforms of one ear of one patient, relative to the waveforms obtained with the highest intensity stimulus are presented. The two waveforms obtained with the highest stimulus intensity are used as the reference. In this manner, for each intensity level, there will be four latency values determined with respect to the waveform obtained at the highest intensity level. We may then take the average of the four values as the latency for that particular stimulus intensity. Before the cross correlation technique is applied, the amplitudes of all waveforms are scaled such that the levels are equivalent to the reference signal (that of the highest stimulus intensity) and the mean is removed from each ABR.

In determining the overall latencies of ABRs, two GCC were realized, those being the Approximate Maximum Likelihood (AML) and the basic Cross Correlator (CC). All the necessary spectrums are estimated directly from the observed ABRs using FFTs. Because the recorded waveforms are extremely noisy, ABRs filtered via MC1 implementation will be used in the determining of latencies of waveforms. For the set of ABRs used in this analysis, it was determined by an audiologist that the threshold is around 65 dBs. Latencies, therefore, are only determined for ABRs as low as 65 dB stimulus intensity. Fig 7.5 shows the Latency-Intensity curves achieved by the AML and CC GCCs. It is seen that, as the intensity is decreased, the latency increases. This is in agreement with the expected behaviour of the ABRs. A trained clinician can now use

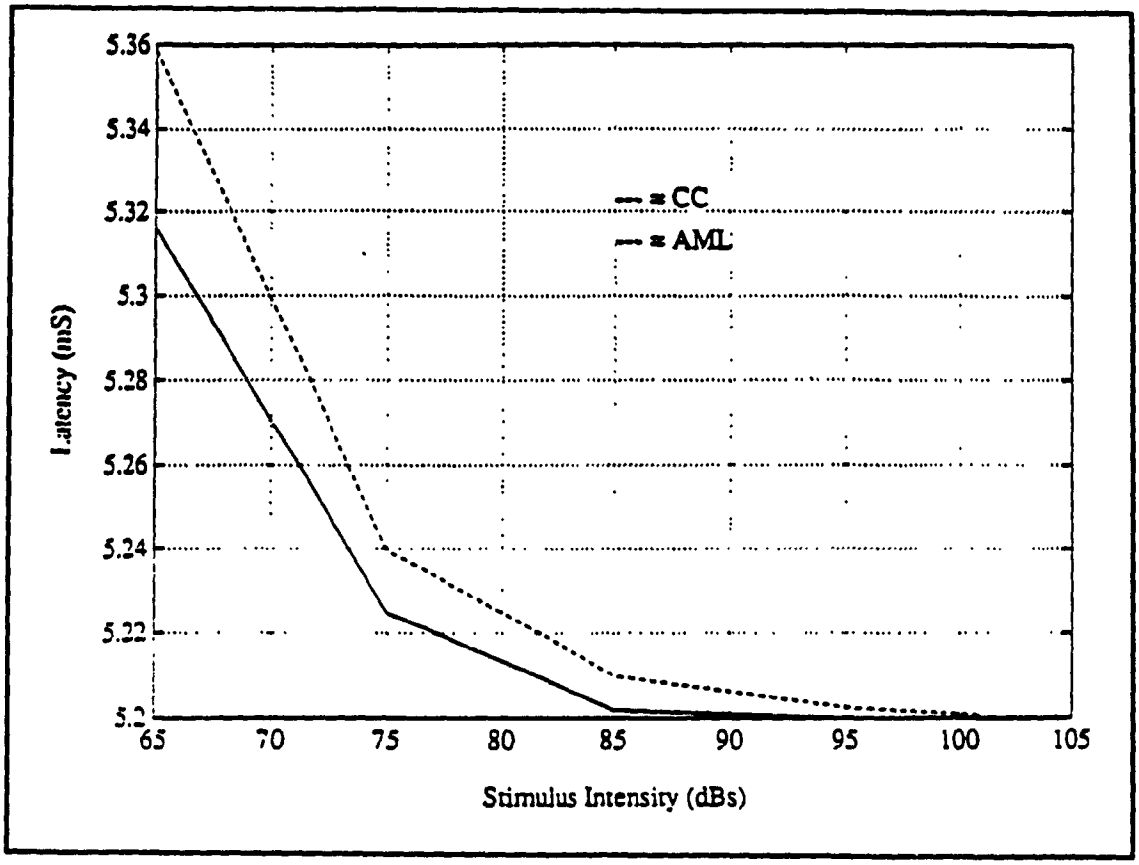

Figure 7.5. Latency-Intensity relationship as determined by the AML and CC GCCs.

these LI curves to compare against the accepted standards. Should the result deviate out of the norm, certain diagnosis can be deduced. In Fig 7.5, latencies are biased by 5.2 mS, as this is the approximate latency of the fifth component of the ABR obtained by applying the highest stimulus intensity level. Now, for lower stimulus intensities, the latencies will be relative to this value. By this shift in the latencies, it is easier for a clinician to compare against the available standard latencies of the fifth component.

Chapter 8

Conclusion

Auditory evoked Brainstem Responses (ABRs) are used in determining certain aspects of a person's hearing capacity. Due to their non-invasive nature, they are highly masked by the background EEG. At times, particularly near the hearing threshold, averaging does not help the ABR to stand out. Plethora of signal processing algorithms have been applied to enhance the ABRs, but, none have triumphed.

In this thesis, constrained adaptive IIR filters (known as CANFs in literature) were applied to enhance ABRs. It was shown by the error surface analysis how to circumvent the problem of CANFs (direct form implementation) converging to a local minimum. By using a cascade connection of second order CANFs, local minima were shown to be eliminated. Enhanced performance of the Conventional Cascade (CC) implementation was shown via simulations.

By decoupling the estimation of the CANF parameters and the filtering operation in the Modified Cascade One (MC1) and Modified Cascade Two (MC2), it was shown via simulations that the overall performance increased significantly. Since, all three implementations are equivalent for large data, these improved results of using MC1 and MC2 configurations, are with respect to short data only.

Having realized the performance improvement, these implementations of the CANFs were applied to real ABR data. In the previous chapter, it was shown how the CANF based filtering provides for noise reduced ABRs. These noise free waveforms were then used to determine hearing threshold of two patients. It was clearly shown that there

exists a wide disparity in the threshold determined before and after the filtering. The presence of a response was clear in certain cases, where the unfiltered waveforms suggested its absence. The patient of Example 2 in section 2 of chapter 7 was clinically evaluated to be deaf for the ear considered, however, using the filtered waveform in the evaluation the extent of his deafness need be reconsidered. Although, the time-domain results of CC and MC1 both indicated similar performance in terms of the presence of the ABR, the results of MC1 revealed the various components of ABRs much more clearly. Using ABR spectra to decide the presence of a response, one can conclude that MC1 is superior.

In the last section of chapter 7, the Approximate Maximum Likelihood and basic Cross Correlators were used to determine the latencies of ABRs, relative to the stimulus intensity levels. From these results, it appears that the difference between the two implementations is negligible (AML being slightly better, expected result according to theory and simulations) as far as the clinical evaluation is concerned. The latencies are somewhat smaller than expected. Before either of these two algorithms can be accepted or dismissed as compared to the visual evaluation of latencies, extensive study using a large number of ABRs must be conducted. For the patient in section 7.3, the AML and CC do not suggest latency estimation using GCCs to be better than the visual evaluation.

Suggested Future Work

The event-related waveforms used in this work were all averaged waveforms. The repeatability of these waveforms was assumed in the averaging process. That is, the evoked response (desired ABR) may not be similar from trial to trial. It should be realized that any phase changes or slight frequency drifts from trial to trial will not distort the averaged waveform, if the length of the observation interval leads to infinity. However, if we deal with finite observation interval, particularly with very short one, the averaged response will invariably be distorted. Thus, a possible alternative for detection of ABR may be to process N individual evoked responses (before averaging) using the above CANF methods to determine the approximate frequencies of the two component. This can be accomplished by transforming the CANF coefficient to the appropriate frequency values. The consistency of the frequency estimates for the N waveforms due to the same stimulus can then be used by some automated detection criteria to make a decision as to the presence or absence of an ABR. If the presence of an ABR is detected, then knowing that a response exists, these N waveform can be averaged to yield the desired signal for further processing to yield other clinically important ABR parameters such as latencies and the various ABR components.

APPENDIX A

The constant C in Eq 2.21 is given by

$$C = \frac{1}{\sqrt[N]{2\Pi |Q_R|}} = \prod_{k=1}^{N} \frac{1}{\sqrt{2\pi |Q_R(k)|}} , \qquad \text{A1}$$

where

$$Q_R(k) = \begin{bmatrix} S_{xx}(k) & S_{xy}(k) \\ S_{xy}^*(k) & S_{yy}(k) \end{bmatrix}$$

Hence,

$$|Q_R| = S_{xx}(k)\, S_{yy}(k) - S_{xy}(k)\, S_{xy}^*(k) \qquad \text{A2}$$

Suppressing the discrete frequency argument (k) and under the assumption $S_{xy}(f) = 0$, Eq A2 can be written as

$$|Q_R| = (S_{ss} + S_{n_1 n_1})(S_{ss} + S_{n_2 n_2}) - \alpha^2 S_{ss}^2 \qquad \text{A3}$$

Substituting A3 in A1 yields the constant C that is independent if the delay, D.

APPENDIX B

Eq 2.23 can be written as the sum of two terms as

$$J = J_2 + J_3 \qquad \text{B1}$$

J_2 and J_3 will be derived and the independence of J_2 on the delay, D will be shown. The inverse of the Power Spectral Density matrix, $Q_R(f)$ of Eq 2.23 can be written in terms of the Magnitude Coherence Function as

$$Q_R^{-1}(f) = \frac{1}{1 - |\gamma_{xy}(f)|^2} \begin{bmatrix} \frac{1}{S_{yy}(f)} & \frac{-Sxy(f)}{S_{xx}(f)S_{yy}(f)} \\ \frac{-S_{xy}^*(f)}{S_{xx}(f)S_{yy}(f)} & \frac{1}{S_{xx}(f)} \end{bmatrix} \qquad \text{B2}$$

where

$$|\gamma_{xy}(f)|^2 = \frac{S_{xy}^2(f)}{S_{xx}(f)S_{yy}(f)}$$

which will exist provided $|\gamma_{xy}(f)|^2 \neq 1$. Substituing B2 in the Eq 2.23 and performing some mathematical manipulations yields

$$J_2 = \int_{-\infty}^{\infty} \frac{1}{1 - |\gamma_{xy}(f)|^2} \left[\frac{|X(f)|^2}{S_{xx}(f)} + \frac{|Y(f)|^2}{S_{yy}(f)} \right] df \qquad \text{B3a}$$

and

$$J_3 = -\int_{-\infty}^{\infty} \frac{[A(f) + A^*(f)]}{[1 - |\gamma_{xy}(f)|^2]}\, df \qquad \text{B3b}$$

where

$$A(f) = X(f)Y^*(f)\frac{\alpha S_{ss}(f)e^{j2\pi fD}}{S_{xx}(f)S_{yy}(f)}$$

For x(t) and y(t) real signals, letting $S_{xy}(f) = \frac{X(f)Y^*(f)}{T}$ and under the assumptions

$S_{n_1 n_2}(f)=0$, J_3 can be rewritten as

$$J_3 = -2T\int_{-\infty}^{\infty} S_{xy}(f)\frac{1}{|S_{xy}(f)|}\frac{|\gamma_{xy}(f)|^2}{[1 - |\gamma_{xy}(f)|^2]}e^{j2\pi fD}\, df \qquad \text{B4}$$

From Eq B3a, it is easliy seen that J_2 is independent of D and J_3 is as required.