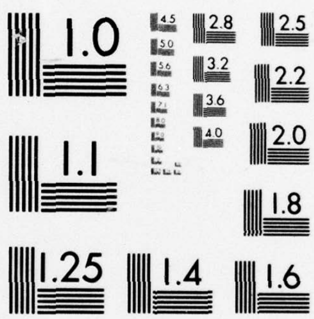
AD-A053 852 NAVAL POSTGRADUATE SCHOOL MONTEREY CALIF F/G 9/3
AN ADAPTIVE RECURSIVE FILTER.(U)
DEC 77 S KO
UNCLASSIFIED NL

1 OF 2
AD
A053852



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

NAVAL POSTGRADUATE SCHOOL

Monterey, California

THESIS

AN ADAPTIVE RECURSIVE FILTER

by

Soon-Ju Ko

December 1977

Thesis Advisor: Sydney R. Parker

Approved for public release; distribution unlimited

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle) An Adaptive Recursive Filter	5. TYPE OF REPORT & PERIOD COVERED Electrical Engineer December, 1977
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Soon-Ju Ko	8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940	12. REPORT DATE December 1977
	13. NUMBER OF PAGES 109
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Naval Postgraduate School Monterey, California 93940	15. SECURITY CLASS. (of this report) Unclassified
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

An adaptive recursive digital filter is presented in which feedback and feedforward gains are adjusted adaptively to minimize a least square performance function on a sliding window averaging process. A two-dimensional version of the adaptive filter is developed and its performance compared with the optimal Wiener filter. The filter is shown to be effective in separating three diagonal trajectory streaks from a background of correlated

DD FORM 1473, 1 JAN 73 EDITION OF 1 NOV 65 IS OBSOLETE

S/N 0102-014-6601 |

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered) noise added to white noise. Although the recursive adaptive filter approaches the optimal Wiener filter in performance, it does not require a priori statistical knowledge as does the Wiener filter to which it is compared. The results indicate that the recursive adaptive filter "learns" the statistics and adapts.

Approved for public release; distribution unlimited

AN ADAPTIVE RECURSIVE FILTER

by

Soon-Ju Ko
Lieutenant, Republic of Korea Navy
B.S.E.E., Seoul National University, 1975

Submitted in partial fulfillment of the
requirements for the degree of

ELECTRICAL ENGINEER

from the
NAVAL POSTGRADUATE SCHOOL
December 1977

Author Soonjuko

Approved: Sydney R. Parker
Thesis Advisor

Donald E. Kirk
Second Reader

Donald E. Kirk
Chairman, Department of Electrical Engineering

George J. Haltiner
Dean of Science and Engineering

ABSTRACT

An adaptive recursive digital filter is presented in which feedback and feedforward gains are adjusted adaptively to minimize a least square performance function on a sliding window averaging process. A two-dimensional version of the adaptive filter is developed and its performance compared with the optimal Wiener filter. The filter is shown to be effective in separating three diagonal trajectory streaks from a background of correlated noise added to white noise. Although the recursive adaptive filter approaches the optimal Wiener filter in performance, it does not require a priori statistical knowledge as does the Wiener filter to which it is compared. The results indicate that the recursive adaptive filter "learns" the statistics and adapts.

TABLE OF CONTENTS

I. INTRODUCTION -- 7

II. LINEAR STOCHASTIC PROCESSES ------------------------10

A. INTRODUCTION ---------------------------------------10

B. DEFINITION AND PROPERTIES OF LINEAR STOCHASTIC PROCESSES ------------------------------11

C. MODELING OF LINEAR STOCHASTIC PROCESS ------------22

1. Introduction ---------------------------------22

2. Method of Least Squares ---------------------25

3. Method of Filter Response -------------------34

III. ADAPTIVE FILTERS ------------------------------------43

A. NONRECURSIVE FILTER ----------------------------43

1. Introduction ---------------------------------43

2. Performance Surface, Gradient and Wiener Solution ------------------------------45

3. LMS Algorithm for One-Dimensional Filter -----48

4. Two-Dimensional Adaptive Filter -------------52

a. Adaptive Filter Structure ---------------52

b. Wiener Solution -------------------------54

c. LMS Algorithm ---------------------------56

B. RECURSIVE FILTER --------------------------------58

1. Introduction ---------------------------------58

2. One-Dimensional Adaptive Recursive Filter -- 59
a. Structure ---------------------------- 59
b. Problem in Wiener Solution ------------ 61
c. LMS Algorithm ------------------------ 64
3. Two-Dimensional Recursive Adaptive Filter
IV. ADAPTIVE NOISE CANCELLER ------------------------ 77
A. THE CONCEPT OF ADAPTIVE NOISE CANCELLING ------- 77
B. NOISE CANCELLING WITHOUT AN EXTERNAL REFERENCE INPUT ------------------------------- 82
V. EXPERIMENT AND RESULTS --------------------------- 85
V. CONCLUSIONS -- 105
BIBLIOGRAPHY --- 108
INITIAL DISTRIBUTION LIST ---------------------------- 109

I. INTRODUCTION

The term "filter" is often applied to any device or system that processes incoming signals or other data in such a way as to eliminate noise, to smooth signals, to identify each signal as belonging to a particular class, or to predict the input signal from instant to instant. There is an abundance of literature covering the theories involved under the headings of estimation, identification, modeling, prediction, etc. The usual method of estimating a signal corrupted by noise is to pass it through a filter that tends to suppress the noise while leaving the signal relatively unchanged. The design of such filters falls in the domain of optimal filtering, which originated with the pioneering work of Wiener [8] and was extended and enhanced by the work of Kalman-Bucy [9] and others.

Filters used for the above purpose can be fixed or adaptive. The design of a fixed filter is based on a priori knowledge of both signal and noise statistics. On the other hand, adaptive filters have the ability to adjust their own parameters automatically, and their design requires little or no a priori knowledge of signal or noise characteristics. This work presents an approach to signal filtering using an adaptive filter that is in some sense self-designing. The adaptive filter described here bases its own "design" (its adjustment of internal parameters) upon estimated (measured) statistical characteristics of input and output signals.

The statistics are not measured explicitly and then used to design the filter; rather, the filter design is accomplished in a single process by a recursive algorithm that automatically updates the system parameters with the arrival of each new data sample. It is assumed that the input and the output at the sampling instants are the only measurable quantities of the system. It is also assumed that the unknown filter coefficients (parameters) to be designed enter linearly in the difference equations which describes the self-designing process.

The steepest descent method is employed in which the prevailing filter parameter vectors are perturbed at each iteration in a manner so as to decrease a prescribed functional (error criterion or cost function) to be minimized. The steepest descent method is one of the well known gradient based algorithms.

For the case where the functional being minimized is the mean square error, where the error is the difference between filter output signal and the desired signal, the filter is called the least mean square filter (LMS filter). Various adaptive algorithms are currently available depending upon the cost function and the method used to minimize cost function.

The popularly used performance criteria are the least mean square criterion, the maximum likelihood ratio (MLR) criterion, and the maximum signal to noise ratio (SNR) criterion. Here the LMS criterion only is studied and the steepest descent method is employed. Inevitable errors in the estimation of the statistics prevent the adaptive filter from delivering optimal performance since the adaptive filter is not based on the a priori knowledge of statistics. In Chapter II, the concept of linear stochastic processes is reviewed as a preliminary study for this thesis, and the modeling of stochastic processes is studied. These can be considered as background material for the following chapters.

In Chapter III, the concept of adaptive filters is introduced and the structure of the signal and the mathematical model of the processor is delivered. The algorithm for the nonrecursive adaptive filter by Widrow [1] is reviewed and the new algorithm for the recursive adaptive filter is developed as is the two-dimensional adaptive filtering process.

In Chapter IV, the adaptive noise cancelling concept is analyzed rather qualitatively and its application to the special case in which no desired signal is available is analyzed. In Chapter V an experiment is performed through computer simulation to check the feasibility of algorithms developed in the previous chapter and a comparison with the optimal Wiener solution is made. In Chapter VI, the conclusions are presented together with a summary of the experimental results and suggestions for further research.

II. LINEAR STOCHASTIC PROCESSES

A. INTRODUCTION

The problem of defining a random process is of considerable importance in the analysis of systems subject to noise disturbance. Often a partial definition of the process will suffice as in the case of linear least mean square error filtering, where only a knowledge of the correlation function is required. For other problems, such as those involving nonlinear filtering, more complete information will generally be needed. A complete description of a random process requires a knowledge of the distribution functions of all orders. But in practice few processes apart from the normal and Markov are defined in this manner. For the purpose of analysis, a model to generate the random process is desirable and for a model to give a complete description of the process, the distribution functions should be derivable from the model. While both continuous and discrete-time linear process may be defined, only the discrete-time case will be considered here. The discrete-time linear process can be considered to be the result of the digital filtering of a sequence of independent and identically distributed (IID) random variables.

The linear processes are important since they are inherently simple in terms of physical considerations and form a class which includes many discrete time normal random processes. In the following section the definition and properties of the linear processes are summarized.

B. DEFINITION AND PROPERTIES OF LINEAR STOCHASTIC PROCESSES

It has been found useful in the theory of stochastic processes to divide stochastic processes into two broad classes: stationary and nonstationary. Intuitively, a stationary process is one whose distribution remains the same as time progresses, because the random mechanism producing the process is not changing as time progresses. A nonstationary process is one which is not stationary.

Let $\{x(i),\ i \varepsilon T\}$ be a stochastic process with finite second moments. Its mean value sequence, denoted by $m(i)$, is defined for all i in T by

$$m(i) = E\,[x(i)] \qquad (2\text{-}1)$$

and its covariance kernel, denoted by $K\,(j.i)$, is defined for all j and i in T by

$$K\,(j.i) = \mathrm{Cov}\,[\,x\,(j),\ x\,(i)] \qquad (2\text{-}2)$$

An index set T is said to be a linear index set if it has the property that the sum $i+j$ of any two numbers i and j of T also belongs to T. Examples of such index sets are $T = \{1,2,\ldots\}$, $T = \{o,\pm 1, \pm 2, \ldots\}$, $T=\{i;i \geq o\}$ and $T=\{i:-\infty < i < \infty\}$.

A stochastic process $\{x(i),\ i \varepsilon T\}$, whose index set T is linear, is said to be

i) strictly stationary of order k, where k is a given positive integer, if any k points $i,\ i+1, \ldots i+k$ in T^{+}, where $T^{+} \triangleq \{x\,(i),\ i \geq o\}$, and any j in T^{+}, the k dimensional random vectors $\{x(i),\ x\,(i+1) \ldots x\,(i+k)\}$ and $\{x(i+j), \ldots x(i+j+k)\}$ are identically distributed.

ii) strictly stationary if for any integer k, it is strictly stationary of order k.

iii) widesense stationary (covariance stationary) if it possesses finite second moments, if its index set T is linear, and if its covariance kernel K(j,i) is a function only of the absolute difference $|j-i|$, in the sense that there exists a function R(n) such that for all j and i in T^+

$$K(j,i) = \text{Cov}[x(j), x(i)] = R_{xx}(j-i) \qquad (2\text{-}3)$$

or more precisely, $R_{xx}(m)$ has the property for every i and j in Z^+

$$\begin{aligned} \text{Cov}\ [x(i),x(i+m)] &= E[x(i)x(i+m)] \\ &= R_{xx}(m) \qquad (2\text{-}4) \end{aligned}$$

We call $R_{xx}(m)$ the covariance function (autocorrelation function) of wide sense stationary time series $\{x(i),\ i\varepsilon T^+\}$.

The second problem concerns the concepts of ergodicity and the strong law of large numbers in terms of linear processes. To present a complete discussion of this question is not reasonable for review purposes, but it is interesting to consider certain aspects of it. For strict sense stationary processes, the ergodic theorem is the strong law of large numbers and states that

if $\{x(i),i\varepsilon T^+\}$ is a strict sense stationary, ergodic random process and $E[|x(o)|]<\infty$,

then $\lim_{m\to\infty} \frac{1}{m} \sum_{i=1}^{m} x(i)=E[x(o)]$ with probability 1. (2-5)

In general, a stochastic process is said to be ergodic if it has the property that sample (or time) averages formed from an observed sequence of the process may be used as an approximation to the corresponding ensemble average.

Stationarity and ergodicity concepts are readily extended to two-dimensional random fields (for two dimensional signal, the term random field is preferred to random process).

The assumption that the field is stationary means that the statistics of a point in the field are not dependent on the location of the point. Then, a stationary, two-dimensional field has an autocorrelation function defined as:

$$R_{xx}(m,n) \triangleq E\{x(k,\ell)x(k+m,\ell+n)\} \tag{2-6}$$

and it is also said to be ergodic if the statistical (ensemble) average of random field x(k,l) is equal to the spatial averaging of all points. That means

$$E[x(k,\ell)] = \langle x \rangle \tag{2-7}$$

where $\langle x \rangle$ by definition represents spatial averaging

$$\langle x \rangle = \lim_{\substack{M_1\to\infty \\ M_2\to\infty}} \frac{1}{M_1} \frac{1}{M_2} \sum_{0}^{M_1} \sum_{0}^{M_1} x(k,\ell) \tag{2-8}$$

Now, consider a stationary sequence of random variables $\{x(i),\ i\varepsilon T\}$. The correlation function of the sequence may be written in the form

$$R_{xx}(n) = \int_{-\pi}^{\pi} e^{iwn}\, dF(w) \quad , \tag{2-9}$$

where F(w) is a nondecreasing function, called the spectral distribution. If F(w) is absolutely continuous with $F'(w) = f(w)$ almost everywhere, we may write

$$R_{xx}(n) = \int_{-\pi}^{\pi} e^{iwn} f(w)\, dw \qquad (2\text{-}10)$$

Under certain conditions, (e.g. $\sum_{n=0}^{\infty} |R_{xx}(n)| < \infty$, finite), the correlation function may be inverted to yield the spectral density as

$$f(w) = \frac{1}{2\pi} \sum_{-\infty}^{\infty} R_{xx}(n) e^{-inw} \qquad (2\text{-}11)$$

A sequence of random variables $\{x(i),\ i\varepsilon T\}$ is said to be a process of moving average if it admits the mean square representation

$$x(i) = \sum_{-\infty}^{\infty} h(i-j)\, u(j) \qquad (2\text{-}12)$$

where $\{u(j),\ j\varepsilon T\}$ is a collection of orthornomal random variables (sequence of white noise). If the sequence $\{h(i),\ i\varepsilon T\}$ is one sided (ie, $h(i) = 0,\ i<0$), then $\{x(i),\ i\varepsilon T\}$ is called a one sided moving average process.

A process is said to be regular if the error for prediction one time unit ahead is nonzero. It is known that a process is one of moving averages if and only if its spectral distribution function is absolutely continuous. Furthermore, a process with an absolutely continuous spectral distribution function is regular if its moving average representation is one sided. These facts serve to motivate the following definition of a linear process.

DEFINITION [5]

A linear process $\{x(i), i\varepsilon T\}$ is one having the structure

$$x(j) = \sum_{0}^{\infty} h(i)u(j-i) = \sum_{-\infty}^{j} h(j-i)u(i), \qquad (2\text{-}13)$$

where $\{u(i),\ i\varepsilon T\}$ is a sequence of independent and identically distributed (IID) random variables. The set of real constants $\{h(i),\ i\varepsilon T+\}$ is such that $\sum_{0}^{\infty}|h(i)| < \infty$, and the function $H(z) = \sum_{0}^{\infty} h(i)\ z^{-i}$ where z is a complex variable, is analytic and has no poles outside the unit circle in the z plane. The correlation function of this process is given by

$$R_{xx}(n) = \sum_{0}^{\infty} h(j)h(j+n) \qquad (2\text{-}14)$$

and the corresponding spectral density $f(w)$ is

$$f(w) = \frac{1}{2\pi}\left|\sum_{0}^{\infty} h(j)e^{ijw}\right|^2 = \frac{1}{2\pi}\left|H(e^{iw})\right|^2 \qquad (2\text{-}15)$$

It is assumed further that the process (2-13) has a rational spectral density, that is

$$f(w) = \frac{1}{2\pi}\left|H(e^{iw})\right|^2 = \frac{1}{2\pi}\left|\frac{B(e^{iw})}{A(e^{iw})}\right|^2, \qquad (2\text{-}16)$$

where both $A(e^{iw})$ and $B(e^{iw})$ are polynomials in e^{iw} of finite order with all their poles inside the unit circle. The process (2-13) may be generated by passing the sequence $\{u(i),\ i\varepsilon T\}$ through the digital filter $H(z)$. By the assumption in the definition of the linear process and by the restrictions on the spectrum, there exists an inversion $D(z)$ where

$$D(z) = \frac{A(z)}{B(z)} = \frac{1}{H(z)}$$

In general, D(z) and H(z) may be infinite polynomials in z. D(z) may be written as the one-sided sequence

$$D(z) = \sum_{0}^{\infty} d_i z^{-i} \qquad (2\text{-}17)$$

Passing the x(i) sequences through the digital filter will recover the generating sequence u(i), that is

$$u(j) = \sum_{-\infty}^{j} d(j-i)x(i) = \sum_{0}^{\infty} d(i)x(j-i) \qquad (2\text{-}18)$$

This is called an operation inversion. In general, we will have, assuming $a_o = 1$

$$H(z) = \frac{\sum_{0}^{n} b_i z^{-i}}{1 + \sum_{1}^{m} a_i z^{-i}} = \sum_{0}^{\infty} h(i) z^{-i} \qquad (2\text{-}19)$$

and the process$\{x(i), i \varepsilon T\}$ may be represented in two ways

$$\text{i)} \quad x(j) = \sum_{0}^{\infty} h(i)u(j-i)$$

$$\text{ii)} \quad x(j) = \sum_{0}^{m} b_i u(j-i) - \sum_{1}^{n} a_i x(j-i) \qquad (2\text{-}20)$$

The second representation indicates that if H(z) is an all-pole function, then we have

$$\text{iii)} \quad \sum_{0}^{m} a_i x(j-i) = u(j). \qquad (2\text{-}21)$$

In this case, since inversion uses only a finite number of past samples, the process is called "finitely invertable." It is clear that finitely invertable linear processes form a subclass of autoregressive schemes for which case the set $\{u(i)\}$ in (2-21) would be orthonomal rather than independent. The concepts and definitions above can be readily extended to a two-dimensional linear process. A two-dimensional linear process $\{x(m,n),\ m\varepsilon z_1,\ n\varepsilon z_2\}$ could have the structure

$$x(m,n) = \sum_{0}^{\infty}\sum_{0}^{\infty} h(k,\ell)u(k-m,\ell-n)$$

$$= \sum_{-\infty}^{m}\sum_{-\infty}^{n} h(k-m,\ell-n)u(k,\ell)\ , \qquad (2\text{-}22)$$

where $\{u(k,\ell),\ k\varepsilon z_1,\ \ell\varepsilon z_2\}$ is a two-dimensional sequence of IID random variables with zero mean and unit variance. The set of real constant $\{h(k.\ell),\ k\varepsilon z_1^{+},\ \ell\varepsilon z_2^{+}\}$ is such that $\sum_{0}^{\infty}\sum_{0}^{\infty}|h(k,\ell)|<\infty$, and the function $H(Z_1,Z_2) = \sum_{0}^{\infty}\sum_{0}^{\infty}h(k,\ell)Z_1^{-k}Z_2^{-\ell}$, where Z_1 and Z_2 are complex variables, is analytic. The correlation function of this process is given by

$$R_{xx}(m,n) = \sum_{0}^{\infty}\sum_{0}^{\infty}h(k,\ell)h(k+m,\ell+n) \qquad (2\text{-}23)$$

and the corresponding spectral density $f(w_1,w_2)$ is

$$f(w_1,w_2) = \left(\frac{1}{2\pi}\right)^2 \left|\sum_{0}^{\infty}\sum_{0}^{\infty} h(k,\ell)e^{iw_1k}\ e^{iw_2\ell}\right|$$

$$= \left(\frac{1}{2\pi}\right)^2 \left|\ H(e^{iw_1},\ e^{iw_2})\ \right|^2 \qquad (2\text{-}24)$$

With the same reasoning as in the one-dimensional case, we have in general,

$$H(Z_1,\ Z_2) = \frac{\sum_{0}^{N_1}\sum_{0}^{N_2} b_{ij}\ Z_1^{-i}Z_2^{-j}}{1+\sum_{\substack{i=o\\(i,j)\neq(o,o)}}^{M_1}\sum_{j=o}^{M_2} a_{ij}\ Z_1^{-i}\ Z_2^{-j}}$$

$$= \sum_{0}^{\infty}\sum_{0}^{\infty} h(k,\ell)Z_1^{-k}Z_2^{-\ell}. \qquad (2\text{-}25)$$

and the process $\{x(k,l),\ k\varepsilon Z_1,\ l\varepsilon Z_2\}$ may be represented in two ways

$$\text{i)}\quad x(k,\ell) = \sum_{0}^{\infty}\sum_{0}^{\infty} h(i.j)u(k-i,\ell-j)$$

$$\text{ii)}\quad x(k,\ell) = \sum_{0}^{N_1}\sum_{0}^{N_2} b_{ij}u(k-i,\ell-j) - \underset{(i,j)\neq(o,o)}{\sum_{0}^{M_1}\sum_{0}^{M_2}} a_{ij}x(k-i,\ell-j) \qquad (2\text{-}26)$$

It should be noted that the moving average scheme would be accomplished by passing the orthonomal IID random variables through the nonrecursive filter and the autoregressive schemes through the recursive filter for both one-dimensional and two-dimensional random process.

In the study of systems subject to the random signals, the concept of power spectral density function is of importance.

For a given transfer function of a linear filter, the cross power spectrum between input and output of the filter, and the output power spectrum is of primary concern.

Consider a (continuous) system subjected to a random input signal. Given a linear system with a transfer function $H(jw)$, the input to the filter a stationary process $x(t)$ with an autocorrelation function $R_{xx}(z)$ and a power spectral density function $G_{xx}(w)$, then the following relationships are obtained

$$G_{xy}(w) = G_{xx}(w)H^*(jw)$$

$$G_{yy}(w) = G_{xy}(w)H(jw) \qquad (2\text{-}27)$$

Combining above two equations,

$$G_{yy}(w) = G_{xx}(w)|H(jw)|^2, \qquad (2\text{-}28)$$

where $G_{xy}(w)$ is the cross power spectrum and $G_{yy}(w)$ is the output power spectrum, the cross correlation would be calculated by using the inverse Fourier transform of $G_{xy}(w)$.

For the continuous two-dimensional linear system of $H(jw_1, jw_2)$, subjected to a stationary two-dimensional input signal of power spectrum $G_{xx}(w_1, w_2)$.

$$G_{xy}(w_1, w_2) = G_{xx}(w_1, w_2) H(jw_1, jw_2)$$

$$G_{yy}(w_1, w_2) = G_{xy}(w_1, w_2) \; H(jw_1, jw_2) \quad (2\text{-}29)$$

Again combining above two relationships, the output power spectral density function is

$$G_{yy}(w_1, w_2) = G_{xx}(w_1, w_2) |H(jw_1, jw_2)|^2 \quad (2\text{-}30)$$

Consider a discrete linear system to which a random input sequence is applied with a transfer function $H(z)$. The input sequence is a stationary process $x(i)$ with an autocorrelation function $R_{xx}(m)$ and its $\mathfrak{Z}$-transform $G_{xx}(z)$, where $G_{xx}(z)$ is equivalent to the power spectral density in the continuous case, i.e.

$$G_{xx}(z) \triangleq \mathfrak{Z}[R_{xx}(m)] \quad (2\text{-}31)$$

Then,

$$G_{xy}(z) = G_{xx}(z) \; H(z)$$

$$G_{yy}(z) = G_{xx}(z) \; H(z) H(z^{-1}) \quad (2\text{-}32)$$

For the two-dimensional discrete case,

$$G_{xy}(z_1, z_2) = G_{xx}(z_1, z_2) H(z_1, z_2)$$

and $G_{yy}(z_1, z_2) = G_{xx}(z_1, z_2) H(z_1, z_2) H(z_1^{-1}, z_2^{-1})$ (2-33)

As a summary of this section, a discrete-time linear process can be considered to be the result of digital filtering an independent identical random sequence having zero mean and unit variance. The moving average scheme is the result of filtering through the nonrecursive filter and the autoregressive scheme is that of filtering through the recursive filter.

And for a linear system, the relations between transfer function, power spectrum, and auto correlation are given by

$$G_{yy} = G_{xx} |H|^2$$

$$G_{xy} = G_{xx} H$$

R (autocorrelation function) = $\mathfrak{J}$ transform of power spectral density function.

The Figure 2-1 shows the block diagram which describes the various relations and concepts.

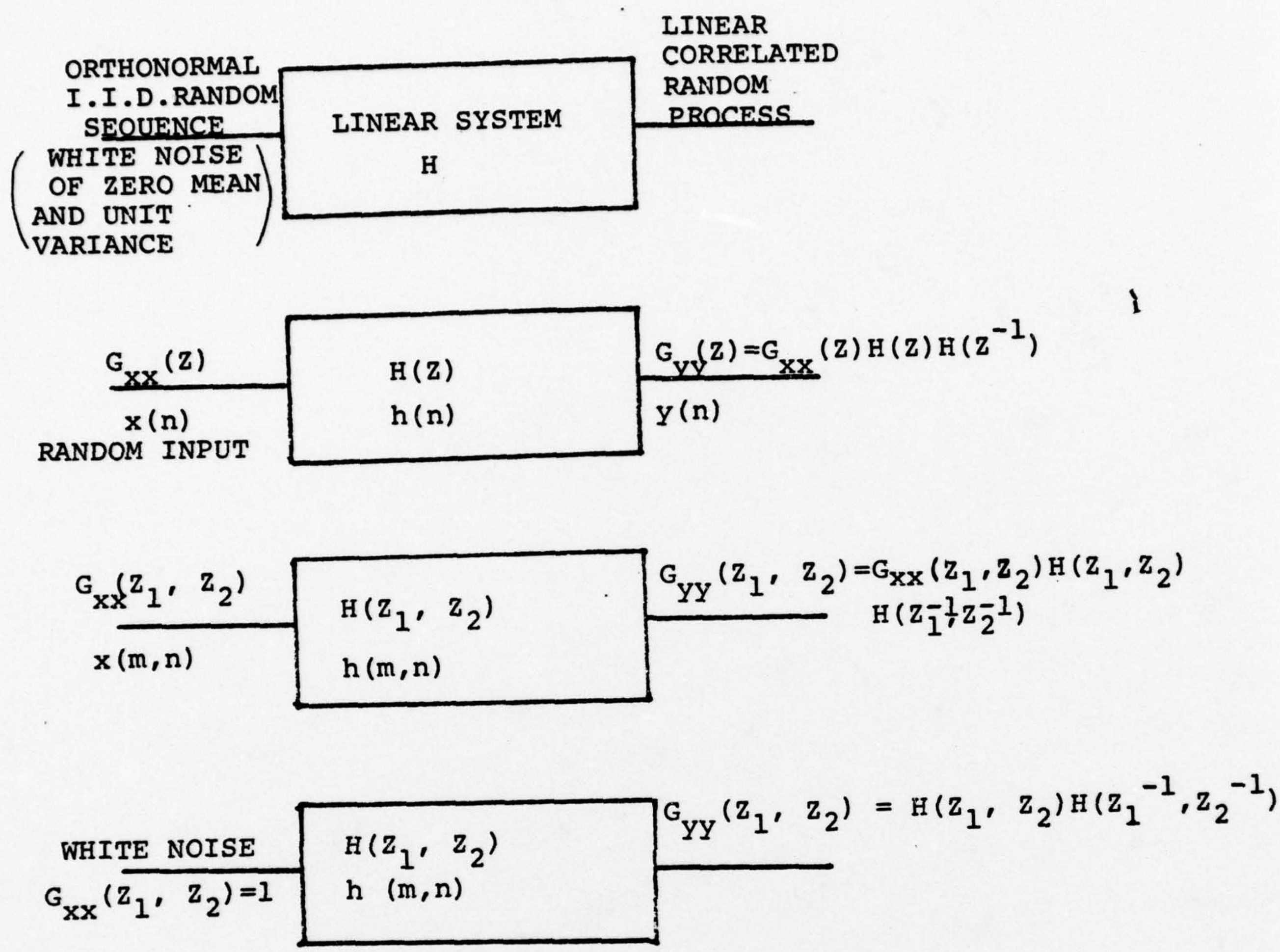


FIGURE 2 -1
STATISTICAL
PROPERTIES OF LINEAR SYSTEM

C. MODELING OF LINEAR STOCHASTIC PROCESS

1. Introduction

A more active concern at this time is that of system modeling. It has been shown in a previous section that a linear stochastic process (or field) can be generated by filtering white noise through a linear filter. The problem can be stated as follows. What is the filter equation (difference equation) that produces a typical random process with a specified autocorrelation function? That is, with the knowledge of second-order statistics, determine the filter coefficient a's and b's in equations (2-20) and (2-26). It is clear that if one is successful in developing a parametric model for the behavior of some random process, then the model can be used for different applications, such as prediction, estimation, smoothing, etc.. As far as the general modeling problem goes, one of the most powerful models currently in use is that where a signal x(n) is considered to be the output of some system (filter) with unknown input u(n) such that the following relationship holds

$$x(n) = -\sum_{k=1}^{p} a_k x(n-k) + G\sum_{\ell=0}^{q} b_\ell u(n-\ell), \quad b_0=1 \qquad (2\text{-}34)$$

where a_k, $1\leq k\leq p$, b_ℓ $1 \leq\ell\leq q$

and the gain G are the parameters of the hypothesized system. Equation (2-23) says that the "output" x(n) is a linear function of past outputs and present and past inputs. That is, the signal x(n) is predictable from linear combinations of past outputs and inputs. For the two-dimensional case, the difference equation corresponding to equation (2-34) would be

$$x(k,\ell) = -\sum_{i=o}^{M_1}\sum_{j=o}^{M_2} a_{ij}x(k-i,\ell-j)+G\sum_{\ell_1=u}^{L_1}\sum_{\ell_2=u}^{L_2} b_{\ell 1,\ell 2}u(k-\ell_1,\ell-\ell_2)$$

where $(i,j)\neq(0,0)$ (2-35)

Equations (2-34) and (2-35) can also be specified in the frequency domain by taking the z transform of both sides of Eq (2-34) and Eq (2-35).

$$H(z) = \frac{X(Z)}{U(Z)} = G\,\frac{1+\sum_{\ell=1}^{q} b_\ell Z^{-\ell}}{1+\sum_{k=1}^{p} a_k Z^{-k}} \qquad (2\text{-}36)$$

where $X(Z) = \sum_{n=-\infty}^{\infty} x(n)Z^{-n}$ is the Z transform of x(n), and U(Z) is the z transform of u(n).

For the two-dimensional case,

$$H(Z_1,Z_2)=\frac{X(Z_1,Z_2)}{U(Z_1,Z_2)}=G\,\frac{\sum_{\ell_1=0}^{L_1}\sum_{\ell_2=0}^{L_2} b_{\ell_1\ell_2}Z_1^{-\ell_1}Z_2^{-\ell_2}}{1+\sum_{\substack{i=o\\(i.j)\neq(o.o)}}^{M_1}\sum_{j=o}^{M_2} a_{ij}Z_1^{-i}Z_2^{-j}} \qquad (2\text{-}37)$$

where $X(Z_1,Z_2) = \mathfrak{Z}[x(k,\ell)]$

$U(Z_1,Z_2) = \mathfrak{Z}[u(k,\ell)]$

$H(Z)$ and $H(Z_1,Z_2)$ in Eqs (2-36) and (2-37) are the general pole zero models.

The roots of the numerator and denominator polynomials are the zeros and poles of the model, respectively. There are two special cases of the model that are of interest.

i) all-zero model $a_k = 0$, $a_{ij}=0$

ii) all-pole model $b_\ell = 0$, $b_{\ell_1 \ell_2} = 0$

$$1 \leq \ell \leq q$$

$$0 \leq \ell_1 \leq L_1$$

$$0 \leq \ell_2 < L_2$$

$$\text{But } (\ell_1, \ell_2) \neq (0,0)$$

As mentioned in section II-B, the all-zero model is known in the statistical literature as the moving average (MA) model and the all-pole model is then known as the autoregressive (AR) model. The pole-zero model is then known as the autoregressive moving average (ARMA) model. It should be recalled here that we are interested in the case where $u(n)$ or $u(k,\ell)$ is white noise, and this will be treated as a special case in the following. The modeling problem can be stated as "given signal $x(n)$ or $x(k,\ell)$, find the filter coefficients (a's and b's) and the gain, G, in Equation (2-34) in some manner." Two approaches will be given for a solution of the above problem. The first is the method of least squares which is based on the optimal estimation concept, and the second is the filter response method in which linear system properties are used. The one-dimensional case will be treated first, then two-dimensional case including some examples. For example, a lowpass correlated random process (field) and band limited random process (field) are chosen since they represent practical examples.

2. The Method of Least Squares

Although the following can be applied to the deterministic signal and stochastic processes, stationary or nonstationary, it is emphasized for only a stationary random process and only the all-pole model is considered [6]. In the all-pole model, we assume that the signal x(n) is given by as a linear combination of past values and some input u(n):

$$x(n) = -\sum_{k=1}^{P} A_k x(n-k) + Gu(n) \qquad (2\text{-}38)$$

where G is a gain factor.

Here, it is assumed that the input u(n) is totally unknown, which is the case in many applications. Therefore, the signal x(n) can be predicted only approximately from a linearly weighted summation of past samples.

Let this approximation of x(n) be $\hat{x}(n)$, where

$$\hat{x}(n) = -\sum_{k=1}^{P} A_k x(n-k) , \qquad (2\text{-}39)$$

then the error between the actual value x(n) and the predicted value $\hat{x}(n)$ is given by

$$e(n) = x(n) - \hat{x}(n) = x(n) + \sum_{k=1}^{P} A_k x(n-k). \qquad (2\text{-}40)$$

e(n) is also known as the residual. In the method of least squares, the parameter A_k's are obtained as a result of the minimization of the mean square error with respect to all of the parameters.

If the signal x(n) is assumed to be a sample of a random process, then the error in equation (2-40) is also a sample of a random process. In the least square method, we minimize the expected value of the square of the error.

$$E[e^2(n)] = E\{[x(n) + \sum_{k=1}^{P} A_k x(n-k)]^2\} \qquad (2\text{-}41)$$

$E[e^2(n)]$ is minimized by setting

$$\frac{\partial E[e^2(n)]}{\partial A_i} = 0, \quad 1 \leq i \leq p \qquad (2\text{-}42)$$

From (2-41) and (2-42) we obtain the set of equations

$$\sum_{k=1}^{P} A_k E[x(n-k)x(n-i)] = -E[x(n)x(n-i)] \quad 1 \leq i \leq P \qquad (2\text{-}43)$$

Then the minimum average error is given by

$$E_p = E[x^2(n)] + \sum_{k=1}^{P} A_k E(x(n)x(n-k)) \qquad (2\text{-}44)$$

For a stationary process x(n), we have

$$E[x(n-k)x(n-i)] = R_{xx}(i-k)$$

where $R_{xx}(i)$ is the autocorrelation of the process.

Note that equations (2-42) and (2-44) lead to the well known orthogonality principle [7]. Since in the least squares method, we assumed that the input is unknown, it doesn't make much sense to determine a value for the gain G. However, there are certain interesting points that can be made.

Equation (2-39) can be written as

$$x(n) = -\sum_{k=1}^{P} A_k x(n-k) + e(n) \qquad (2\text{-}45)$$

Comparing (2-45) and (2-38), it is seen that the only input signal u(n) that will result in the signal x(n) as output is that where $Gu(n) = e(n)$. (2-46)

That is, the input signal is proportional to the error signal. e(n) can be also considered as the modeling error. The error variance can be calculated by Equation (2-41) and the filter coefficient A_k (k=1 . . . p) would be calculated by equation (2-43) if the correlation function of process x(n) is available.

At this moment, it should be recalled that a linear random process is generated by linear filtering of white noise. Therefore, we are interested in white noise inputs for the purpose of modeling a given linear random process. That is, the input u(n) is assumed to be a sequence of uncorrelated samples (white noise) with zero mean and unit variance.

$$E[u(n)] = 0, \quad E[u(n)u(m)] = \delta_{mm}$$

Then the output x(n) forms a stationary random process

$$x(n) = -\sum_{k=1}^{P} A_k x(n-k) + Gu(n) \qquad (2\text{-}47)$$

Multiplying equation (2-34) by x(n-i) and taking the expectation,

$$E[x(n)x(n-j)] = -\sum_{k=1}^{P} A_k E[x(n-k)x(n-i)] + E[Gu(n)x(n-i)] \qquad (2\text{-}48)$$

Noting that u(n) and x(n-i) are uncorrelated for i>0 and recalling that for stationary process, $E[x(n)x(n-i)]=R_{xx}(i)$, Equation (2-35) turns out to be

$$R_{xx}(i) = -\sum_{k=1}^{p} A_k R_{xx}(i-k) \text{ for } p \geq i > o \qquad (2\text{-}49)$$

and R(o) can be obtained by plugging x(n) of Equation (2-38) into Equation (2-48)

$$R_{xx}(o) = -\sum_{k=1}^{p} A_k R_{xx}(k) + G^2 \qquad (2\text{-}50)$$

Therefore, the gain can be given by

$$G^2 = R_{xx}(o) + \sum_{k=1}^{p} A_k R_{xx}(k) \qquad (2\text{-}51)$$

It is noted that through Equation (2-46), that is Gu(n)=e(n), the white noise input of zero mean and unit variance generates the random process e(n), which is again white with zero mean and variance of G^2. Therefore, from Equation (2-49), the recursive filter coefficients A_k,(k=1, ...p) can be calculated and using these values the gain G would be calculated by Equation (2-51) with the knowledge of autocorrelation function of a given class of linear random process. So far, modeling of one-dimensional stochastic process has been considered. Similar reasoning can readily be extended to the modeling of two-dimensional random fields. Again, the two-dimensional all-pole model is considered.

$$x(k,\ell) = -\sum_{\substack{i=o \\ (i,j)\neq(o,o)}}^{M_1} \sum_{j=o}^{M_2} A_{ij}\, x(k-i,\ell-j) + Gu(k,\ell) \qquad (2\text{-}52)$$

Let's define the set $\Omega(k,\ell)$ such that for all i,j

$(k-i,\ \ell-j)\ \varepsilon\Omega(k,\ell)$,

all the values of $x(k,\ell)$ in $\Omega(k,\ell)$ will be used to estimate the point $x(k,\ell)$.

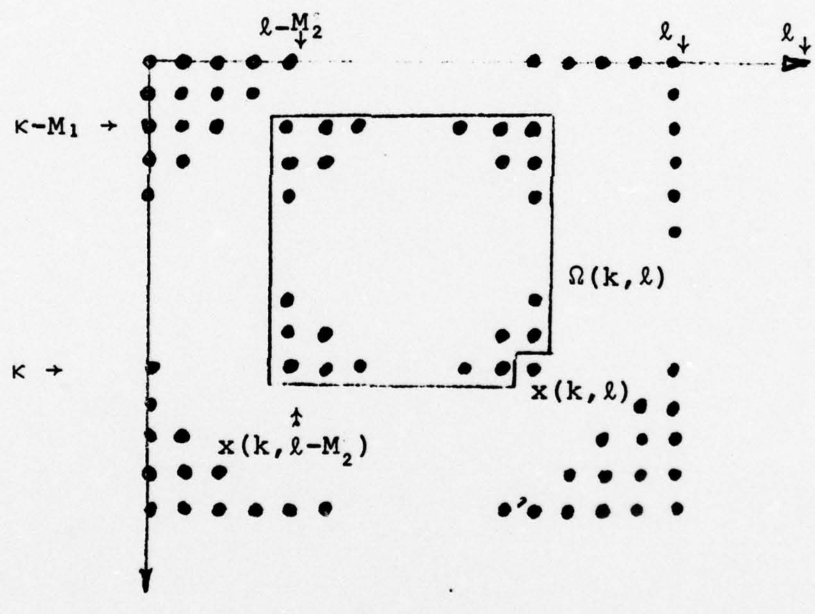


FIGURE 2-2

DEFINITION OF $\Omega(k,\ell)$

Again the coefficients A_{ij} will be determined such that the mean squared error is minimized. The estimate of $x(k,\ell)$, $\hat{x}(k,\ell)$, is given by a linear combination of the previous values.

$$\hat{x}(k,\ell) = - \Sigma_{\Omega}\Sigma x(k-i,\ell-j) \qquad (2\text{-}53)$$

The mean squared error is

$$E[e^2(k,\ell)]=E\{[x(k,\ell)- \hat{x}(k,\ell)]^2\} \qquad (2\text{-}54)$$

If $E[e^2(k,\ell)]$ is minimized, $\hat{x}(k,\ell)$ is the "linear least squares estimate" of $x(k,\ell)$.

Going through the same procedure as in the one-dimensional case, that is, substituting (2-53) in (2-54) and differentiating with respect to each A_{ij}, setting derivatives equal to zero, we obtain the following set of simultaneous equations for the unknown A_{ij}

$$E\{[x(k,\ell)-\hat{x}(k,\ell)]x(i,j)\} = o \text{ for all} (ij)\varepsilon\Omega, \qquad (2\text{-}55)$$

which says that the coefficients A_{ij} must be such that the estimation error $[x(k,\ell)-\hat{x}(k,\ell)]$ is statistically orthogonal to each $x(i,j)$ that is used to form the linear estimate. This is known as the orthogonality principle in linear least square estimation.

The modeling error is the difference between the true value $x(k,\ell)$ and the estimate $\hat{x}(k,\ell)$. By definition,

$$e(k,\ell)=x(k,\ell)-\hat{x}(k,\ell)=Gu(k,\ell) \qquad (2\text{-}56)$$

from the equation (2-52) and (2-53).

Again, we are interested in the white noise field of zero mean and unit variance. Then the modeling error is also a random field.

Rewriting the Equation (2-52) in terms of the error $e(k,\ell)$ gives

$$x(k,\ell)=-\Sigma_{\Omega}\Sigma\ A_{ij}x(k-i,\ell-j)+e(k,\ell) \qquad (2\text{-}57)$$

To calculate the error variance, multiply (2-57) by $x(k-m,\ell-n)$ and take the expectations

$$E[x(k,\ell)x(k-m,\ell-n)]=-\ \Sigma_{\Omega}\Sigma A_{ij}E[x(k-i,\ell-j)x(k-m,\ell-n)]$$
$$+E[e(k,\ell)x(k-m,\ell-n)] \qquad (2\text{-}58)$$

For the stationary process,

$E[x(k-i,\ell-j)x(k-m,\ell-n)]=R_{xx}[m-i,n-j]$, then (2-58) will be

$$R_{xx}(m,n)=-\Sigma_{\Omega}\Sigma A_{ij}R_{xx}(m-i,n-j) \text{ for all}(m,\ n)\epsilon\Omega. \qquad (2\text{-}59)$$

The second term on the right side of (2-58) is zero, because of the orthogonality principle and $R(0,0)$ can be obtained by the following equation:

$$R_{xx}(0,0)=-\Sigma_{\Omega}\Sigma\ A_{ij}R_{xx}(i,j)+G^2 \qquad (2\text{-}60)$$

Therefore the modeling error variance,

$E[e^2(k,\ell)]=E\{[(Gu(k,\ell)]^2\} = G^2$ is given by the equation:

$G^2=E[e^2(k,\ell)]=R_{xx}(0,0)+\Sigma_{\Omega}\Sigma\ A_{ij}R_{xx}\ (i,j)$ and the mean of error $e(k,\ell)$ is $E[e(k,\ell)]=E[Gu(k,\ell)]=0$

Again with the knowledge of autocorrelation function of the two-dimensional stationary random field, the filter coefficient A_{ij} can be obtained from the Equation (2-59) and using these values of A_{ij}, the gain G in Equation (2-52) can be calculated by Equation (2-60).

<u>Example 1</u> Consider a one-dimensional stationary band limited random process for which the autocorrelation function is given by

$$R_{xx}(m) = \rho^{|m|} \cos(w_o m)$$

Find a model which describes this process. An all-pole model is chosen such that

$$x(n) = - \sum_{k=1}^{P} A_k x(n-k) + Gu(n)$$

where $E[e(n)] = 0$

$E[u(n)u(m)] = \delta mn$

One has to choose the order of the difference equation; here, a second order model (p=2) is chosen. Then

$$x(n) = -A_1 x(n-1) - A_2 x(n-2) + Gu(n)$$

The problem shrinks down to that of finding A_1, A_2, G with the given $R_{xx}(m)$.

From Equation (2-49)

$$R_{xx}(i) = - \sum_{k=1}^{2} A_k R_{xx}(i-k)$$

$$R_{xx}(1) = - A_1 R_{xx}(o) - A_2 R_{xx}(-1)$$

$$R_{xx}(2) = - A_1 R_{xx}(1) - A_2 R_{xx}(0)$$

Putting this in matrix form

$$\begin{pmatrix} -R_{xx}(o) & -R_{xx}(-1) \\ -R_{xx}(1) & -R_{xx}(0) \end{pmatrix} \begin{pmatrix} A_1 \\ A_2 \end{pmatrix} = \begin{pmatrix} R_{xx}(1) \\ R_{xx}(2) \end{pmatrix}$$

From $Rxx(m) = \rho^{|m|} \cos(w_o m)$

$$\begin{pmatrix} -1 & -\rho \cos w_o \\ -\rho \cos w_o & -1 \end{pmatrix} \begin{pmatrix} A_1 \\ A_2 \end{pmatrix} = \begin{pmatrix} \rho \cos w_o \\ \rho^2 \cos 2 w_o \end{pmatrix}$$

Therefore $A_1 = -\dfrac{\rho \operatorname{cox} w_o (1-\rho^2 \operatorname{cox} 2 w_o)}{1 - \rho^2 \cos^2 w_o}$

$$A_2 = \frac{\rho^2 \sin^2 w_o}{1 - \rho^2 \cos^2 w_o}$$

From Equation (2-51)

$$G^2 = R_{xx}(0) + \sum_{k=1}^{2} A_k R_{xx}(k)$$

$$= 1 + A_1 R_{xx}(1) + A_2 R_{xx}(2)$$

$$= 1 + A_1 \rho \cos w_o + A_2 \rho^2 \cos 2 W_o$$

$$= 1 - \frac{\rho^2 \cos 2w_o (1 - \rho^2 \cos 2 w_o)}{1 - \rho^2 \cos^2 w_o} + \frac{\rho^4 \sin^2 w_o \cos 2 w_o}{1 - \rho^2 \cos^2 w_o}$$

Example 2 Given a stationary two-dimensional random field with

$Rxx(m,n) = \rho^{|m|} \rho^{|n|}$,

find the autoregressive model of this random field (most mono-chromatic images can be assumed to have this form of auto-correlation function). A first-order model ($M_1 = 1$, $M_2 = 1$) is chosen. Then equation (2-52) can be written as follows:

$$x(k,\ell) = -A_{10} x(k-1,\ell) - A_{11} x(k-1, \ell-1) - A_{01} x(k,\ell^{-1}) + Gu(k,\ell)$$

Where $u(k,\ell)$ is white noise with zero mean and unit variance.

Then, using Equation (2-59)

$$R_{xx}(m,n) = -\sum_{\substack{i=o \\ (i,j)\neq(o.0)}}^{1} \sum_{j=0}^{1} A_{ij} R_{xx}(m-i,n-j) \text{ for all}(m, n) \in \Omega .$$

$$Rxx(1,0) = - A_{10} R_{xx}(0,0) - A_{11} R_{xx}(0 -1) - A_{01} Rxx(1,-1)$$

$$Rxx(1,1) = - A_{10} R_{xx}(0,1) - A_{11} R_{xx}(0.0) - A_{01} Rxx(1, 0)$$

$$Rxx(0,1) = - A_{10} R_{xx}(-1,1) - A_{11} R_{xx}(-1,0) - A_{01} R_{xx}(0, 0).$$

Putting this in matrix form gives

$$\begin{bmatrix} -Rxx(0,0) & -Rxx(0,-1) & -Rxx(1,-1) \\ -Rxx(0,-1) & -Rxx(0, 0) & -Rxx(1,0) \\ -Rxx(-1,1) & -Rxx(-1,0) & -Rxx(0,0) \end{bmatrix} \begin{bmatrix} A_{10} \\ A_{11} \\ A_{01} \end{bmatrix} = \begin{bmatrix} Rxx(1,0) \\ Rxx(1,1) \\ Rxx(0,1) \end{bmatrix}$$

For the given auto correlation function above

$$\begin{bmatrix} -1 & -\rho & -\rho^2 \\ -\rho & -1 & -\rho \\ -\rho^2 & -\rho & -1 \end{bmatrix} \begin{bmatrix} A_{10} \\ A_{11} \\ A_{01} \end{bmatrix} = \begin{bmatrix} \rho \\ \rho^2 \\ \rho \end{bmatrix}$$

which yields

$$A_{01} = -\rho$$

$$A_{11} = \rho^2$$

$$A_{10} = -\rho$$

The modeling error variance or the square of the gain G^2 can be calculated by Equation (2-60)

$$G^2 = R_{xx}(0,0) + \sum_{\substack{i=0 \\ (i,j)\neq(0,0)}}^{1} \sum_{j=0}^{1} A_{ij} R_{xx}(i,j)$$

$$= 1 + A_{01}R_{xx}(0,1) + A_{11}R_{xx}(1,1) + A_{10}R_{xx}(1,0)$$

$$= (1-\rho^2)^2$$

Therefore, the complete model for $R_{xx} = \rho^{|m|}\rho^{|n|}$ is

$$x(k,\ell) = \rho x(k,\ell-1) + \rho^2 x(k, \ell) + \rho x(k-1,\ell) + (1-\rho^2)u(k,\ell)$$

where $E\,(u(k.\ell)\,) = 0$

$$E[u(k.\ell)\; u(k-p,\ell-q)] = \delta_{pq}.$$

3. Method of Filter Response

Another method of modeling a linear stationary random process is based on the concept that a linear random process is a result of filtering white noise through a linear filter. In Section B. of this chapter, the properties of linear systems have been discussed.

For the discrete system, it is known that the filter output power spectral density is the Z transformation of the output correlation function. That is,

$$Gyy(Z) \triangleq \mathfrak{Z}[R_{yy}(m)]$$

$$Gyy(Z_1, Z_2) = \mathfrak{Z}[R_{yy}(m,n)]$$

and also it is noted that

$$Gyy(Z) = Gxx(Z)\,H(Z)\,H(Z^{-1})$$

$$Gyy(Z_1, Z_2) = Gxx(Z_1, Z_2)\,H(Z_1, Z_2)\,H(Z_1^{-1}, Z_2^{-1}),$$

where $Gxx(Z)$, $Gxx(Z_1, Z_2)$ is the input power spectral density function and $H(Z)$, $H(Z_1, Z_2)$ is the transfer function of the filter. Denoting the white noise input as $u(n)$ or $u(k,\ell)$ and the output of filter, which is a linear stationary random process we are going to model as $x(n)$ or $X(k,\ell)$, the problem can be stated as follows: For a given autocorrelation function of a linear random process (field) $R(m)$, $(R(m,n))$, find a linear system such that when the input is white noise, the output of the filter gives a given autocorrelation function $R(m)$, $(R(m,n))$.

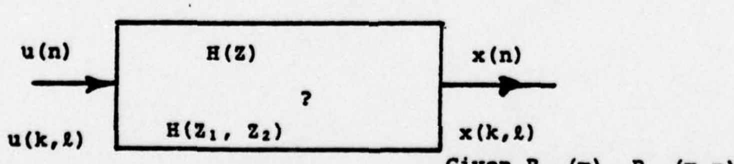


FIGURE 2-3

THE MODELING PROBLEM IN THE DISCRETE CASE

If the input is white noise, then

$G_{uu}(Z) = \text{Const},$

$G_{uu}(Z_1, Z_2) = \text{Const}.$

Therefore, the solution for the required filter is to find a function $H(Z)$ or $H(Z_1, Z_2)$ which satisfies

$G_{xx}(Z) = \text{Const}\cdot H(Z)H(Z^{-1})$

$$G_{xx}(Z_1, Z_2) = \text{Const}\cdot H(Z_1, Z_2)H(Z_1^{-1}, Z_2^{-1}) \quad (2\text{-}62)$$

where $G_{xx}(Z)$, $G_{xx}(Z_1, Z_2)$ is known through the relation

$G_{xx}(Z) = \mathfrak{Z}[R_{xx}(m)]$

$G_{xx}(Z_1, Z_2) = \mathfrak{Z}[R_{xx}(m,n)]$

since $R_{xx}(m)$, $R_{xx}(m,n)$ is given.

But for the two-dimensional case, there is an inherent difficulty in factorization of $G_{xx}(Z_1, Z_2)$ to $H(Z_1, Z_2)H(Z_1^{-1}, Z_2^{-1})$. Therefore, only separable functions can be modeled by this technique.

Example 1 For a given stationary linear random process with

$R_{xx}(n) = \sigma^2\rho^{|m|}\cos(w_o m)$, $m = o, 1, 2 \ldots$

Find a difference equation which will give a random process with autocorrelation above.

$G_{xx}(Z) = \mathfrak{Z}[R_{xx}(m)]$

$$= \frac{\sigma^2(1-\rho^2).[-Z\rho\cos w_o+(1+\rho^2)-Z^{-1}\cos w_o]}{(1-2\rho Z\cos w_o+\rho^2Z^2)(1-2\rho Z^{-1}\cos w_o+\rho^2Z^{-2})} \quad (2\text{-}51)$$

The second step is to find a factored expression for $G_{xx}(Z)$, i.e.

$G_{xx}(Z) = G_1(Z) \,.\, G_1(Z^{-1})$

Assuming that $G_{xx}(Z)$ has the form

$$G_{xx}(Z) = \sigma^2(1-p^2) \frac{az^{-1} + b}{1-2\rho z^{-1}\cos w_o+\rho^2 z^{-2}} \frac{az + b}{1-2\rho z \cos w_o+\rho^2 z^2} \quad (2\text{-}52)$$

Comparing (2-52) and (2-51), a and b can be obtained as

$$a = \frac{1}{2} (\quad 1-\rho \cos w_o+\rho^2 \quad + \quad 1+\rho \cos w_o+\rho^2$$

$$b = \frac{1}{2} \quad 1-\rho \cos w_o+\rho^2 \quad\quad 1+\rho \cos w_o+\rho^2$$

From equation (2-50), if we choose $H(Z) = \dfrac{az^{-1} + b}{1-2\rho z^{-1}\cos w_o+\rho^2 z^{-2}}$

then the term $\sigma^2(1-\rho^2)$ in Equation (2-52) can be considered as $G_{uu}(Z)$

Therefore

$$R_{uu}(m) = \begin{cases} (1-\rho^2)\,\sigma^2 & \text{if } m=0 \\ 0 & \text{if } m\neq 0 \end{cases}$$

and the complete model is drawn block diagram form in Figure 2-4.

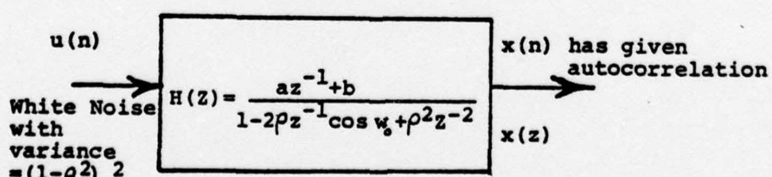


FIGURE 2-4

FILTER FOR ONE-DIMENSIONAL BAND PASS PROCESS

To put the system's input-output relation into state equation form, it is defined for convenience as

$$u(z) = z^{-1} w(z)$$

where w(n) is also white noise with same autocorrelation function

$$R_{ww}(m) = \begin{cases} (1-\rho^2)\sigma^2 & \text{if } m=0 \\ 0 & \text{if } m\neq 0 \end{cases}$$

then the transfer function is

$$H(Z) = \frac{X(Z)}{U(Z)} = \frac{az^{-1} + b}{1-2\rho Z^{-1}\cos w_o+\rho^2 z^{-2}}$$

Defining

$$X_1(z) = -\rho^2 z^{-1} X_2(z)$$

$$X_2(z) = \frac{U(z)}{1-2\rho \cos w_o z^{-1}+\rho^2 z^{-2}}$$

$$X(z) = (az^{-1}+b)X_2(z)$$

then

$$X_1(n) = -\rho^2 x_2(n-1)$$

$$X_2(n) = x_1(n-1) + 2\rho \cos w_o x_2(n-1) + w(n-1)$$

and

$$X(n) = ax_2(n-1)+bx_2(n) = -\rho^{-2}ax(n)+bx_2(n)$$

Putting the state and output equations in matrix form gives

$$\begin{pmatrix} x_1(n) \\ x_2(n) \end{pmatrix} = \begin{pmatrix} 0 & -\rho^2 \\ 1 & 2\cos w_o \end{pmatrix} \begin{pmatrix} x_1(n-1) \\ x_2(n-1 \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} w(n-1)$$

$$x(n) = (-\rho^{-2}a \quad b) \begin{pmatrix} x_1(n) \\ x_2(n) \end{pmatrix}$$

Example 2 The two-dimensional "band limited" discrete Markov process is defined by the autocorrelation function

$$R_{xx}(m,n) = \sigma^2 [\rho_1^{(m)} \cos (w_1 m)] [\rho_2^{(m)} \cos(w_2 n)] \quad \begin{matrix} m=0,1,2... \\ n=0,1,2... \end{matrix}$$ then

the discrete power spectral density

$$G_{xx}(z_1, Z_2) = \sigma^2 (1-\rho_1^2)(1-\rho_2^2) \frac{A(z_1, z_2)}{B(Z_1, Z_2)}$$

where

$$A(Z_1, Z_2) = [-Z_1 \mathrm{Cos} w_1 + (1+\rho_1^2) - Z_1^{-1} \mathrm{Cos} W_1][-Z_2 \mathrm{Cos} W_2 + (1+\rho_2^2) - Z_2^{-1} \mathrm{Cos}\ w_2]$$

$$B(Z_1, Z_2) = [(1-2\rho_1 z_1 \cos w_1 + \rho_1^2 z_1^2)(1-2\rho_1 z_1^{-1} \cos w_1 + \rho^2 z_1^{-2})]$$
$$[(1-2\rho_2 z_2 \cos w_2 + \rho_2^2 z_2^2)(1-2\rho_2 z_2^{-1} \cos w_2 + \rho_2^2 z_2^{-2})]$$

Putting $A(Z_1, Z_2)$ in the following form

$$A(Z_1, Z_2) = [(a_1 z_1^{-1} + b_1)(a_2 z_2 + b_1)][(a_2 z_2^{-1} + b_2)(a_2 z_2 + b_2)]$$

and comparing this equation and above $A(Z_1 Z_2)$, a_1, a_2, b_1, b_2 obtains

$$a_1 = \frac{1}{2}\left(\sqrt{1-\rho_1 \cos w_1 + \rho_1^2} + \sqrt{1 + \rho_1 \cos w_1 + \rho_1^2}\right)$$

$$a_2 = \frac{1}{2}\left(\sqrt{1-\rho_2 \cos w_2 + \rho_2^2} + \sqrt{1 + \rho_2 \cos w_2 + \rho_2^2}\right)$$

$$b_1 = \frac{1}{2}\left(\sqrt{1-\rho_1 \cos w_1 + \rho_1^2} - \sqrt{1 + \rho_1 \cos w_1 + \rho_1^2}\right)$$

$$b_2 = \frac{1}{2}\left(\sqrt{1-\rho_2 \cos w_2 + \rho_2^2} - \sqrt{1 + \rho_2 \cos w_2 + \rho_2^2}\right)$$

Let $$H(Z_1, Z_2) = \frac{(a_1 z_1^{-1} + b_1)(a_2 z_2^{-1} + b_2)}{(1-2\rho_1 z_1^{-1} \cos w_1 + \rho_1^2 z_1^{-2})(1-2\rho_2 z_2^{-1} \cos w_2 + \rho_2^2 z_2^{-2})}$$

then from (2-50)

$$R_{uu}(m,n) = \begin{cases} \sigma^2(1-\rho_1^{\,2})(1-\rho_2^{\,2}) & \text{if } n=o,\ m=0 \\ o & \text{if } n\neq 0,\ m\neq 0 \end{cases}$$

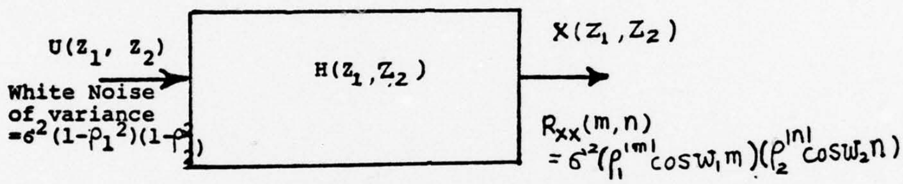


FIGURE 2-5

FILTER TO GENERATE BAND PASS RANDOM FIELD

It is convenient to define.

$$U(Z_1\ ,\ Z_2) = Z_1^{-1}Z_2^{-1}W(z_1,\ z_2)$$

$W(k,\ell)$ is also white noise with the same statistics as $w(k,\ell)$.

$$R_{ww}(n,m) = \begin{cases} \sigma^2(1-\rho_1^{\,2})(1-\rho_2^{\,2}) & \text{if } n=0 \text{ and } m=0 \\ 0 & \text{if } n\neq 0 \text{ or } m\neq 0 \end{cases}$$

Then, the filter has the form

$$\frac{X(Z_1,\ Z_2)}{W(Z_1,\ Z_2)} = \frac{(a_1z_1^{-1}+b)\ (\ a_2z_2^{-1}+\ b_2)\ Z_1^{-1}Z_2^{-1}}{(1-2\rho_1z_1^{-1}\cos\ w_1+\rho_1z_1^{-2})\,(1-2\rho_2z_2^{-1}\cos\ w_2+\rho_2^2z_2^{-2})}$$

The following definitions are made:

$$N_1(Z_1, Z_2) = -\rho_1^2 z_1^{-1} N_2(Z_1, Z_2)$$

$$N_2(Z_1, Z_2) = \frac{Z_2^{-1} w(Z_1, Z_2)}{1-2\rho_1 z_1^{-1}\cos w_1 + \rho_1^2 z_1^{-2}}$$

$$X_1(Z_1, Z_2) = (a_1 z_1^{-1} + b_1) N_2(Z_1, Z_2)$$

From the last three definitions one can write the set of difference equations

$$\begin{pmatrix} N_1(k,\ell) \\ N_2(k,\ell) \end{pmatrix} = \begin{pmatrix} 0 & -\rho_1^2 & N_1(k,\ell-1) \\ 1 & 2\rho_1\cos w_1 & N_2(k,\ell-1) \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} w(k,\ell-1)$$

$$x_1(k,\ell) = -\rho_1^2 a_1 N_1(k,\ell) + b_1 N_2(k,\ell)$$

Now, additional definitions are made.

$$M_1(Z_1, Z_2) = -\rho_2^2 z_2^{-1} M_2(z_1, z_2)$$

$$M_2(Z_1, Z_2) = \frac{z_1^{-1} x_1(z_1, z_2)}{1-2\rho_2 z_2^{-1}\cos w_2 + \rho_2^2 z_2^{-2}}$$

From these definitions it follows that

$$X(Z_1, Z_2) = (a_2 z_2^{-1} + b_2)\, M_2(Z_1, Z_2) \qquad (2\text{-}55)$$

Then

$$\begin{pmatrix} M_1(k,\ell) \\ M_2(k,\ell) \end{pmatrix} = \begin{pmatrix} 0 & -\rho_2^2 \\ 1 & 2\rho_2\cos w_2 \end{pmatrix} \begin{pmatrix} M_1(k-1,\ell) \\ M_2(k-1,\ell) \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} x_1(k-1,\ell)$$

$$X(k,\ell) = -\rho_2^2 a_2 M_1(k,\ell) + b_2 M_2(k,\ell) \qquad (2\text{-}56)$$

Combining these all together, the following form is obtained.

$$\begin{pmatrix} M_1(k+1,\ell) \\ M_2(k+1,\ell) \\ N_1(k,\ell+1) \\ N_2(k,\ell+1) \end{pmatrix} = \begin{pmatrix} 0 & -\rho_2^{\,2} & 0 & 0 \\ 1 & 2\rho_2\cos w_2 & -\rho_1^2 a_1 & b_1 \\ 0 & 0 & 0 & -\rho_1^{\,2} \\ 0 & 0 & 1 & 2\rho_1\cos w_1 \end{pmatrix} \begin{pmatrix} M_1(k,\ell) \\ M_2(k,\ell) \\ N_1(k,\ell) \\ N_2(k,\ell) \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} w(k,\ell)$$

and

$$x(k,\ell) = [-\rho_2^{\,2} a_2 \quad b_2 \quad 0 \quad 0\,] \begin{pmatrix} M_1(k,\ell) \\ M_2(k,\ell) \\ N_1(k,\ell) \\ N_2(k,\ell) \end{pmatrix}$$

Two methods of modeling have been discussed. Some of the above examples will be used in later chapters. It should be noted again that the filter response method cannot be used for the case where the autocorrelation function is nonseparable.

III. ADAPTIVE FILTERS

A. NONRECURSIVE FILTER

1. Introduction

Many forms of adaptive filters have been described in the literature, some of which have been shown to be optimal (or suboptimal) in certain applications. The special form of an adaptive nonrecursive filter developed by Widrow [1] is reviewed here to give some insights to the recursive adaptive filter developed in next section.

The filter to be considered here consists of a tapped delay line, variable weights whose input signals are the signals at the delay line taps, a summer to add the weighted signals, and machinery to adjust the weights automatically. The impulse response of such a discrete system is completely controlled by the weights. The adaptation process automatically seeks an optimal filter impulse response by adjusting the weights.

Two kinds of processes take place in the adaptive filter: training and operating. The training (adaptation) process is concerned with adjusting the weights, and the operating process consists in forming output signals by weighting the delay line tap signals, using the weights resulting from the training process. During the training process, an additional input signal, "the reference (or desired) response," must be supplied to the adaptive filter along with the usual input signals.

This requirement may in some case restrict the use of this particular form of adaptive filter. An example illustrating the use of the desired signal is the case of modelling an unknown system by a discrete adaptive filter as shown in Figure 3-1. Here a discrete input signal x(n) is applied to an unknown system to be modeled. The discrete adaptive model is supplied with an input x(n). The output of the unknown system d(n) is compared with the output y(n) of the adaptive system. This system can self-adapt to minimize the mean square error, (throughout this thesis, the mean square error is chosen as the performance measure), where the error is defined as the difference between the output of the adaptive model and the desired signal (for this problem the desired signal is the output of the unknown system to be modeled).

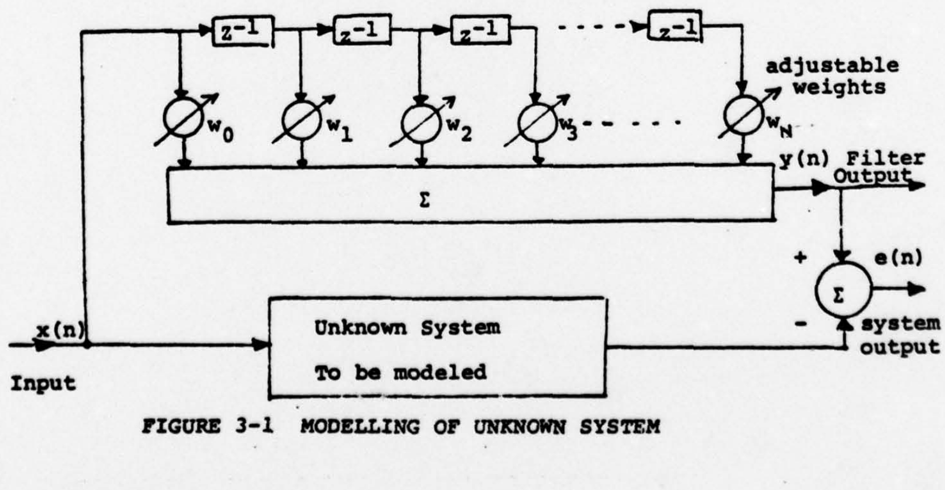


FIGURE 3-1 MODELLING OF UNKNOWN SYSTEM

Then

$$y(n) = \sum_{i=0}^{N} w_i x\,(n-i) \qquad (3\text{-}1)$$

$$e(n) = y(n) - d(n) \qquad (3\text{-}2)$$

Noting that equation (3-1) is the convolution summation, the sequence of weights $\{w_i\}$ can be seen as the impulse response of the adaptive system.

The weights w_i are adjusted to minimize $E(e^2)$.

It will be shown that if the input and output signals of the system being modeled are stationary, the error signal has a mean square value which is a quadratic function of the weight settings.

For the minimization of mean square error, the steepest descent method is used. Throughout this thesis, the terms "filter coefficient updating process" and "adaptation process" are used interchangeably and it is assumed that the input to the adaptive system and the desired signal are stationary random processes (or random fields for the two-dimensional case).

2. Performance Surface, Gradient and the Wiener Solution.

The input signals are weighted and summed to form an output signal [Equation (3-1)]. Introducing the vector notation such that $\overline{W}^T_{(n)} = [w_0, w_1, w_2 \; . \; . \; . \; w_N]$

and $\overline{X}^T_{(n)} = [x(n), x(n-1), x(n-2) \; . \; . \; . \; x(n-N)]$ (3-3)

Then Equation (3-1), which describes the linear combination (operating process), can be written in matrix form.

$$y(n) = \overline{W}^T\overline{X} = \overline{X}^T\overline{W} \qquad (3-4)$$

and $e(n) = y(n) - d(n)$

$$= \overline{W}^T\overline{X} - d(n) \qquad (3-5)$$

The square of this error is

$$e^2(n) = \overline{W}^T\overline{X}\,\overline{X}^T\overline{W} - 2d(n)\,\overline{W}^T\overline{X} + d^2(n) \qquad (3-6)$$

the mean square error, the expected value of $e^2(n)$, is

$$E[e^2(n)] = \overline{d}^2(n) - 2\overline{W}^T R_{xd} + \overline{W}^T R_{xx}\,\overline{W} \qquad (3-7)$$

where the vector of cross-correlation between the input signals and the desired response is defined as

$$E[d(n)\,\overline{X}(n)] = E\begin{bmatrix} d(n)x(n) \\ d(n)x(n-1) \\ \vdots \\ d(n)x(n-N) \end{bmatrix} \overset{\Delta}{=} R_{xd} \qquad (3-8)$$

and where the correlation matrix of the input signal is defined as

$$E[\overline{X}(n)\overline{X}^T(n)] = E\begin{bmatrix} X(n)X(n) & X(n)X(n-1) & \dots & X(n)X(n-N) \\ X(n-1)x(n) & X(n-1)X(n-1) & \dots & X(n-1)X(n-N) \\ & & & \vdots \\ & & & X(n-N)X(n-N) \end{bmatrix} \overset{\Delta}{=} R_{xx} \qquad (3-9)$$

It may be observed from (3-7) that for stationary input signals, the mean-square error is precisely a second order function of the weights. The mean square error performance function may be visualized as a bowl shaped surface, a quadratic function of the weight variables. Then the adaptive process has the job of continually seeking the "bottom of the bowl." A means of accomplishing this by the well-known method of steepest descent is discussed below.

In the non-stationary case, the bottom of the bowl may be moving, and the orientation and curvature of the bowl may be changing. The adaptive process has to track the bottom of the bowl when inputs are nonstationary. The method of steepest descent uses the gradient of the performance surface in seeking its minimum. The gradient at any point on the performance surface may be obtained by differentiating the mean-square error function of Equation (3-5) with respect to the weight vector. The gradient is

$$\nabla\{E(e^2(n))\} = -2\,R_{xd} + 2\,R_{xx}\,\overline{W} \qquad (3\text{-}10)$$

To find the "optimal" weight vector $\overline{W}_{LMS}$, i.e. the one that yields the least mean square error, set the gradient to zero. Accordingly,

$$R_{xd} = R_{xx}\,\overline{W}_{LMS}$$

$$\overline{W}_{LMS} = R_{xx}^{-1}\,R_{xd} \qquad (3\text{-}11)$$

Equation (3-11) is known as the Wiener-Hopf equation in matrix form.

Then the minimum mean square error may be obtained by substituting (3-11) into (3-7)

$$E\,[e^2(n)]\ \min = \overline{d}^2(n) - \overline{W}^T_{LMS}\, R_{xd} \qquad (3\text{-}12)$$

3. LMS Algorithm

In seeking the minimum mean-square error by the method of steepest descent, one begins with an initial guess as to where the minimum point of the mean-square error surface may be. This means that one begins with a set of initial conditions for the weights. The gradient vector is then measured, and the next guess is obtained from the present guess by making a change in the weight vector in the direction of the negative of the gradient vector. The method of steepest descent can thus be described by the following relation

$$\overline{W}(n+1) = \overline{W}(n) + k\nabla[E(e^2(n))] \qquad (3\text{-}13)$$

The expression for $\nabla[E(e^2(n))]$ is obtained by using Equation (3-10).

$$\overline{W}(n+1) = \overline{W}(n) + 2k\,R_{xx}\overline{W} - 2k\,R_{xd} \qquad (3\text{-}14)$$

The gradient vector $\nabla[E(e^2(n))]$ is the gradient of the expectation of the squared error function when the weight vector is $\overline{W}(n)$.

When the performance function is quadratic, the gradient is a linear function of the weights. The advantage of working with the quadratic performance surface lies both in this linear relation and in the fact that such a surface has a unique minimum.

The purpose of the adaptation process is to find an exact or an approximate solution to the Wiener-Hoff equation (3-11). One way of finding the optimum weight vector is simply to solve (3-10). Although this solution is generally straight forward, it could present serious computational problems when the number of weights N is large and when input data rates are high. In addition to the necessity of inverting an N x N matrix, this method may require as many as N(N+1)/2 autocorrelation and cross correlation measurements to be made to obtain the elements of R_{xx}, R_{xd}.

No perfect solution of equation (3-11) is possible in practice to estimate perfectly the elements of the correlation matrices.

A method for finding approximation solutions to (3-11) is presented below. The accuracy of this method is limited by statistical sample size, since weight values are found that are based on finite-time measurements of input-data signals.

This method does not require explicit measurements of correlation functions, nor does it require matrix inversion. It is the "LMS" algorithm based on the steepest descent method. This algorithm does not even require squaring, averaging, or differentiation in order to make use of gradients of mean-square error functions.

When using the LMS algorithm, changes in the weight vector are made along the direction of the estimated gradient vector.

Accordingly

$$\overline{W}(n+1) = \overline{W}(n) + k\hat{\nabla}\{[E(e^2(n))]\} \qquad (3\text{-}13)$$

Where

$\overline{W}(n) \triangleq$ weight vector before adaptation

$\overline{W}(n+1) \triangleq$ weight vector after adaptation

$k \triangleq$ scalar constant controlling rate of convergence and stability $(k<o)$

$\hat{\nabla}[E(e^2(n))] =$ estimate of gradient of $E[e^2(n)]$ with respect to $\overline{W}$ with $\overline{W} = \overline{W}(n)$

One method for obtaining the estimated gradient of the mean square error function is to take the gradient of a single time sample of the squared error; that is

$$\hat{\nabla}[E(e^2(n)] = \nabla[e^2(n)] = 2e(n)\ \nabla[e(n)] \qquad (3\text{-}14)$$

From Equation (3-4)

$$\nabla[e(n)] = \nabla[y(n)-d(n)] = \nabla[\overline{W}^T(n)\overline{X}(n)-d(n)]$$
$$= \overline{X}(n) \qquad (3\text{-}15)$$

Thus,

$$\hat{\nabla}[E(e^2(n))] = 2e(n)\overline{X}(n)$$
$$= 2[\overline{W}^T(n)\overline{X}(n)-d(n)]\overline{X}(n) \qquad (3\text{-}16)$$

The gradient estimate of (3-16) is unbiased, as will be shown by the following argument: For a given weight vector $\overline{W}(n)$, the expected value of the gradient estimate is:

$$E\{\hat{\nabla}\ [E(e^2(n))]\} = 2E\{[\overline{W}^T(n)\overline{X}(n)-d(n)]\overline{X}(n)\}$$
$$= 2R_{xx}\overline{W} - 2R_{xd} \qquad (3\text{-}17)$$

Comparing (3-17) and (3-10), we see that

$$E\{ \hat{\nabla}[E(e^2(n))]\} = \nabla E [e^2(n)] \quad (3\text{-}18)$$

and therefore for a given weight vector, the gradient estimate $\hat{\nabla}[E(e^2(n))]$ is unbiased.

Using the gradient estimation formula (3-16), the weight iteration rule Equation (3-13) becomes

$$\overline{W}(n+1) = \overline{W}(n) + 2k\, e(n)\, \overline{X}(n) \quad (3\text{-}19)$$

and the next weight vector is obtained by adding to the present weight vector scaled by the value of error. This is the LMS algorithm. Looking at Equation (3-19), the adaptation process is a simple first order recursion equation which can be realized as shown below.

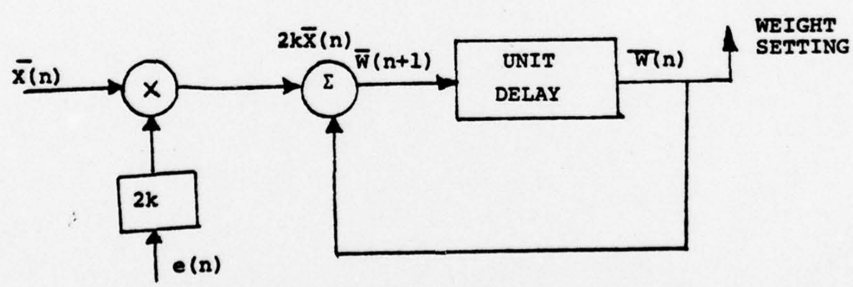


FIGURE - 3-2 FILTER COEFFICIENT UPDATING

To this point, the basic concept of LMS nonrecursive adaptive filter has been introduced and reviewed. More details can be seen in [1]. Widrow [1] showed that the weight vector mean converges to the Wiener solution and that the bounds of the step size k should be in the region such that

$$-\frac{1}{\lambda_{max}} < k < o \quad \text{for the stability and convergence,}$$

where λ_{max} is the maximum eigenvalue of R_{xx}.

4. Two-dimensional adaptive filter.

a. Adaptive filter structure

The input-output relation of the two-dimensional filter is given by two-dimensional convolution.

$$y(k,\ell) = \sum_{i=o}^{p} \sum_{j=o}^{q} w_{ij}\, x(k-i,\ \ell-j), \qquad (3\text{-}20)$$

where $y(k, \ell)$ is the filter output

and W_{ij} is the finite impulse response of filter.

Here, it is assumed that the input $X(k,\ell)$ is a stationary random field.

In Equation (3-20), a set of two-dimensional stationary input signals is weighted and summed to form an output signal and the filter output is intended to match a desired (reference) signal in accordance with the minimization of mean squared error, where the error is the difference between filter output and desired signal.

$$e(k,\ell) \stackrel{\Delta}{=} y(k,\ell) - d(k,\ell). \qquad (3\text{-}21)$$

Introducing the vector notation such that

$$\overline{W}^T = [W_{oo}\ W_{o1}\ .\ .\ .\ W_{oq}\ W_{10}\ W_{11}\ \cdots\ W_{1q}\ W_{20}\ \cdots\cdots W_{pq}]$$

and

$$\overline{X}^T = [x(k,\ell),\ x(k,\ell-1),\ \ldots x(k,\ell-q) x(k-1,\ell) x(k-1,\ell-1) \ldots x(k-1,\ell-q)$$

$$x(k-2,\ell)\ .\ .\ .\ .\ .\ x(k-p,\ell\ -q)] \qquad (3-22)$$

then Equation(3-20)can be written in matrix form.

$$y(k,\ell) = \overline{W}^T\overline{X} = \overline{X}^T\overline{W} \qquad (3-23)$$

where $\overline{W}$ is a weight vector of dimension (p+1)(q+1)x1

$\overline{X}$ is a input signal vector of dimension (p+1)(q+1)x1

The weight vector of the filter is supposed to be adjusted in the direction such that performance criterion (mean square error) is to be minimized. Thus, the linear combinatorial system in Equation (3-20) will be given with variable weights.

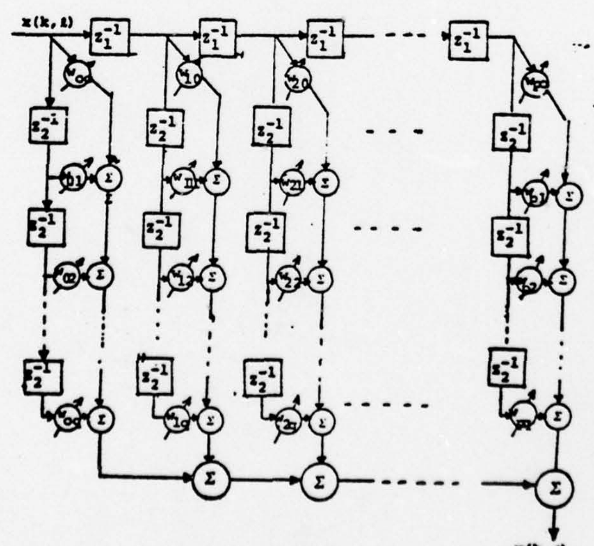


FIGURE 3-3 LINEAR COMBINATORIAL SYSTEM IN TWO-DIMENSIONAL ADAPTIVE FILTER

In Figure 3-3, the linear combinatorial structure is given. Then, the adaptive nonrecursive filter can be drawn as following:

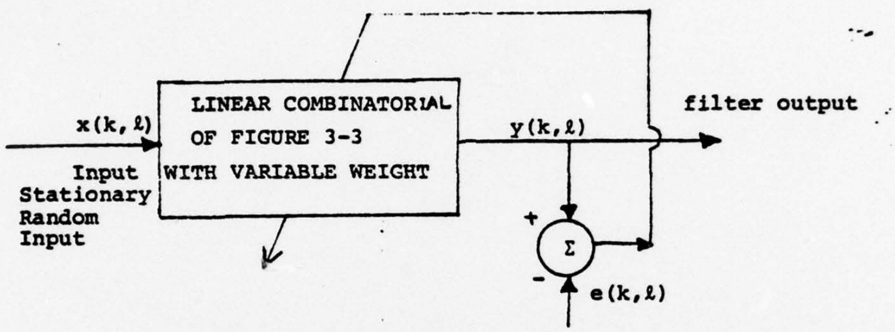


FIGURE 3-4 STRUCTURE OF NONRECURSIVE ADAPTIVE FILTER

b. Wiener solution

From equation (3-22) and (3-23), the error signal can be written by

$$e(k,\ell) = \overline{W}^T\overline{X} - d(k,\ell) \qquad (3\text{-}24)$$

The square of this error is

$$e^2(k,\ell) = \overline{W}^T\overline{X}\overline{X}^T\overline{W} - 2d(k,\ell)\ \overline{X}^T\overline{W} + d^2(k,\ell) \qquad (3\text{-}25)$$

The expected value of $e^2(k,\ell)$ is

$$E[e^2(k,\ell)]=E[d^2(k,\ell)]-2R_{xd}\overline{W}^T+\overline{W}^T R_{xx}\overline{W} \qquad (3\text{-}26)$$

where the vector of cross correlations between the input signal and desired response is defined as

$$E[d(k,\ell)\bar{X}] = E\begin{bmatrix} d(k,\ell)x(k,\ell) \\ d(k,\ell)x(k,\ell-1) \\ \cdot \quad \cdot \\ \cdot \quad \cdot \\ \cdot \quad \cdot \\ d(k,\ell)x(k,\ell-q) \\ d(k,\ell)x(k-1,\ell) \\ d(k,\ell)x(k-1,\ell-q) \\ \cdot \quad \cdot \\ \cdot \quad \cdot \\ \cdot \quad \cdot \\ d(k,\ell)x(k-p,\ell-q) \end{bmatrix} \triangleq R_{xd} \tag{3-28}$$

and where the correlation matrix of the input signals is defined as

$E[\overline{XX}^T] =$

$$E\begin{bmatrix} x_1x_1 & x_1x_2 & x_1x_3 \cdots\cdots & x_1x_{(p+q)(q+1)} \\ x_2x_1 & x_2x_2 & x_2x_3 \cdots\cdots & x_2x_{(p+1)(q+1)} \\ x_3x_1 & x_3x_2 & x_3x_3 \cdots\cdots & x_3x_{(p+1)(q+1)} \\ \cdot & \cdot & \cdot & \cdot \quad \cdot \quad \cdot \\ \cdot & \cdot & \cdot & \cdot \quad \cdot \quad \cdot \\ \cdot & \cdot & \cdot & \cdot \quad \cdot \quad \cdot \\ x_{(p+1)(q+1)}x_1 & x_{(p+1)(q+1)}x_2 & \cdots & x_{(p+1)(q+1)}x_{(p+1)(q+1)} \end{bmatrix} \tag{3-29}$$

$= R_{xx}$, where $x_{(p+1)(q+1)} = x(k-p,\ell-q)$

The gradient at any point on the performance surface can be obtained by differentiating the mean square error function of equation (6)

$$\nabla[E(e^2(k,\ell))] = \frac{\partial E(e^2(k,\ell)}{\partial \overline{W}}$$

$$= -2R_{xd} + 2R_{xx}\overline{W}$$

To find the "optimal" weight vector $\overline{W}_{LMS}$ that yield the least mean square error, set the gradient to zero. Accordingly,

$$R_{xd} = R_{xx}\overline{W}$$

$$\overline{W}_{LMS} = R_{xx}^{-1} R_{xd} \qquad (3\text{-}30)$$

Equation (7) is the Wiener Hopf equation in matrix form, again the minimum mean square error is obtained by substituting (3-30) into (3-26).

$$E[e^2(k,\ell)]_{min} = E[d^2(k,\ell)] - \overline{W}_{LMS}^T R_{xd} \qquad (3\text{-}31)$$

c. LMS Algorithm

Consider a two-dimensional field $x(k,\ell)$ to be processed (usually two-dimensional filters are used in processing discrete two-dimensional image fields) and assume that the two-dimensional field consists of NxN discrete points (which may be a sensed signal by NxN pixel elements of a sensor). The adaptation processes is that of adjusting the filter coefficient W_{ij} in accordance of minimization of the mean square error as in the one-dimensional case. The adaptation scheme may be predetermined as being columnwise scanning or diagonal, or row-wise scanning. Here the row scanning process is adopted as shown in Figure 3-5.

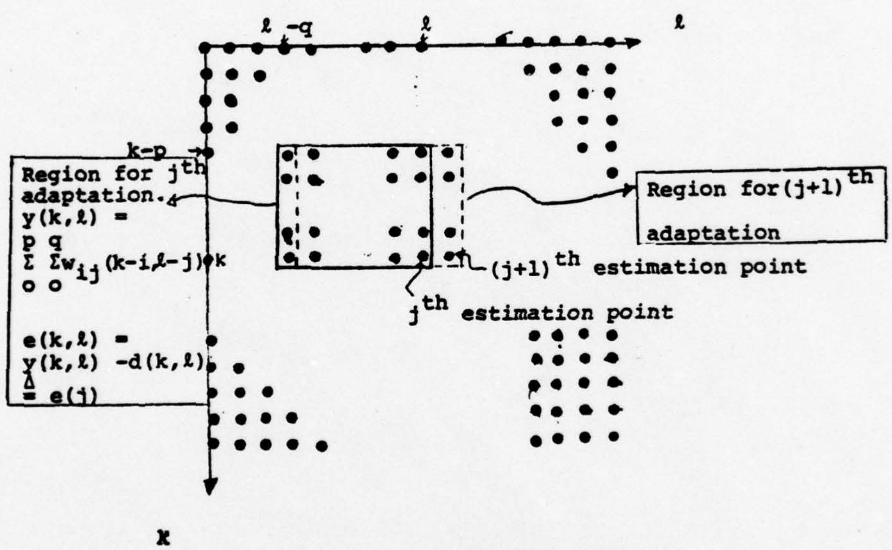


FIGURE 3-5 ADAPTATION SCHEME

Therefore, the NxN adaptation processes will be required to complete the filtering of an NxN two-dimensional field. Using $e(k,\ell)$ to denote the error at the jth iteration (then $e(k\pm 1,\ell)$ is the (j+1)th error), then the filter coefficient updating process can be described by

$$\overline{W}(j+1) = \overline{W}(j) + u\nabla[E(e^2(k,\ell)]$$

where $\overline{W}(j+1) \triangleq$ coefficient vector after adaptation

$\overline{W}(j) \triangleq$ coefficient vector before adaptation

$u \triangleq$ negative scalar constant controlling rate of convergence and stability.

The gradient of the mean square error is to be estimated by

$$\hat{\nabla}[E(e^2(k,\ell)] = \nabla[e^2(k,\ell)]$$

where $e(k,\ell) = y(k,\ell) - d(k,\ell)$

then $\overline{W}(j+1) = \overline{W}(j) - ue(k,\ell)\nabla[e(k,\ell)]$

$$= \overline{W}(j) - 2ue(k,\ell)\overline{X}$$

where $\overline{X}$ is defined by equation (3-22).

Along with $y(k,\ell) = \overline{W}^T\overline{X}$

and $e(k.\ell) = y(k,\ell) - d(k,\ell)$,

the LMS algorithm will be completed.

B. RECURSIVE FILTER

1. Introduction

In the previous section, it is shown that adaptive non-recursive filters have a finite impulse response; that is, they can produce only zeros with no poles in the filter transfer function. This limits the capability of transversal adaptive filters in many applications. To overcome this limitation a new adaptive filter structure is described which is capable of producing poles in the transfer function. The basic configuration considered here is quite standard; that is, the present output sample of the filter y(n) is a linear combination of the present and past samples of the input x(n), x(n-1), . .x(n-M) and the past samples of the output, y(n-1), y(n-2). . . y(n-N). The present output sample of the filter is compared against a reference sample. The resulting error samples are used to adjust the filter parameters, feed forward gains and feed back gains to minimize some error function. The one-dimensional recursive filter is developed first, then it is extended to the two-dimensional adaptive filter.

Recently Feintuck [2] and White [3] have proposed a technique for making digital filters with zeros and poles adaptive. This development may enhance the possiblity of obtaining accurate models for unknown systems. The new approach is developed into an algorithm. It employs the steepest-descent criterion for parameter adjustment but it differs in the estimation of mean squared error gradient vector from Feintuck [2] and Widrow [1].

2. <u>One-Dimensional Adaptive Recursive Filter</u>

a. Structure

The recursive filter is described by its transfer function

$$\frac{Y(Z)}{X(Z)} = \frac{\sum_{i=o}^{M} b_i Z^{-i}}{1 + \sum_{i=1}^{N} a_i Z^{-i}} \qquad (3\text{-}32)$$

In the time domain, the input-output relation of the digital filter is given by

$$y(n) = \sum_{i=o}^{M} b_i x(n-i) - \sum_{i=1}^{N} a_i y(n-i) \qquad (3\text{-}33)$$

where $y(n)$ = nth sample of the filter output

$x(n)$ = nth sample of the filter input

a_i = feedback coefficients $i = 1,2,.\ N$

b_i = feed forward coefficients $i=0,1,2..N$

The output samples of the filter are intended to match those of a reference (or desired) signal $d(n)$. In accordance with the minimization of some error criterion, the filter parameters a_i, b_i will be adjusted at every iteration. The general scheme of the adaptive recursive structure is given in Figure 3-6. The two finite length transversal filters are used in the forward path and feedback path to form the recursive filter of Equation (3-33).

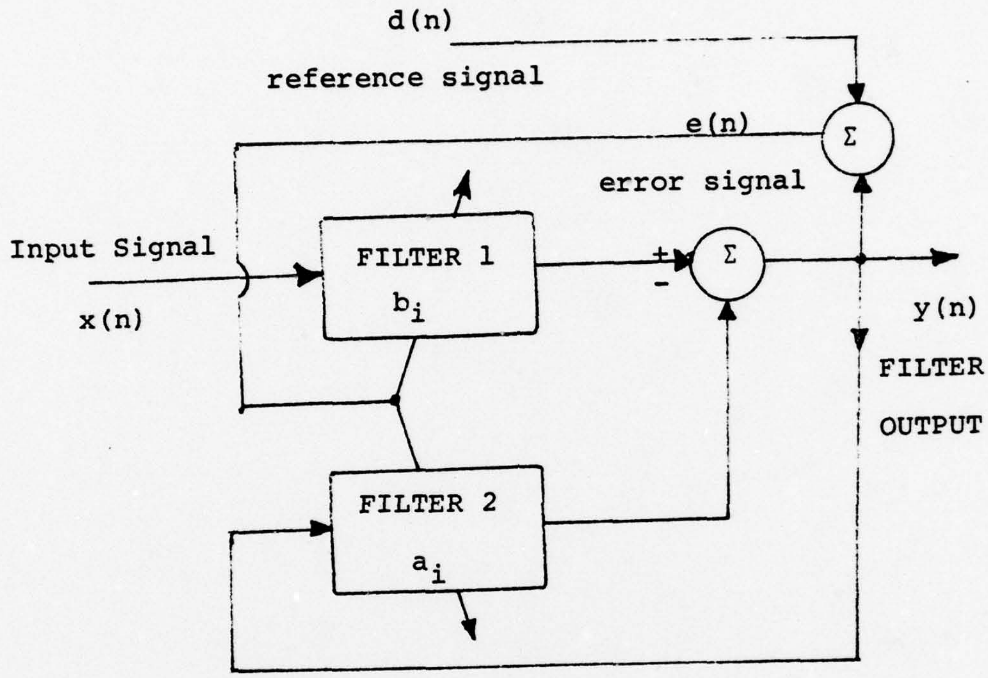


FIGURE 3-6 ADAPTIVE RECURSIVE LMS FILTER USING TWO TRANSVERSAL ADAPTIVE FILTERS

b. Problem in Wiener Solution

Introducing vector notation for the signals and sets of filter coefficients we have

$\bar{A}^T = [\, a_1, a_2, \ldots . \, aN]$

$\bar{B}^T = [\, b_o, b_1, \ldots . \, b_M]$

$\bar{X}(n)^T = [\, x(n), x(n-1) \, . \ldots \, x(n-M)]$

$\bar{Y}(n)^T = [\, y(n-1), y(n-2) \, . \, . \, . \, y(n-N)]$

Then Equation (3-33) can be written as

$$y(n) = \overline{B}^T\overline{X}(n) - \overline{A}^T\,\overline{Y}(n) \qquad (3\text{-}34)$$

where $\overline{A}$ is the feedback coefficient vector (nxl)

$\overline{B}$ is the feed forward coefficient vector[(M+l)xl]

$\overline{X}(n)$ is the input signal vector at nth iteration [(M+lxl]

$\overline{Y}(n)$ is the output signal vector at nth iteration (Nxl)

The performance criterion is again minimum mean squared error, where the error is the difference between filter output and desired signal (reference signal).

That is, the filter is used to estimate a desired waveform d(n) in a minimum mean square error sense. Assume that the observables are stationary and zero mean and let e(n) denote the error waveform at nth sample, then

$$\begin{aligned} e(n) &\triangleq y(n)-d(n) \qquad (3\text{-}35)\\ &= \overline{B}^T\overline{X}(n)-\overline{A}^T\overline{Y}(n) - d(n) \end{aligned}$$

and the mean square error is

$$\begin{aligned} E[e^2(n)] &= E[(\overline{B}^T\overline{X}(n)-\overline{A}^T\overline{Y}(n)-d(n)\ )^2] \\ &= E[\overline{B}^T\overline{X}(n)\overline{X}(n)^T\overline{B}-2\overline{B}^T\overline{X}(n)\overline{Y}^T(n)\overline{A}+\overline{A}^T\overline{Y}(n)\overline{Y}^T\overline{A} \\ &\quad -2\overline{B}^T d(n)\overline{X}(n)+2A^T d(n)\overline{Y}(n)+d^2(n)] \\ &= E[d^2(n)]+\overline{B}^T R_{xx}\overline{B}+\overline{A}^T R_{yy}\overline{A}-2\overline{B}^T R_{xy}\overline{A} \\ &\quad -2\overline{B}^T R_{dx}+2\overline{A}^T R_{dy} \qquad (3\text{-}36) \end{aligned}$$

where $R_{xx} = E[\overline{X}(n)\overline{X}^T(n)]$

$R_{yy} = E[\overline{Y}(n)\overline{Y}^T(n)]$

$R_{dx} = E[d(n)\overline{X}(n)]$

$R_{dy} = E[d(n)\overline{Y}(n)]$

and $R_{ry} = E[\overline{X}(n)\overline{Y}^T(n)]$

The theory of Wiener filtering employs known second-order input statistics to dictate the impulse response of the linear filter that minimizes the mean square error; that is, as in the previous section, the knowledge of second order statistics R_{dx}, R_{xx} is assumed to calculate the optimum impulse response (optimum weight vector in nonrecursive adaptive filter) W_n . But here in the recursive algorithm, it is also required that the autocorrelation of the output, R_{yy}, and the cross correlation of the output and the input, R_{xy}, and the cross correlation of the output with the desired waveform, R_{dy}, should be assumed known. Thus, the set of statistics mentioned above is assumed to be known for a moment, and will be used to determine the weights in the recursive filter. The statistics for the fixed parameter filter are not a function of these statistics, but instead the weights are a function of these statistics. Therefore, R_{xy}, R_{dx} and R_{yy} are to be considered constant when the differentiation is made with respect to $\bar{A}$ and $\bar{B}$.

The set of weights (filter coefficient vectors) which minimize the mean square error can be found by getting the gradient vector with respect to filter parameters equal to zero.

$$\frac{\partial E[e^2(n)]}{\partial \bar{A}} = 2R_{yy}\bar{A} - 2\,R_{xy}\bar{B} + 2\,R_{dy} = 0$$

$$\bar{A} = R_{yy}^{-1}\,(R_{dy} - R_{xy}\,\bar{B}) \qquad (3\text{-}37)$$

and

$$\frac{\partial\, E[e^2(n)]}{\partial\, \bar{B}} \triangleq \nabla_B[E(e^2(n))]$$

$$= 2R_{xx}\bar{B} - 2R_{xy}\bar{A} - 2\,R_{dx}$$

$$\bar{B} = R_{xx}^{-1}\,(R_{dx} + R_{xy}\bar{A}) \qquad (3\text{-}38)$$

Thus, one can solve for the filter coefficients if all the second order statistics are known. But without knowing the impulse response of the filter, the R_{xy}, R_{dy} and R_{yy} can not be calculated with only input and reference signal statistics. Noting that we are looking for the impulse response which minimizes the mean square error in some way, it is clear that R_{xy}, R_{dy}, R_{yy} are not available and so the Wiener approach is not feasible.

C. LMS ALGORITHM

An iterative gradient search technique (the method of steepest descent) is revisited. Here, in the recursive algorithm, it updates the filter coefficients with steps proportional to the gradient vector. This updating process is

$$\begin{aligned} \bar{A}(n+1) &= \bar{A}(n) + k_a \Delta \bar{A} \\ &= \bar{A}(n) + k_a \nabla_A [E(e^2(n))] \\ \bar{B}(n+1) &= \bar{B}(n) + k_b \Delta \bar{B} \\ &= \bar{B}(n) + k_b \nabla_B [E(e^2(n)] \end{aligned} \quad (3\text{-}39)$$

where

$\bar{A}(n)$, $\bar{B}(n)$ $\triangleq$ filter coefficient vectors before adaptation

$\bar{A}(n+1)$, $\bar{B}(n+1)$ $\triangleq$ filter coefficient vectors after adaptation

k_a, k_b $\triangleq$ scalar constants controlling rate of convergence and stability (k_a, $k_b < o$)

$\nabla_A[E(e^2(n))]$, $\nabla_B[E(e^2(n))]$ $\triangleq$ gradient vectors with respect to $\bar{A}$ and $\bar{B}$ respectively.

The updating process (3-39) can be considered as a first order filtering process with an input proportional to the gradient vector. But the gradients $\nabla_A[E(e^2(n))]$ and $\nabla_B[E(e^2(n))]$ should be estimated because the output statistics are not available apriori or an infinite statistical sample would be required to estimate perfectly the elements of the correlation matrices in Equations (3-37) and (3-38). A method of estimating these gradients will be presented.

Widrow [1] obtained the estimated gradient of the mean square error function by taking the gradient of a single time sample of the squared error (instantaneous estimates) when he discussed the nonrecursive adaptive filter (see previous section).

Here, in this thesis work, a new method of estimating gradients is proposed. This is to approximate the mean squared error $E(e^2(n)]$ by an average of a finite number of points at every iteration and take the gradient of this instead of taking the instantaneous error square, that is, the approximation used is

$$E(e^2(n)) \approx \frac{1}{L} \sum_{\ell=0}^{L-1} e^2(n-\ell) \qquad (3\text{-}40)$$

For $E(e^2(n))$, the average of the square error for the previous L points is taken and then gradient is evaluated for the approximate mean square error. The estimated gradient of mean square error is

$$\hat{\nabla}_A E[e^2(n)] = \nabla_A\left[\sum_{\ell=0}^{L-1} e^2(n-\ell)\right]$$

$$\hat{\nabla}_B E[e^2(n)] = \nabla_B\left[\sum_{\ell=0}^{L-1} e^2(n-\ell)\right] \qquad (3\text{-}41)$$

For convenience, the $(\frac{1}{L})$ term in (3-40) was dropped. Introducing the vector notation for error signal

$$\overline{\varepsilon}(n) = \begin{bmatrix} e(n) \\ e(n-1) \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ e(n-L+1) \end{bmatrix}$$

then it is seen that the error signal vector is an (Lxl) vector. The estimated gradient (Equation (3-41)) can be put into the matrix form

$$\hat{\nabla}_A E[e^2(n)] = \nabla_A[\overline{\varepsilon}^T(n)\ \overline{\varepsilon}(n)]$$

$$\hat{\nabla}_B E[e^2(n)] = \nabla_B[\overline{\varepsilon}^T(n)\ \overline{\varepsilon}(n)]$$

Substituting the estimated gradients in Equation (3-39), the updating process for the filter coefficients is:

$$\overline{A}(n+1) = \overline{A}(n) + K_a \nabla_A[\overline{\varepsilon}^T(n)\overline{\varepsilon}(n)]$$

and

$$\overline{B}(n+1) = \overline{B}(n) + K_b \nabla_B[\overline{\varepsilon}^T(n)\overline{\varepsilon}(n)] \qquad (3\text{-}42)$$

The function $\overline{\varepsilon}^T(n)\ \overline{\varepsilon}(n)$ is a scalar function of the coefficient vectors $\overline{A}$ and $\overline{B}$, that is,

$$\overline{\varepsilon}^T(n)\overline{\varepsilon}(n) = f(\overline{A}, \overline{B})$$

Therefore, by definition, the gradient of $f(\overline{A}, \overline{B})$ with respect to $\overline{A}$ and $\overline{B}$ is

$$\frac{\partial f}{\partial \overline{A}} = \begin{bmatrix} \frac{\partial f}{\partial a_1} \\ \frac{\partial f}{\partial a_2} \\ \vdots \\ \frac{\partial f}{\partial a_N} \end{bmatrix} \qquad \frac{\partial f}{\partial \overline{B}} = \begin{bmatrix} \frac{\partial f}{\partial b_o} \\ \frac{\partial f}{\partial b_1} \\ \vdots \\ \frac{\partial f}{\partial b_M} \end{bmatrix}$$

It follows that

$$\hat{\nabla}_A[E(e^2(n))] = \nabla_A[\overline{\varepsilon}^T(n)\overline{\varepsilon}(n)]$$

$$= \begin{bmatrix} \frac{\partial(\overline{\varepsilon}^T(n)\overline{\varepsilon}(n))}{\partial a_1} \\ \frac{\partial(\overline{\varepsilon}^T(n)\overline{\varepsilon}(n))}{\partial a_2} \\ \vdots \\ \frac{\partial(\overline{\varepsilon}^T(n)\overline{\varepsilon}(n))}{\partial a_N} \end{bmatrix} = \begin{bmatrix} 2\overline{\varepsilon}^T(n) \frac{\partial \overline{\varepsilon}(n)}{a_1} \\ 2\overline{\varepsilon}^T(n) \frac{\partial \overline{\varepsilon}(n)}{\partial a_2} \\ \vdots \\ 2\overline{\varepsilon}^T(n) \frac{\partial \overline{\varepsilon}(n)}{\partial a_N} \end{bmatrix} \tag{3-43}$$

and

$$\hat{\nabla}_B(E(e^2(n))) = \nabla_B(\overline{\varepsilon}(n)^T\overline{\varepsilon}(n))$$

$$= \begin{bmatrix} \frac{\partial(\overline{\varepsilon}(n)^T\overline{\varepsilon}(n))}{\partial b_o} \\ \frac{\partial(\overline{\varepsilon}(n)^T\overline{\varepsilon}(n))}{\partial b_1} \\ \vdots \\ \frac{\partial(\overline{\varepsilon}(n)^T\overline{\varepsilon}(n))}{\partial b_N} \end{bmatrix} = \begin{bmatrix} 2\,\overline{\varepsilon}(n)^T \frac{\partial \overline{\varepsilon}(n)}{\partial b_o} \\ 2\,\overline{\varepsilon}(n)^T \frac{\partial \overline{\varepsilon}(n)}{\partial b_1} \\ \vdots \\ 2\overline{\varepsilon}(n)^T \frac{\partial \overline{\varepsilon}(n)}{\partial b_N} \end{bmatrix} \tag{3-44}$$

Consider the terms $\frac{\partial \bar{\varepsilon}(n)}{\partial a_p}$, $\frac{\partial \bar{\varepsilon}(n)}{\partial b_q}$ in equation (3-43),(3-44), where p=1,2,...N and q = 0,1,2,...M.

Since $\bar{\varepsilon}(n)$ and e(n) are defined as

$$e(n) \triangleq y(n) - d(n)$$

$$\bar{\varepsilon}(n)^T = (e(n),\ e(n-1),\ \ldots\ e(n-L+1),$$

$$\frac{\partial \bar{\varepsilon}(n)}{\partial a_p} = \begin{pmatrix} \frac{\partial e(n)}{\partial a_p} \\ \frac{\partial e(n-1)}{\partial a_p} \\ \vdots \\ \frac{\partial e(n-L+1)}{\partial a_p} \end{pmatrix} = \begin{pmatrix} \frac{\partial y(n)}{\partial a_p} \\ \frac{\partial y(n-1)}{\partial a_p} \\ \vdots \\ \frac{\partial y(n-L+1)}{\partial a_p} \end{pmatrix} \quad p=1,2,\ldots N \qquad (3\text{-}45)$$

and

$$\frac{\partial \bar{\varepsilon}(n)}{\partial b_q} = \begin{pmatrix} \frac{\partial e(n)}{\partial bq} \\ \frac{\partial e(n+1)}{\partial bq} \\ \vdots \\ \frac{\partial e(n-L+1)}{\partial bq} \end{pmatrix} = \begin{pmatrix} \frac{\partial y(n)}{\partial bq} \\ \frac{\partial y(n-1)}{\partial bq} \\ \vdots \\ \frac{\partial y(n-L+1)}{\partial bq} \end{pmatrix} \quad q=0,1,2,\ldots M \qquad (3\text{-}46)$$

Note that $\frac{\partial \bar{\varepsilon}(n)}{\partial ap}$ and $\frac{\partial \bar{\varepsilon}(n)}{\partial bq}$ are (Lx1) vectors and $\bar{\varepsilon}(n)^T$ is an (1xL) vector.

Therefore, $\hat{\nabla}_A[E(e^2(n))]$ and $\hat{\nabla}_B[E(e^2(n))]$ in Equation (3-41) are (Nxl) and [(M+1)xl] vectors respectively.

Equations (3-45) and (3-46) may be considered as sensitivity vectors which tell how much the change in a_p and bq affect the outputs y(n), y(n-1) . . . y(n-L+1).

To calculate the elements of the estimated gradients of mean square error in equation (3-41), we should calculate the sensitivity vector of equation (3-45) and (3-46) first.

From the recursive equation (3-33),

$$\frac{\partial y(n)}{\partial a_p} = \frac{\partial}{\partial a_p}\left[\sum_{i=o}^{M} b_i X(n-i) - \sum_{i=1}^{N} A_i y(n-i)\right]$$

$$= - y(n-p) - \sum_{i=1}^{N} A_i \frac{\partial y(n-i)}{\partial a_p} \qquad (3-47)$$

$$p = 1, 2, \ldots N$$

$$\frac{\partial y(n)}{\partial b_q} = x(n-q) - \sum_{j=1}^{N} a_i \frac{\partial y(m-i)}{\partial bq}$$

$$q = 0, 1, 2, \ldots M \qquad (3-48)$$

The sensitivity vector components given by Equations (3-47) and (3-48) can be interpreted as being the response of a linear system with transfer function.

$$H(Z) = \frac{1}{1+a_1 z^{-1} + a_2 z^{-2} + \ldots a_N z^{-N}} \qquad (3-49)$$

Henceforth, this will be called the "sensitivity filter." Equation (3-49) is an all pole filter (recursive filter) with input signals [-y(n-p)] and [x(n-q)] respectively.

Now, what are the initial conditions characterizing the recursive relationships of the sensitivity filter?

$x(n-q)$ is the q time units delayed signal of input $x(n)$ to the adaptive recursive filter of Equation (3-33) and $y(n-p)$ is the p time units delayed signal of the output $y(n)$ of the recursive filter.

From $x(n) = o$

$y(n) = o$ for $n < o$,

$x(n-q)$ and $y(n-p)$ are sequences with the first q elements and first p elements zero respectively. And since changes in the ap and bq coefficients have no effects on the system's response until n=p and n=q respectively, it follows that the initial conditions are:

$$\frac{\partial y(n)}{\partial ap} = o \quad \text{for } n=o, 1 \ldots p-1$$

$$\frac{\partial y(n)}{\partial bq} = o \quad \text{for } n=o, 1 \ldots q-1$$

A summary of this algorithm is the following:

1. Calculate the sensitivity vector components through the sensitivity recursive filter by equations (3-47) and (3-48).
2. Calculate the estimated gradient by equation (3-43) and (3-44).
3. Calculate the filter coefficient vector by equation (3-42).
4. Calculate the filter output by equation (3-34)
5. Form the e(n) vector $\overline{\varepsilon}(n)$, then go back to the 1st step.

Note that due to the fact that the gradient of finite point square error average is used for the estimation of the true gradient of mean square, this filter cannot give an optimal solution, but the more averaging points are used, the better performance is expected. It should be noted that if L=1, that is,

$$\hat{\nabla}_A[E(e^2(n))] = \nabla_A[e^2(n)]$$
$$\hat{\nabla}_B[E(e^2(n))] = \nabla_B[e^2(n)] \ ,$$

this corresponds to using the instantaneous error square for estimating the gradient, and this filter reduces to the adaptive recursive filter proposed by White [3]. If the further approximation is made that the sensitivity components of equation (3-47), (3-48) are

$$\frac{\partial y(n)}{\partial ap} = -y(n-p)$$
$$\frac{\partial y(n)}{\partial bq} = x(n-q) \ ,$$

then the estimated gradient is

$$\hat{\nabla}_A[E(e^2(n))] = -2\, e(n) \begin{pmatrix} y(n-1) \\ y(n-2) \\ \cdot\ \cdot \\ \cdot\ \cdot \\ y(n-N) \end{pmatrix} = -2e(n)\overline{Y} \qquad (3\text{-}50)$$

and

$$\hat{\nabla}_B[E(e^2(n))] = 2\, e\,(n) \begin{pmatrix} x(n) \\ x(n-1) \\ \cdot\ \cdot \\ \cdot\ \cdot \\ x(n-N) \end{pmatrix} = 2e(n)\overline{X} \qquad (3\text{-}51)$$

Then the filter coefficient updating process is

$$\overline{A}(n+1) = \overline{A}(n) - 2k_a e(n)\overline{Y}$$

$$\overline{B}(n+1) = \overline{B}(n) + 2k_b en)\overline{X} \qquad (3\text{-}52)$$

where k_a, $k_b < 0$ and filter output

$$y(n) = \overline{B}(n)^T\overline{X} - \overline{A}^T(n)\overline{Y} \qquad (3\text{-}53)$$

Equations (3-53), (3-52), (3-51), (3-50) are exactly the same as the algorithm proposed by Feintuck [2]. This Feintuck algorithm has an advantage in simplicity when compared with the algorithms proposed by White [3] and proposed here which require additional recursive filters to generate the estimates of the gradient. Thus, it may be useful to extend the Feintuck algorithm to the two-dimensional recursive adaptive filter for simplicity. In the next section, the algorithms proposed by White and Feintuck are extended to the two-dimensional algorithm.

3. Two-dimensional Recursive Adaptive Filter

In this section, a mathematical model of the adaptive recursive filter for the processing of two-dimensional signals is proposed. This can be considered as an extension of Feintuck's algorithm to two-dimensional filters.

Two transversal filters having the same structure as the linear combinatorial system used in the non-recursive two-dimensional processor, are used in the recursive processor, one for the feedforward path and one for the feedback path.

The two-dimensional recursive filter is described by its transfer function.

$$\frac{Y(z_1, z_2)}{X(z_1, z_2)} = \frac{\sum_{i=o}^{p} \sum_{j=o}^{q} b_{ij} z_1^{-i} z_2^{-j}}{1 + \sum_{\substack{m=o \\ (m,n) \neq (0,0)}}^{M} \sum_{n=0}^{N} a_{mn} z_1^{-m} z_2^{-n}}$$

In the spatial domain, the input-output relation of the digital filter is given by

$$y(k,\ell) = \sum_{i=0}^{p} \sum_{j=0}^{q} b_{ij} x(k-i,\ell-j) - \sum_{n=0}^{M} \sum_{n=0}^{N} a_{mn} y(k-m,\ell-n) \qquad (3\text{-}54)$$

The following notation is introduced

$$\overline{B}^T = [b_{oo} \; b_{01} \; \ldots. b_{oq} \; b_{10} \; b_{11} \; \ldots. b_{1q} \; b_{20} \ldots\ldots. b_{pq}]$$

$$\overline{X}^T = [x(k,\ell), x(k,\ell-1), .. x(k,\ell-q) x(k-1,\ell) x(k-1,\ell-1) \ldots x(k-1,\ell-q)$$
$$x(k-2, \ell \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; x(k-p, \ell-q)]$$

and $\overline{A}^T = [a_{01} \; a_{02} \ldots a_{0N} \; a_{10} \, a_{11} \ldots a_{1N} a_{20} \ldots\ldots a_{MN}]$

$$\overline{Y}^T = [y(k,\ell-1), \; y(k,\ell-2) \ldots y(k,\ell-N) \; y(k-1,\ell) y(k-1,\ell-1) \ldots y(k-1,\ell-N)$$
$$y(k-2, \ell) \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; . \; y(k-M,\ell-N) \;]$$

The filter coefficient vectors $\overline{A}$ and $\overline{B}$ are $[(M+1)(N+1)-1]x1$ and $(p+1)(q+1)x1$, respectively, and the input-output signal vectors are again $(p+1)(q+1)x1$ and $[(M+1)(N+1)-1]x1$, respectively. Then equation (22) can be written as

$$y(k,\ell) = \overline{B}^T\overline{X} - \overline{A}^T\overline{Y} \qquad (3\text{-}55)$$

Here, to obtain an estimate of gradients of the mean square error function, a single sample of the square error is taken. That is: $\hat{\nabla}[E(e^2(k,\ell))] = \nabla[e^2(k,\ell)] = 2e(k,\ell)\nabla[e(k,\ell)]$, and again the adaptation scheme (filter coefficient updating process) is used in the same fashion as in the nonrecursive case [see Figure 3-5].

Denoting the error at jth iteration as $e(k,\ell)$ then

$$\bar{A}(j+1) = \bar{A}(j) + 2k_a e(j) \nabla_A (e(j))]$$

$$\bar{B}(j+1) = \bar{B}(j) + 2k_b e(j) \nabla_A (e(j))] \qquad (3\text{-}56)$$

where

$$e(k,\ell) = e(j) \triangleq y(k,\ell) - d(k,\ell) \qquad (3\text{-}57)$$

The components of the vectors $\nabla_A [e(k,\ell)]$ and $\nabla_B [e(k,\ell)]$ can be calculated as following.

From Equation (3-57)

$$\nabla_A [e(k,\ell)] = \nabla_A [y(k,\ell)]$$

$$= \left[\frac{\partial y(k,\ell)}{\partial a_{01}} \cdot \cdot \frac{\partial y(k,\ell)}{\partial a_{0M}} \frac{\partial y(k,\ell)}{\partial a_{10}} \cdot\cdot \frac{\partial y(k,\ell)}{\partial a_{1M}} \frac{\partial y(k,\ell)}{\partial a_{20}} \cdots\cdots\cdots \frac{\partial y(k,\ell)}{\partial a_{MN}} \right]^T \qquad (3\text{-}58)$$

and $\nabla_B [e(k,\ell)] = \nabla_B [y(k,\ell)]$

$$= \left[\frac{\partial y(k,\ell)}{\partial b_{oo}} \cdots \frac{\partial y(k,\ell)}{\partial b_{oq}} \frac{\partial y(k,\ell)}{\partial b_{10}} \cdots \frac{\partial y(k,\ell)}{\partial b_{1q}} \frac{\partial y(k,\ell)}{\partial b_{20}} \cdots\cdots\cdots \frac{\partial y(k,\ell)}{\partial b_{pq}} \right]^T \qquad (3\text{-}59)$$

Note that $\nabla_A [e(k,\ell)]$ and $\nabla_B [e(k,\ell)]$ have the same dimensions as $\bar{A}$ and $\bar{B}$, that is:

$[(M+1)(N+1)-1] \times 1$ and $[(p+1)(q+1)]$, respectively.

From the recursive relation of Equation (3-54)

$$\frac{\partial y(k,\ell)}{\partial a_{rs}} = -y(k-r,\ell-s) - \sum_{m=o}^{M} \sum_{n=o}^{N} a_{mn} \frac{\partial y(k-m,\ell-n)}{\partial a_{rs}} \qquad (3\text{-}60)$$

and $$\frac{\partial y(k,\ell)}{\partial b_{uv}} = x(k-u,y-v) - \sum_{\substack{m=o \\ (m,n)\neq(0,0)}}^{M} \sum_{n=o}^{N} a_{mn} \frac{\partial y(k-m,\ell-n)}{\partial b_{uv}} \qquad (3\text{-}61)$$

The recursive relationship of Equations (3-60) and (3-61) should be noted again, which can be implemented by additional recursive filters.

Forming the instantaneous error gradient of Equations (3-54), (3-55) using the output of additional recursive filters of Equations (3-60) (3-61), the filter coefficient adaptation process of Equations (3-56) (3-57) can be performed. Note that this algorithm corresponds to the two-dimensional version of the algorithm proposed by White [3].

If we make the approximation

$$\frac{\partial y(k,\ell)}{\partial a_{rs}} = - y(k-r,\ell-s)$$

$$\frac{\partial y(k,\ell)}{\partial b_{uv}} = x(k-u, y-v),$$

then it follows that

$$\nabla_A[e(k,\ell)] = \begin{pmatrix} y(k,\ell-1) \\ y(k,\ell-2) \\ \cdot\ \cdot \\ \cdot\ \cdot \\ y(k-1,\ell) \\ y(k-1,\ell-1) \\ \cdot\ \cdot \\ \cdot\ \cdot \\ y(k-2,\ell) \\ \cdot\ \cdot \\ \cdot\ \cdot \\ y(k-M,\ell-N) \end{pmatrix} \triangleq \overline{Y}$$

and $$\nabla_B[e(k,\ell)] = \begin{pmatrix} x(k,\ell) \\ x(k,\ell-1) \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ x(k,\ell-q) \\ x(k-1,\ell) \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ x(k-p,\ell-q) \end{pmatrix} \triangleq \overline{X}$$

Therefore, in this case, the complete algorithm is described by

$$e(k,\ell) \triangleq e(j) = y(k,\ell)-d(k,\ell)$$

$$\overline{A}(j+1) = \overline{A}(j)-2k_a e(j)\ \overline{Y}$$

$$\overline{B}(j+1) = \overline{B}(j)+2k_b e(j)\ \overline{X}$$

and $Y(k,\ell) = \overline{B}^T\overline{X} - \overline{AY}$

This is the two-dimensional version of the algorithm proposed by Feintuck [2].

IV. ADAPTIVE NOISE CANCELLER

A. THE CONCEPT OF ADAPTIVE NOISE CANCELLING

Noise cancelling is a variation of optimal filtering that is highly advantageous in many applications. Specially in Wiener filtering or Kalman filtering, which are optimal, apriori knowledge of both signal and noise statistics are required. Adaptive filters, on the other hand, have the ability to adjust their own parameters automatically, and their design requires little or no apriori knowledge of signal and noise statistics while the Wiener approach utilizes a fixed parameter filter based on known statistics.

Figure (4-1) shows the basic problem and the adaptive noise cancelling solution to it. It makes use of a reference input derived from one (or more) sensors located at the points in the noise field where the signal is weak or undetectable. This input is filtered and substracted from a primary input containing both signal and noise. As a result the primary noise is attenuated or eliminated by cancellation.

At first glance, subtracting noise from a signal seems to be a dangerous procedure. If done improperly it could result in an increase in output noise power. If, however, filtering and subtraction are controlled by an appropriate adaptive process, noise reduction can be accomplished with little risk of distorting the signal or increasing the output noise level.

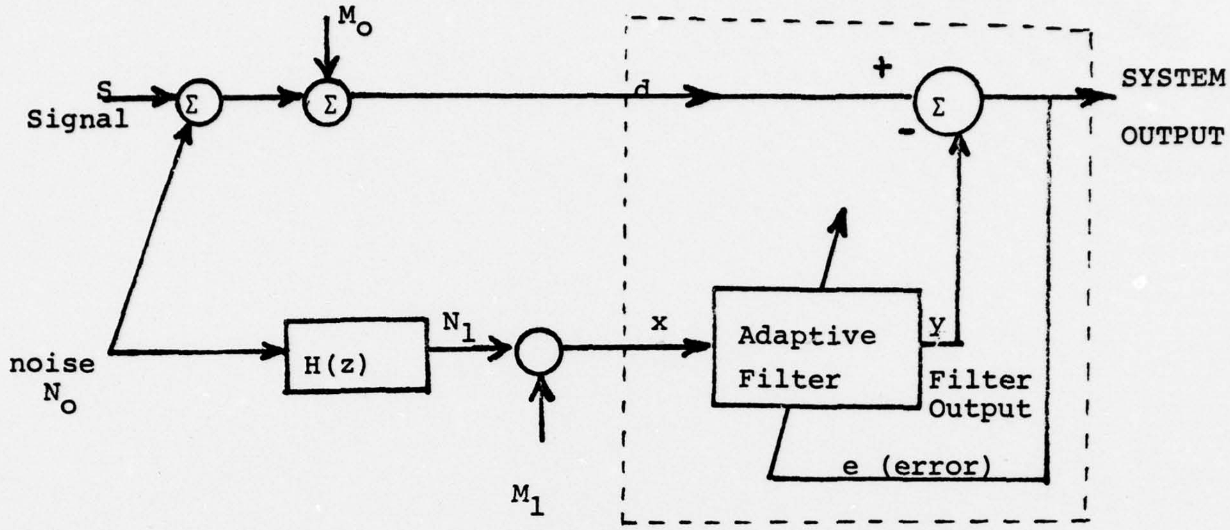


FIGURE 4-1. THE ADAPTIVE NOISE CANCELLING CONCEPT

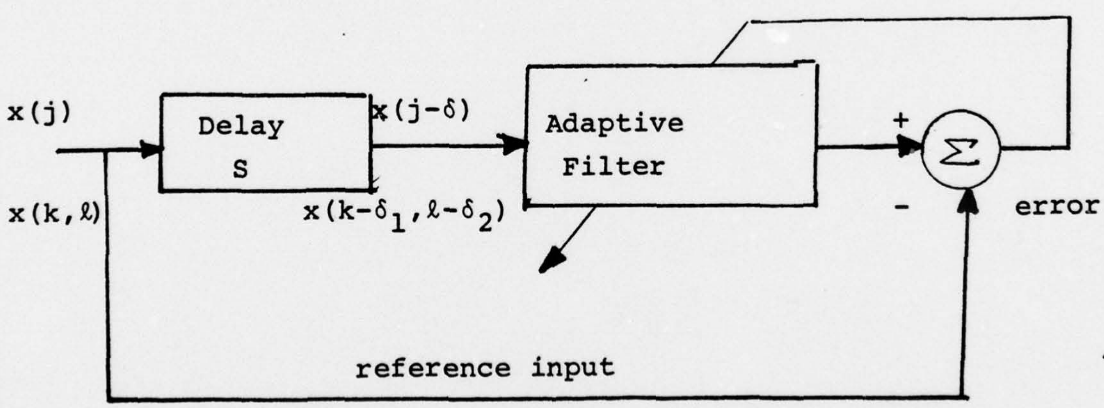


FIGURE 4-2 NOISE CANCELLING WITHOUT AN EXTERNAL REFERENCE SOURCE

The following argument for the above is mainly due to Widrow, et al [4]. In Figure (4-1), a signal S is transmitted over a channel to a sensor that also receives a noise N_o.

A second sensor receives a noise N_1 uncorrelated with the signal but correlated in some unknown way with the noise N_o. In addition to these noises, additive random noises M_o and M_1 uncorrelated with each other and with S, N_o and N_1 are present. Then the reference input is

$$d = s + N_o + M_o \qquad (4\text{-}1)$$

and the primary input

$$x = N_1 + M_1 \qquad (4\text{-}2)$$

The noise $N_1 + M_1$ is filtered to produce an output y that is as close a replica as possible of $N_o + M_o$. This output is subtracted from the reference input $S + N_o + M_o$ to produce the system output

$$z = S + N_o + M_o - y$$

In other words, the practical objective of the noise cancelling system is to produce a system output $z = S + N_o + M_o - y$ that is best fit in the least square sense to the signal S. This objective is accomplished by feeding the system output back to the adaptive filter and adjusting the filter through the LMS adaptive algorithm (described in previous chapter) to minimize total system output power. Note that the system output serves as the error signal for the adaptive process.

Assume for the moment that the noises M_o and M_1 do not exist, then if one knew the characteristics of the channels over which the noise N_o is transmitted to the reference input, it would be possible theoretically to design a fixed filter capable of changing N_1 into N_o. That is, if the correct model of this transmission channel, $H(z)$, is obtained, the adaptive filter would be simply $\frac{1}{H(z)}$, a fixed filter.

Assume that S, N_o, N_1, M_o, M_1, and y are statistically stationary and have zero means. Assume that S is uncorrelated with N_o and N_1 and that M_o and M_1 are uncorrelated with each other and with S, N_o and N_1, and suppose that N_1 is correlated with N_o. The output z is

$$z = S + N_o + M_o - y \qquad (4\text{-}3)$$

squaring, one obtains

$$z^2 = S^2 + (N_o + M_o - y)^2 + 2S\,(N_o + M_o - y) \qquad (4\text{-}4)$$

Taking expectations of both sides and realizing that S is uncorrelated with N_o, M_o and y, yields

$$E[z^2] = E[S^2] + E[(N_o + M_o - y)^2] + 2E[S(N_o + M_o - y)]$$
$$= E[S^2) + E[(N_o + M_o - y)^2] \qquad (4\text{-}5)$$

The signal power $E[S^2]$ will be unaffected as the filter is adjusted to minimize $E[z^2]$. Accordingly, the minimum output power is

$$\min E[z^2] = E[S^2] + \min E[(N_o + M_o - y)^2] \qquad (4\text{-}6)$$

Since the filter is adjusted so that $E(z^2)$ is minimized, therefore $E[(N_o + M_o - y)^2]$ is minimized. The filter output y is then a best least squares estimate of the noise $N_o + M_o$. Moreover, when $E[(N_o + M_o - y)^2]$ is minimized, $E[(z-S)^2]$ is also minimized. Since, from (4-3)

$$z-S = M_o + N_o - y \qquad (4-7)$$

Adjusting or adapting the filter to minimize the total output power is thus equivalent to causing the output z to be a best least square estimate of the signal S for a given structure and adjustability of the adaptive filter and for the given reference input. The output z will contain the signal S plus noise. From (4-3), the output noise is given by $(N_o + M_o - y)$. Since minimizing the $E[z^2]$ minimizes the $E[N_o + M_o - y)^2]$, minimizing the total output power minimizes the output noise power. Since the signal in the output remains constant, minimizing the total output power maximizes the output signal to noise ratio. Note that if $E[(N_o - y)^2] = 0$ can be achieved, then $E[z^2] = E(S^2)$, therefore $y = N_o + M_o$ and $z = S$. In this case, minimizing output power causes the output signal to be perfectly noise free. Also note that, on the other hand, when the reference input is completely uncorrelated with the primary input, the filter will "turn itself off" and will not increase output noise.

In this case, the filter output y will be uncorrelated with the primary input. The output power will be

$$E[z^2] = E[(S+ M_o + N_o)^2] - 2E[y(s+N_o + M_o)] + E[y^2]$$
$$= E[(S + M_o + M_o)^2] + E[y^2] \qquad (4-7)$$

Therefore, minimizing output power requires that $E(y^2)$ be minimized, which is accomplished by making all weights zero, bringing $E[y^2]$ to zero.

It should be noted that in applying adaptive techniques to a practical systems problem, the key step lies in providing an appropriate desired response signal for the adaptation process, that is, the reference input should be provided through the appropriate scheme, while the exact knowledge of statistical characteristicsare not required. In adaptive modeling applications, the desired response is generally available as the output of the unknown system to be modeled. And also in the noise cancelling scheme above, the reference input is available by sensing noise which is correlated with the noise at the primary input in some manner.

In next section, the signal filtering problem is discussed when no external reference input free of signal is available.

B. NOISE CANCELLING WITHOUT AN EXTERNAL REFERENCE INPUT

This section is concerned with signal filtering (estimation) a noise-corrupted signal when no external reference input is available. Here, it is assumed that only the noise-corrupted signal is available, that is, referring to the Figure (4-1) of the previous section, the noise free of signal N_1

which is correlated with N_o that corrupted the signal S is not available; only $S + N_o$ is available.

It is proposed to estimate the signal S by cancelling the noise N_o in some adaptive way. In the following, it is shown how a reference input can be obtained for the adaptation process under certain conditions. Assume that the noise corrupted signal $x = S + N$ is composed of broad band noise N and a narrow band signal S, then the autocorrelation function of the signal is broad and that of the noise is narrow. Also assume that noise N is uncorrelated with the signal and that the mean values of both signal and noise are zero.

Consider a signal delayed by δ units,

$$x(j-\delta) = S\,(j-\delta) + n\,(j-\delta) \qquad (4\text{-}9)$$

where δ is a sufficient number of time units so that the noise component is decorrelated, but the signal component still remains correlated.

Then

$$E[n(j)\,n\,(j-\delta)\,] = 0$$

$$E(S(j)\,S\,(j-\delta)\,] \neq 0, \text{ finite} \qquad (4\text{-}10)$$

For the two-dimensional signal, a signal delayed by δ_1, δ_2 units in the horizontal and vertical direction respectively, where δ_1 and δ_2 are sufficient length of spatial units such that the noise field would be decorrelated but the signal field still remains correlated, then

$$E[N(k,\ell)\ N\ (k-\delta_1,\ell-\delta_2)] = 0$$

$$E[S(k,\ell)\ S\ (k-\delta_1,\ell-\delta_2)] \neq 0, \text{ finite} \qquad (4\text{-}11)$$

and again it is assumed the signal field and noise field are not correlated with each other.

Now if this delayed signal is used as a primary input and the original input used as a reference input to the adaptive filter, then referring to Figure (4-1) of previous section, $S(j-\delta)$ or $S(k-\delta_1,\ell-\delta_2)$ can be considered as N_1 in Figure (4-1) and $N(j-\delta)$ or $n(k-\delta_1,\ell-\delta_2)$ as M_1, and $S(j)$ or $S(k,\ell)$ can be considered as N_o and $N(j)$ or $N(k,\ell)$ as M_o in Figure (4-2), respectively.

From equation (4-10) and the assumptions that the signal and noise are uncorrelated, it is seen that the assumptions made in the last section for the various signals holds here.

Therefore, from the argument in Section IV-A, the filter output would be a good estimate of the signal S. Figure 4-2 shows the noise cancelling (or signal estimation) scheme discussed above.

V. EXPERIMENT AND RESULTS

In this chapter, a computer experiment is performed to check the feasibility of the algorithms derived in Chapter III for certain applications. The signal estimation problem for a noise corrupted signal is treated here for both one-dimensional and two-dimensional cases. Nonrecursive adaptive filtering and recursive filtering have been examined and the performances of adaptive filters are compared to that of the optimal Wiener solution. The adaptive noise cancelling scheme is used for this application.

First, consider a band limited one-dimensional signal S corrupted by noise N; it is desired to estimate the signal. If the statistics of both signal and noise are known apriori, a fixed optimal filter to estimate the signal can be designed by the Wiener Hopf solution of equation (3-11).

Here it is assumed that these statistics are not known apriori but only that the signal is narrowband and the noise is a broad band signal and the signal is entirely uncorrelated with the noise. Then the signal has a wide correlation function while the noise has a narrow correlation function. Separation of this broadband noise and narrowband signal is now required for the estimation of the signal.

It is assumed further that the desired (or reference) signal which is needed for the adaptive process is not available, that is, no other possible reference signal is available which may have some correlation with the signal we want to estimate.

This problem can be considered as an adaptive noise cancelling problem without reference input. Assuming that the noise is white, then from the Figure (4-2) one unit delay is enough to decorrelate the noise component appearing in the adaptive filter input from the noise component in the desired signal. These components will thus appear in the error but not in the filter output. The narrowband component, on the other hand, will not be decorrelated by the delay and will appear in the adaptive filter output.

The input signal would be

$$x(j) = S(j) + N(j)$$

where $S(j)$ bandlimited signal

$N(j)$ white noise

and the reference input would be

$$d(j) = S(j-1) + N(j-1).$$

The form of autocorrelation function of the signal is assumed as

$$R(m) = \rho^{|m|} \cos w_0 m$$

For the purpose of computer simulation, the following values are assigned:

$\rho = 0.95$

$w_0 = 0.025$

and the variance of noise is 0.5.

For the optimum filter design using the above values, a transversal filter having 10 delays was used. From equation (3-11).

$$\overline{W}_{LMS} = R_{xx}^{-1} R_{xd}$$

The autocorrelation matrix R_{xx} was computed as

1.50000	0.94970	0.90137	0.85496	0.81044	0.76774	0.72684	0.68767	0.65020	0.61436
0.94970	1.50000	0.94970	0.90137	0.85496	0.81044	0.76774	0.72684	0.68767	0.65020
0.90137	0.94970	1.50000	0.94970	0.90137	0.85496	0.81044	0.76774	0.72684	0.68767
0.85496	0.90137	0.94970	1.50000	0.94970	0.90137	0.85496	0.81044	0.76774	0.72684
0.81044	0.85496	0.90137	0.94970	1.50000	0.94970	0.90137	0.85496	0.81044	0.76774
0.76774	0.81044	0.85496	0.90137	0.94970	1.50000	0.94970	0.90137	0.85496	0.81044
0.72684	0.76774	0.81044	0.85496	0.90137	0.94970	1.50000	0.94970	0.90137	0.85496
0.68767	0.72684	0.76774	0.81044	0.85496	0.90137	0.94970	1.50000	0.94970	0.90137
0.65020	0.68767	0.72684	0.76774	0.81044	0.85496	0.90137	0.94970	1.50000	0.94970
0.61436	0.65020	0.68767	0.72684	0.76774	0.81044	0.85496	0.90137	0.94970	1.50000

and the crosscorrelation matrix R_{xd} as

R_{xd}^T =

1.00000	0.94970	0.90137	0.85496	0.81044	0.76774	0.72684	0.68767	0.65020	0.61436

Then the optimum Wiener Hopf solution gives the filter coefficients as

$\overline{W}$ = Wiener Weight-Vector =

0.33297
0.21191
0.13475
0.08558
0.05427
0.03434
0.02169
0.01369
0.00870
0.00567

For simulation of the bandlimited signal, the state and output equations of example 1) in Section 2 of Chapter II are used.

For the nonrecursive adaptive filter application, again 10 delays and = -0.005 as a step size in LMS algorithm were used, and 2 delays for both feedforward and feedback path and = -0.001 for k_a, k_b (equation(3-42)), were used in the recursive filter application of both Feintuck's algorithm and the algorithm developed here. Eight points were used for error square averaging for the gradient estimation (L=8, in Equation (3-40)). The experimental results are plotted in the following along with the descriptions and optimal solution for the purpose of comparison. The results indicate the adaptive recursive filter appears to perform as well as the optimal Wiener filter once it reaches a steady state condition.

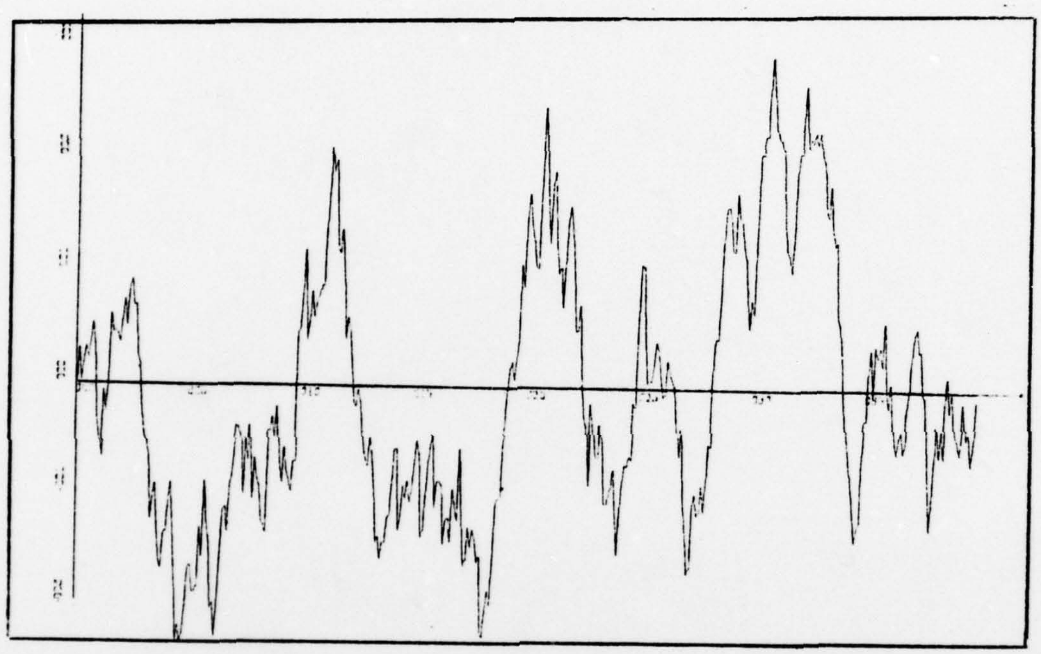


FIGURE R-1 SIGNAL STATISTICS

Autocorrelation = $0.96^{m} \cos(0.025\, m)$

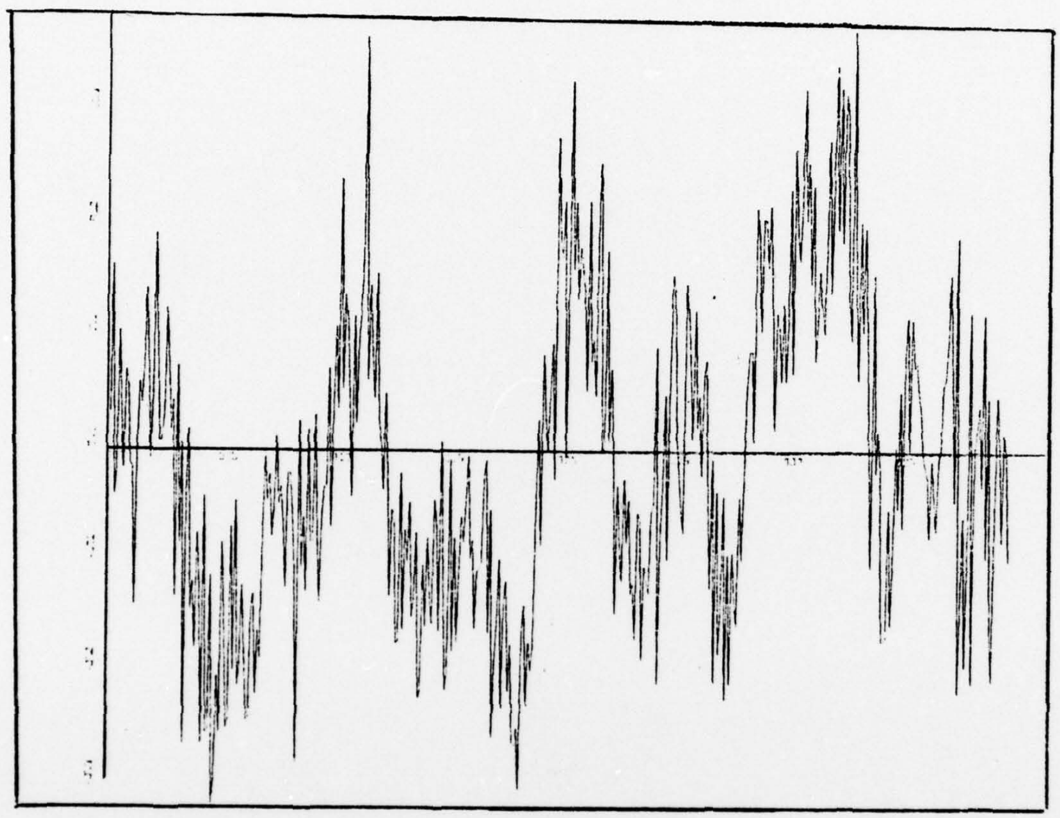


FIGURE R-2 NOISE CORRUPTED SIGNAL

NOISE: WHITE: Zero Mean

: Variance = 0.5

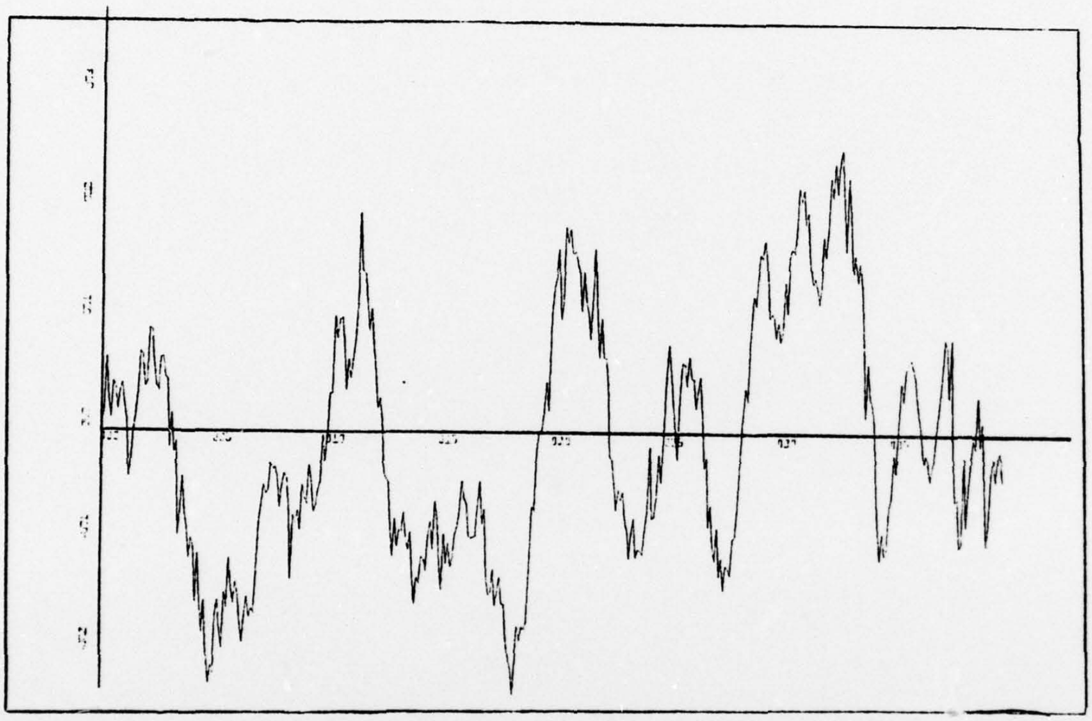


FIGURE R-3 WIENER-HOPF FILTERING

10 Delays are used.

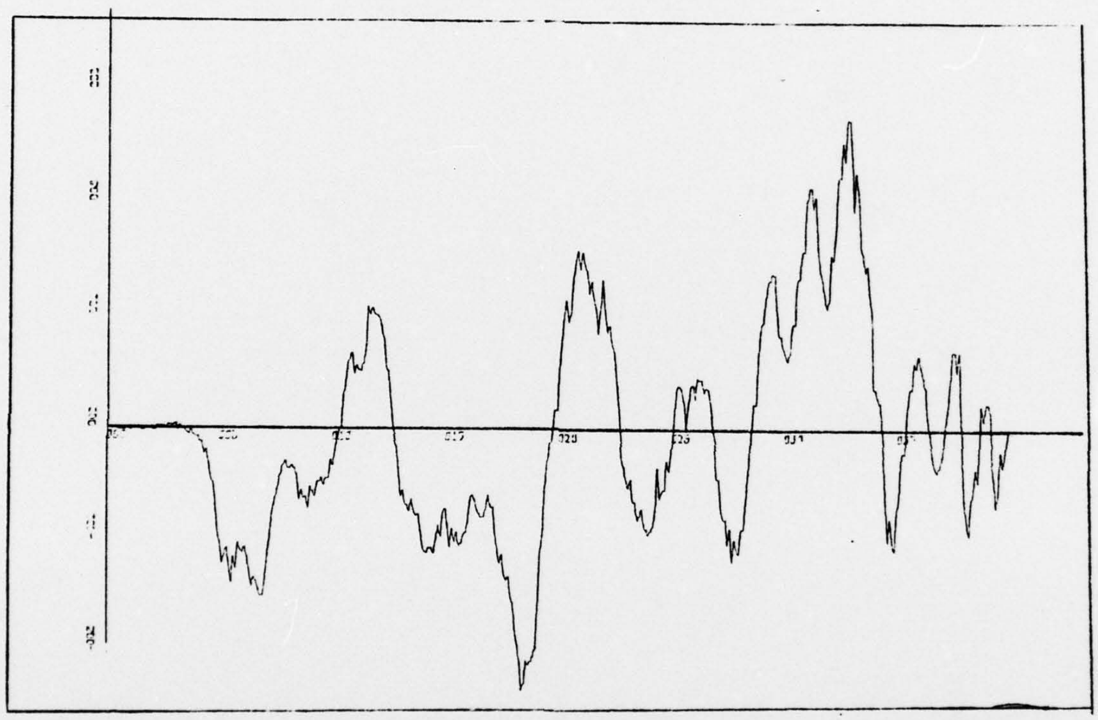


FIGURE R-4 WIDROW'S NONRECURSIVE ADAPTIVE FILTERING

Number of delays used: 10

Stepsize Used: -0.005

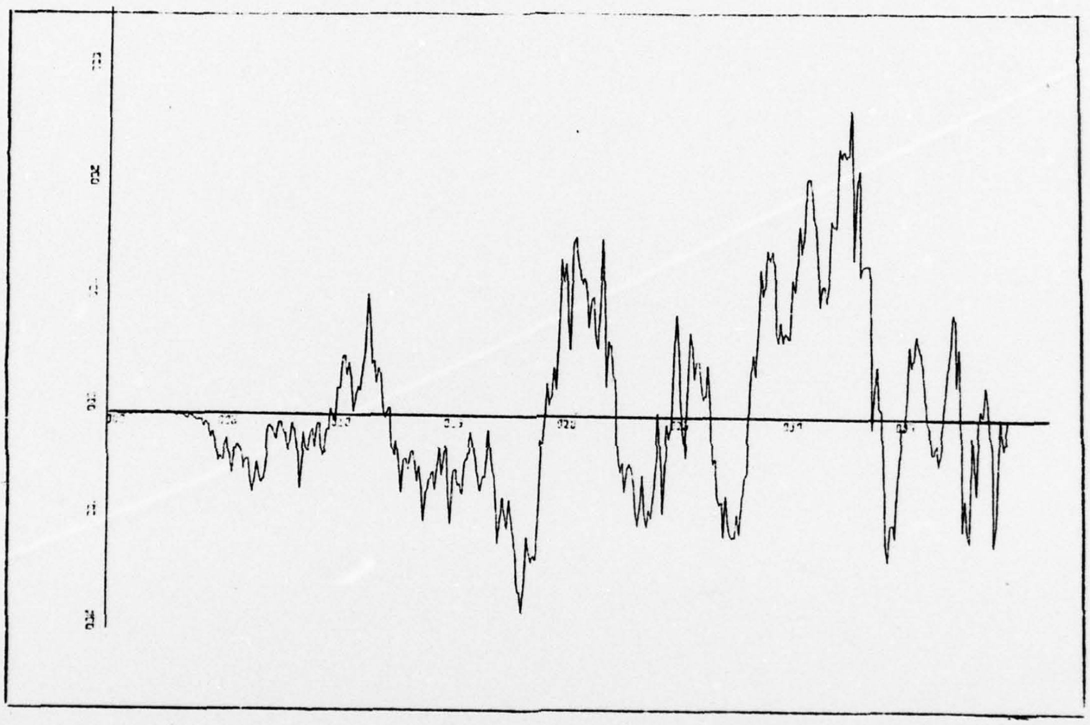


FIGURE R-5 FILTERING BY FEINTUCK'S ALGORITHM

NUMBER OF DELAYS IN FEED FORWARD PATH: 2

IN FEED BACK PATH : 2

STEPSIZE USED: -0.001

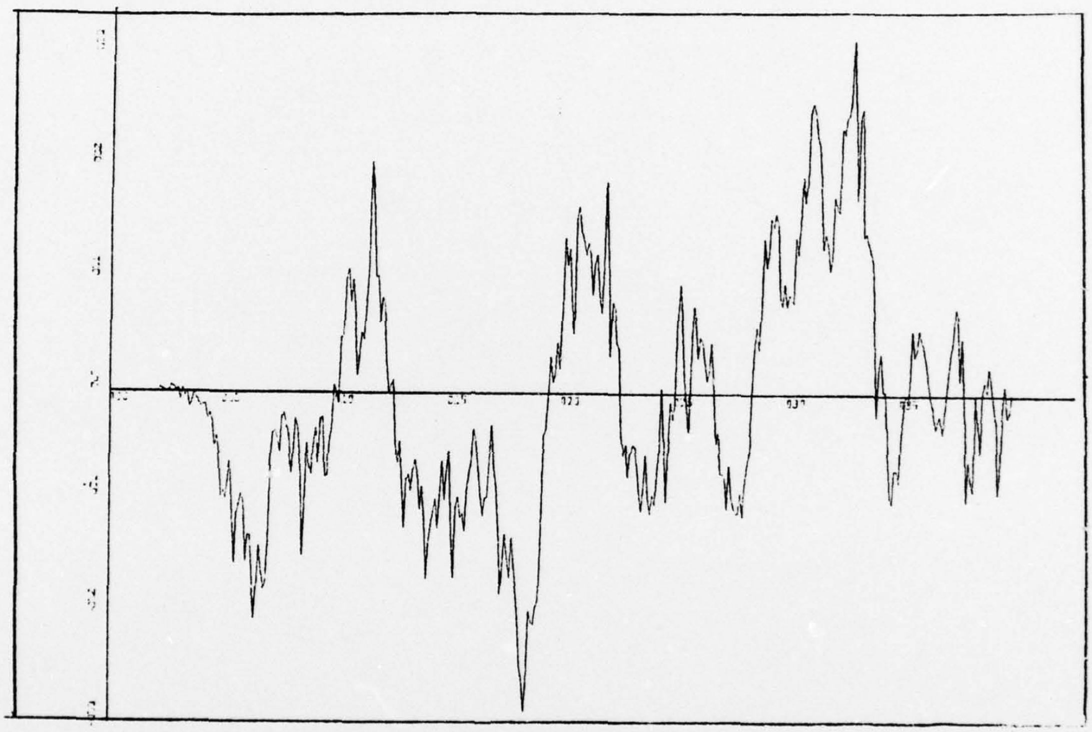


FIGURE R-6   FILTERING BY THE ALGORITHM USING A FINITE POINT MOVING SQUARE ERROR AVERAGE For the estimation of gradient

Number of Delays in Feed Forward Path: 2

in Feed Back Path : 2

Stepsize Used: -0.001

As a second application, consider an image sensed by an image sensing device of an NxN sensing elements array. It is assumed that this image is composed of correlated background and a three diagonal line target trajectory. This image may be interfered with by the internal noise of device (assumed white). Then the output image includes three types of processes:

$$x(k,\ell) = S(k,\ell) + T(k,\ell) + w(k,\ell)$$

where

$S(k,\ell)$ = correlated background

$T(k,\ell)$ = Target strength (three diagonal line)

$W(k,\ell)$ = noise.

Again it is assumed that no statistics are known apriori and the correlated background is a narrowband signal. It is further assumed that the correlated background and noise are uncorrelated with each other. It is proposed to separate the three diagonal lines from the background noise. Again, the same argument holds that this problem is a two-dimensional noise-cancelling problem in which no reference is available. It is further assumed that the correlated background is a band pass process for which the autocorrelation function is

$$R_{ss}(m,n) = \rho_r^{|m|} \rho_v^{|n|} \cos w_h n \cos w_v n$$

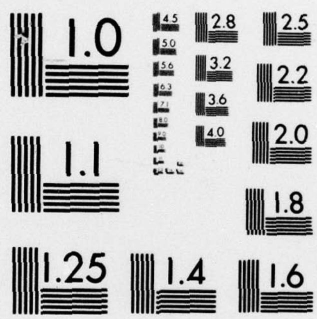
where ρ_h ρ_v represent horizontal and vertical direction correlation coefficients respectively.

AD-A053 852 NAVAL POSTGRADUATE SCHOOL MONTEREY CALIF F/G 9/3
AN ADAPTIVE RECURSIVE FILTER, (U)
DEC 77 S KO

UNCLASSIFIED NL

2 OF 2
AD
A053852

END
DATE
FILMED
6 -78
DDC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

Going back to Figure (4-2) in the previous section, the delay $z_1^{-1} z_2^{-1}$ will be sufficient to decorrelate the white noise and diagonal line target. Then as a result of filtering, the system output (or the residual field) will be the desired signal. It should be noted that this residual signal is composed of an estimate of three diagonal lines and white noise as well as some granularity due to the fixed stepsize[1].

The problem of enhancing the target diagonal line which is subjected to the noise (white noise and adaptation noise) is another problem of interest. It will not be considered in this work.

For the purpose of simulation, the following values were used:

1) $\rho_v = \rho_h = 0.96 \quad w_h = w_v = 0.143$

2) $\rho_v = \rho_h = 0.99 \quad w_h = w_v = 0.143$

The variance of correlated background = 1.0

White noise variance = 0.1

Target diagonal line intensity = 1.8

For the optimal filter design, using above values for $\rho_v = \rho_h = 0.96$, the Wiener-Hopf solution of Equation (3-30)is:

$$\overline{W}_{LMS} = R_{xx}^{-1} R_{xd}$$

where $\overline{W}_{LMS}$, R_{xx}, R_{xd} are defined by equations (3-22), (3-28), (3-29), respectively.

Using p=3, q = 3 in equation (3-20),

R_{xx}=

$$\begin{bmatrix}
1.10000 & 0.97989 & 0.94029 & 0.97989 & 0.96019 & 0.92138 & 0.94029 & 0.92138 & 0.88414 \\ \hline
0.97989 & 1.10000 & 0.97989 & 0.94029 & 0.97989 & 0.96019 & 0.92138 & 0.94029 & 0.92138 \\
0.94029 & 0.97989 & 1.10000 & 0.97989 & 0.94029 & 0.97989 & 0.96019 & 0.92138 & 0.94029 \\
0.97989 & 0.94029 & 0.97989 & 1.10000 & 0.97989 & 0.94029 & 0.97989 & 0.96019 & 0.92138 \\
0.96019 & 0.97989 & 0.94029 & 0.97989 & 1.10000 & 0.97989 & 0.94029 & 0.97989 & 0.96019 \\ \hline
0.92138 & 0.96019 & 0.97989 & 0.94029 & 0.97989 & 1.10000 & 0.97989 & 0.94029 & 0.97989 \\
0.94029 & 0.92138 & 0.96019 & 0.97989 & 0.94029 & 0.97989 & 1.10000 & 0.97989 & 0.94029 \\
0.92138 & 0.94029 & 0.92138 & 0.96019 & 0.97989 & 0.94029 & 0.97989 & 1.10000 & 0.97989 \\
0.88414 & 0.92138 & 0.94029 & 0.92138 & 0.96019 & 0.97989 & 0.94029 & 0.97989 & 1.10000
\end{bmatrix}$$

$(R_{xd})^T$=

$$\begin{bmatrix} 1.10000 & 0.97989 & 0.94029 & 0.97989 & 0.96019 & 0.92138 & 0.94029 & 0.92138 & 0.88414 \end{bmatrix}$$

$(W_{LMS})^T$=

$$\begin{bmatrix} 0.34079 \\ 0.28303 \\ -0.[illegible]1194 \\ \hline 0.24031 \\ 0.[illegible]9847 \\ -0.02430 \\ 0.11629 \\ \hline -0.02334 \\ -0.04117 \end{bmatrix}$$

Only the recursive simulation is performed and compared with the nonrecursive Wiener filter. The simulation results are shown on the following pages and indicate that although the optimal performance of the Wiener filter is not achieved, the adaptive recursive filter performs well.

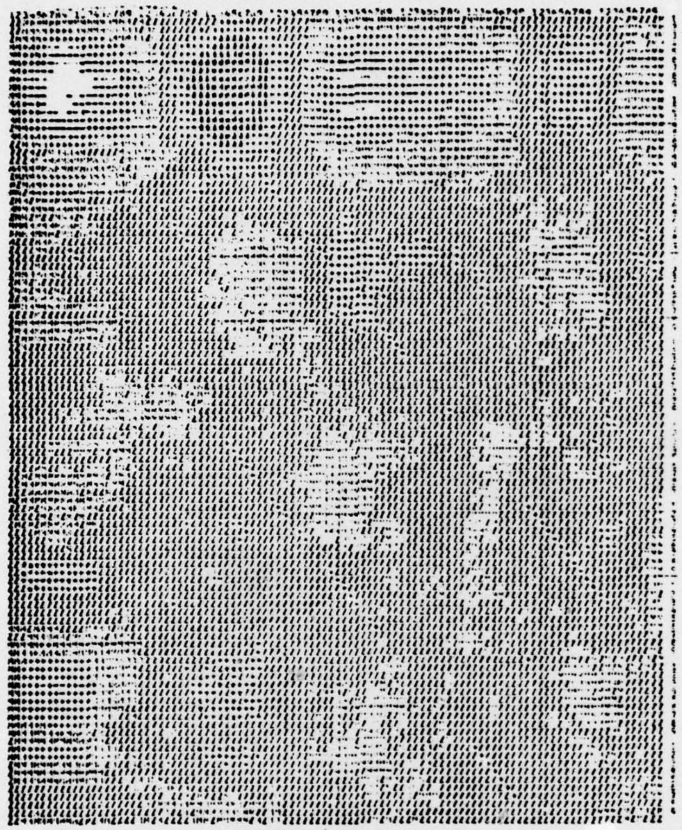


FIGURE R-7 CORRELATED BACKGROUND + THREE STREAKS + WHITE NOISE

Background: $R(m,n) = 0.96^{|m|}\,0.96^{|n|}\cos(0.143\,m)\cos(0.143n)$

Variance = 1.0

White noise: zero mean

variance = 0.1

Streaks intensity: 1.8

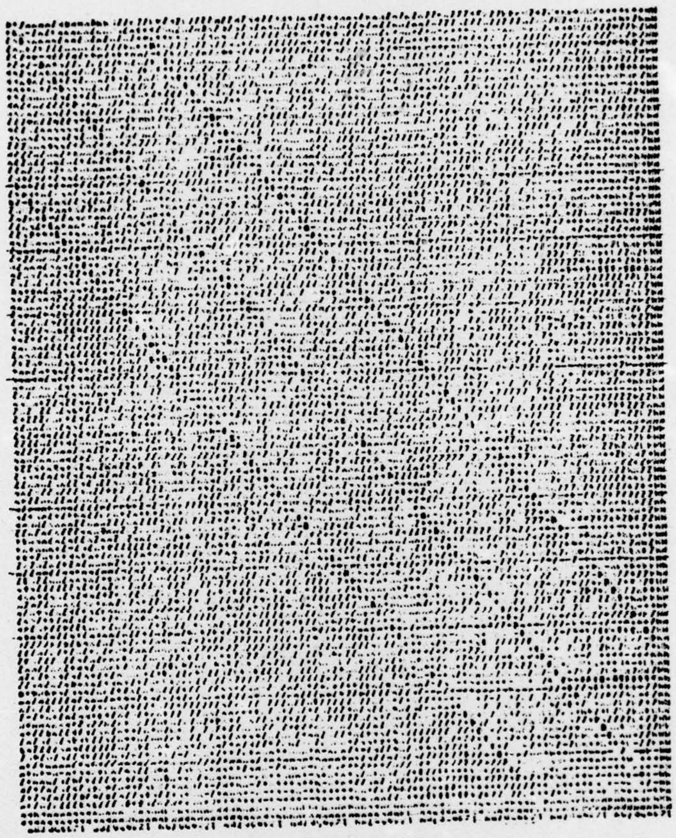


FIGURE R-8 **RESIDUAL SIGNAL AFTER WIENER FILTERING Figure R-7**
White Noise + Streaks

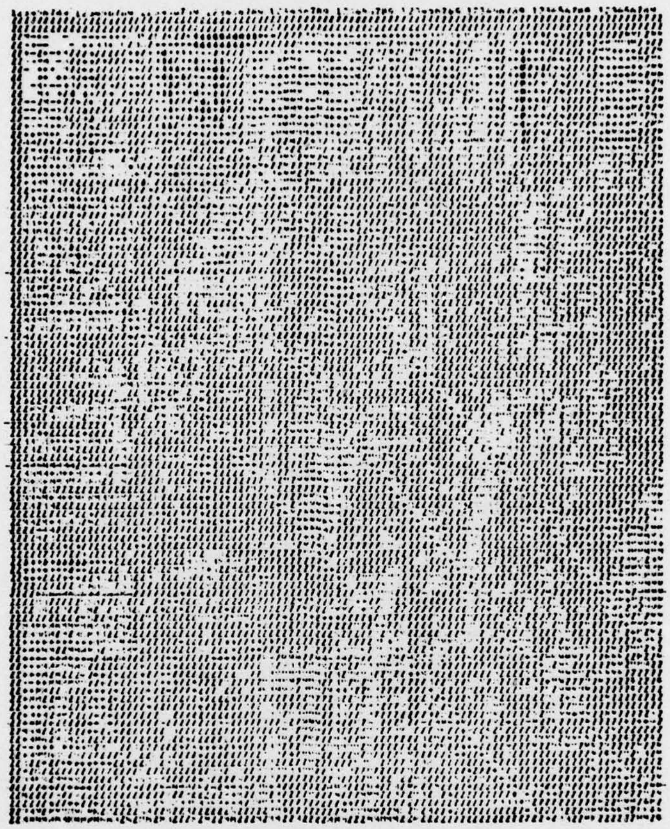


FIGURE R-9 RESIDUAL SIGNAL AFTER RECURSIVE FILTERING

White Noise + Streaks + Adaptation Noise

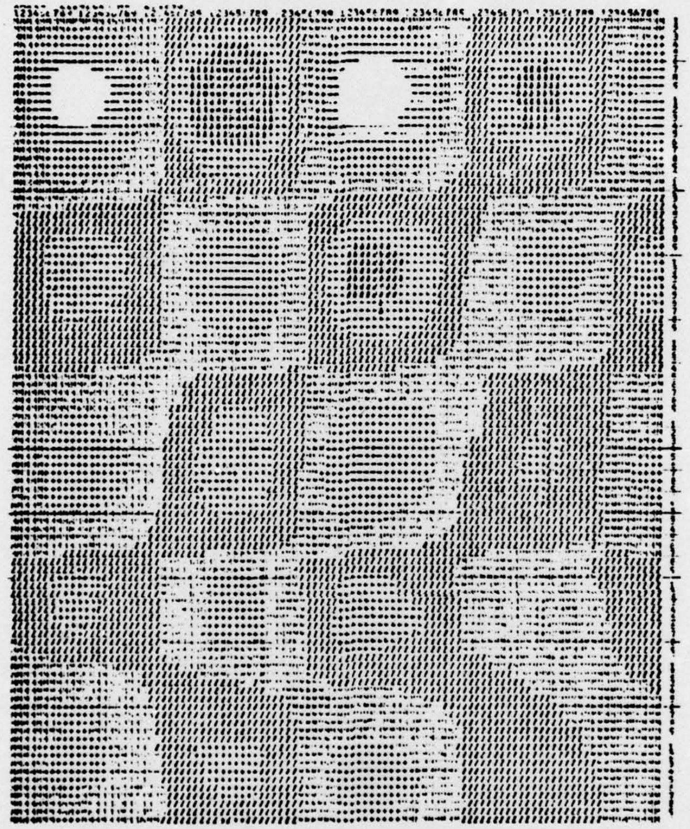


FIGURE R-10 Backgrounds + White noise + Streaks

Background: $R(m,n) = 0.99^{m} \, 0.99^{n} \cos(0.143m)\cos(0.143n)$

Variance = 1.0

White noise: zero mean

Variance = 0.1

Streaks Intensity: 1.8

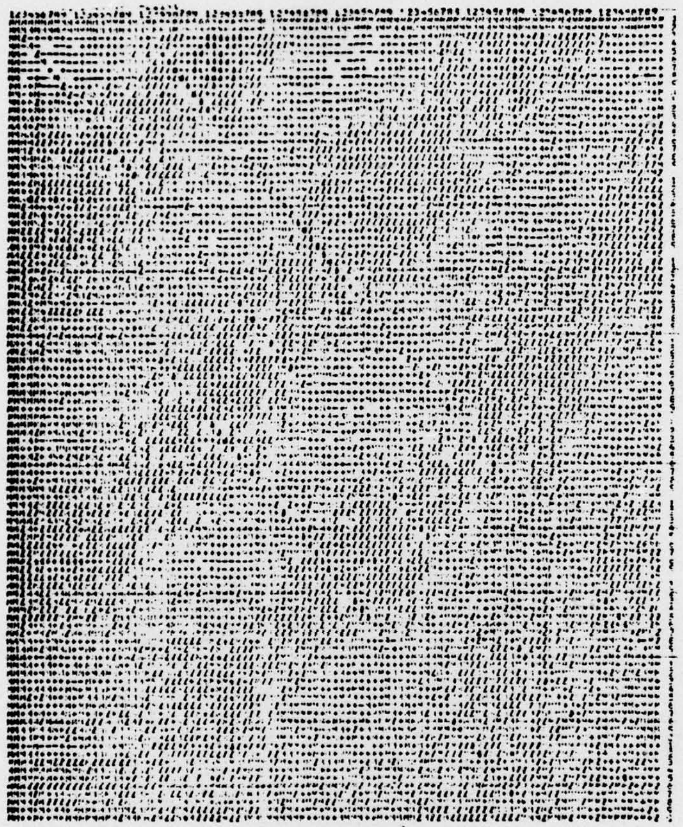


FIGURE R-11 RESIDUAL SIGNAL AFTER WIENER FILTERING

Figure R-10

White Noise + Streaks

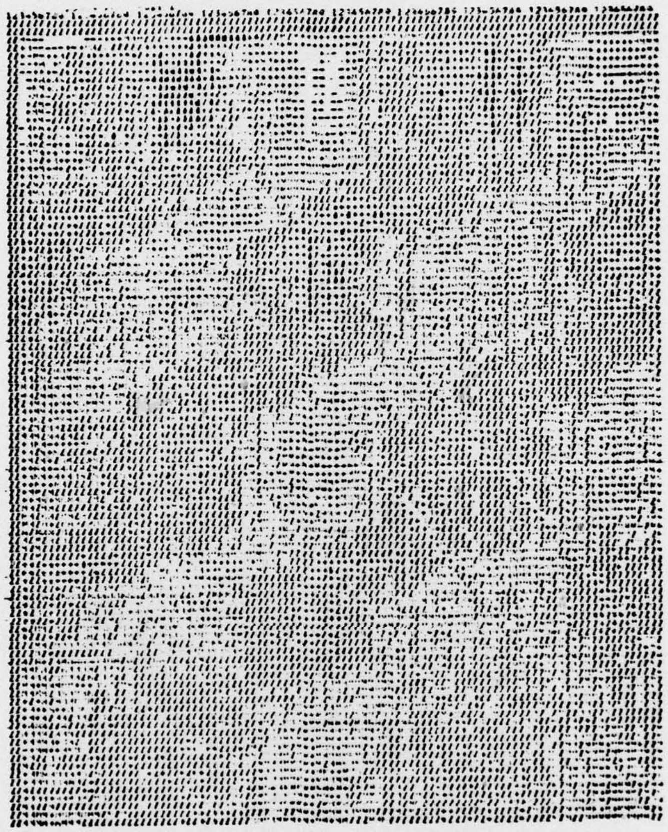


FIGURE R-12 RESIDUAL SIGNAL AFTER ADAPTIVE RECURSIVE FILTERING

White Noise + Streaks + Adaptation Noise

VI. CONCLUSIONS

The study described herein has developed a new algorithm for the one-dimensional signal filtering problem and extends this to two-dimensional processing. It is an adaptive recursive filtering algorithm based on the steepest descent gradient method which employs the finite point square error for the gradient estimation rather than instantaneous square error.

A simplified two-dimensional version of this algorithm is developed. It is designed to estimate the signal in real-time operation in cases where the statistics of both signal and corrupting noise are not available apriori. The algorithm learns the statistics and adapts even though it is not optimal, which means that it seeks the minimum of the error criterion. It should be noted that Widrow's nonrecursive adaptive filtering algorithm gives the global minimum of performance criterion due to the fact that for the stationary process, the mean square error is the quadratic form of weight vectors, but for the recursive adaptive filter, local minima may be found instead of the global minimum. The computer simulation shows that for the examples considered here the algorithms presented learn the statistics of signal and adapt. Several points can be observed through the experimental results [see Figure R-1 through R-12].

1) All the algorithms presented here give a satisfactory result after the transients die out even though they are not optimal.

2) The algorithm which employs the finite point average square error for the gradient estimates gives more rapid convergence than Feintuck's algorithm. The possible reasons may be due to the fact that the output information is fed back and used for the filter coefficients updating process and the sensitivity information propagates through the recursive equation as the iteration proceeds, while Feintuck's algorithm discards the sensitivity information.

3) The algorithm developed here gives the best results among the various algorithms presented at the expense of complex hardware. Note that the required number of additional sensitivity filters (equations (3-47), (3-48)) would be the number of filter coefficients, and due to the L point averaging process, additional storage elements are also needed. The possible reason for the good results may be due to the fact that the averaging process [equation (3-40)] for the gradient estimate gives a smaller error between true gradient and estimated gradients than the gradient estimate based on instantaneous square error does, while both give unbiased gradient estimates.

Due to the emerging interest in adaptive recursive filters, further research on this subject may be worthwhile. The following are left open for further research:

1) Comparison of steady state performance of the recursive adaptive filter with the Kalman filtering technique. It would lead to a better understanding of the performance of the recursive filter to express the filter coefficients , in terms of steady-state Kalman filter gains.

2) Mathematical derivation of the bound in step size of the filter coefficient updating process for convergence and stability. It is believed that this bound may be attained by setting up the constraints first such that the value of performance criterion decreases monotomically to a minimum as the iteration progresses.

3) Modification of the algorithms for the case that partial statistics of signal or noise are available apriori.

4) Derivation of the algorithm based on a different performance criterion such as maximum likelihood ratio, maximum signal to noise ratio, etc.

5) Derivation of the algorithm based on the different minimization techniques such as Newton's method or Fletcher-Powell methods, etc., for a given performance criterion.

LIST OF REFERENCES

1. Widrow, B., "Adaptive Filters," Aspects of Network and System Theory, R. E. Kalman and N. DeClaris, Eds., New York, Holt, Rinehart and Winston, 1970.

2. Feintuck, P. L., "An Adaptive Recursive LMS Filter," Proceedings of IEEE. November 1976.

3. White, S. A., "An Adaptive Recursive Filter," Proceedings 9th Asilomar Conference on Circuits, Systems and Computers, November 1975.

4. Widrow, B., et al, "Adaptive Noise Cancelling: Principles and Applications," Proceedings of IEEE, December 1975.

5. Blake, Ian F., and Thomas, John B., "The Linear Random Process," Proceedings of IEEE, October 1968.

6. Makhoul, John, "Linear Prediction: A Tutorial Review," Proceedings of IEEE, April 1975.

7. Papoulis, A., "Probability, Random Variables and Stochastic Process," New York, McGraw-Hill, 1965.

8. Weiner, N., "The Extrapolation, Interpolation and Smoothing of Stationary Time Series with Engineering Applications," John Wiley and Sons, Inc. New York, 1949.

9. Kalman, R. E., "A New Approach to Linear Filtering on Prediction Problems," ASME Trans., J. Basic Eng., Vol. 83, March 1960, pp. 34-35.