INFORMATION TO USERS

This was produced from a copy of a document sent to us for microfilming. While the most advanced technological means to photograph and reproduce this document have been used, the quality is heavily dependent upon the quality of the material submitted.

The following explanation of techniques is provided to help you understand markings or notations which may appear on this reproduction.

1. The sign or "target" for pages apparently lacking from the document photographed is "Missing Page(s)". If it was possible to obtain the missing page(s) or section, they are spliced into the film along with adjacent pages. This may have necessitated cutting through an image and duplicating adjacent pages to assure you of complete continuity.
2. When an image on the film is obliterated with a round black mark it is an indication that the film inspector noticed either blurred copy because of movement during exposure, or duplicate copy. Unless we meant to delete copyrighted materials that should not have been filmed, you will find a good image of the page in the adjacent frame. If copyrighted materials were deleted you will find a target note listing the pages in the adjacent frame.
3. When a map, drawing or chart, etc., is part of the material being photographed the photographer has followed a definite method in "sectioning" the material. It is customary to begin filming at the upper left hand corner of a large sheet and to continue from left to right in equal sections with small overlaps. If necessary, sectioning is continued again—beginning below the first row and continuing on until complete.
4. For any illustrations that cannot be reproduced satisfactorily by xerography, photographic prints can be purchased at additional cost and tipped into your xerographic copy. Requests can be made to our Dissertations Customer Services Department.
5. Some pages in any document may have indistinct print. In all cases we have filmed the best available copy.

University
Microfilms
International

300 N. ZEEB RD., ANN ARBOR, MI 48106

8216344

Merchant, Gulamabbas Abdulhusein

RECONSTRUCTION OF SIGNALS FROM PHASE: EFFICIENT
ALGORITHMS, SEGMENTATION, AND GENERALIZATIONS

Rice University

PH.D. 1982

University
Microfilms
International 300 N. Zeeb Road, Ann Arbor, MI 48106

RICE UNIVERSITY

RECONSTRUCTION OF SIGNALS FROM PHASE:
EFFICIENT ALGORITHMS, SEGMENTATION, AND GENERALIZATIONS

by

Gulamabbas A. Merchant

A THESIS SUBMITTED
IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE

DOCTOR OF PHILOSOPHY

APPROVED, THESIS COMMITTEE:

Dr. Thomas W. Parks
(Professor of Elect. Eng.) Chairman

Dr. C. Sidney Burrus
(Professor of Elect. Eng.)

Dr. John E. Kilpatrick
(Professor of Chemistry.)

Dr. John E. Dennis
(Professor of Math. Sc.)

HOUSTON, TEXAS

APRIL, 1982

RECONSTRUCTION OF SIGNALS FROM PHASE:
EFFICIENT ALGORITHMS, SEGMENTATION, AND GENERALIZATIONS

Gulamabbas A. Merchant

ABSTRACT

It is well known that the phase of the Fourier transform of a signal contains a significant amount of information about the signal. Under certain circumstances the knowledge of phase is sufficient to recover the original signal. This idea can be used to perform blind deconvolution of a signal which has been distorted by a linear phase filter. However this requires the knowledge of the entire output of the filter.

Here several new formulations of phase-only blind deconvolution have been developed. Each of these formulations lead to a system of simultaneous equations whose coefficient matrix is representable by a sum of a Toeplitz and a Hankel matrix. It is shown that this system of equations can be solved efficiently by a block-Levinson type algorithm. To handle long duration filter output a procedure for approximately reconstructing the original signal from short segments is developed. Some application of the techniques above are presented. It is seen that the signal recovery is excellent. Phase-only blind deconvolution for linear phase filters relies on the reciprocal symmetry of the zeros of the transfer function $H(z)$ of the filter. This is generalized to include the filters whose zeros have other kind of symmetries.

Acknowledgements

This work has been made possible with the help and encouragement of many people. In particular I would like to thank my advisor Dr. T. W. Parks for his valuable guidance and encouragement, Charles Loeffler, Richard Wendt and Howard Coleman for their ideas and suggestions, and Nancy Hadley for her support and help in typing this dissertation.

Finally I would like to thank my mother Rashidabanu, my sister Rukhsana and my brother Arif who have endured my absence from home without complaints and who continuously encouraged me to finish my research.

This work was funded in part by the National Science Foundation Grant ECS 81-00453 and in part by J. N. Tata Foundation of India.

TABLE OF CONTENTS

Chapter 1	HISTORICAL PERSPECTIVE : DECONVOLUTION	
1.1	Introduction	1
1.2	Notation	2
1.3	Methods to Perform Deconvolution	4
1.4	The Ill-Conditioning in Deconvolution	20
1.5	Aims of This Dissertation	25
Chapter 2	THEORY OF PHASE-ONLY SIGNAL RECONSTRUCTION	
2.1	Introduction	27
2.2	Theory of Signal and Phase Relationships	28
2.3	A Conceptual Technique for Computing a Signal from its Phase	36
2.4	Mathematical Relation Between a Sequence and its Phase	40
2.5	Signal Reconstruction - Least Squared Error Criterion	43
2.6	Utilization of Known Section of Sequence	51
2.7	Reconstruction Error Due to Phase Uncertainties	53
2.8	Summary of Results in this Chapter	57
Chapter 3	BLIND DECONVOLUTION USING PHASE-ONLY RECONSTRUCTION	
3.1	Introduction	59
3.2	Blind Deconvolution	59
3.3	Magnitude Weighted Phase-Only Reconstruction	62

3.4	Blind Deconvolution of Linear Phase Filters	
	Using Phase-Only Reconstruction	65
3.4.1	Linear Phase Odd Length Filters	66
3.4.2	Linear Phase Even Length Filters	67
3.5	Uncertainties of Filter Length and Signal Length	69
3.6	Reconstruction Error Due to Measurement Error	71
3.7	Examples of Phase-Only Blind Deconvolution	72
3.8	Conclusions	80
Chapter 4	EFFICIENT ALGORITHM TO SOLVE TOEPLITZ-PLUS-HANKEL SYSTEM OF EQUATIONS	
4.1	Introduction	81
4.2	Notation	82
4.3	Conversion of a Toeplitz-Plus-Hankel Operator to a Block-Toeplitz Operator	83
4.4	Block-Levinson Algorithm	85
4.5	Use of Symmetries to Reduce Computation	92
4.6	General Discussion	94
4.7	Conclusion	95
Chapter 5	APPLICATION OF PHASE-ONLY RECONSTRUCTION TO BLIND DECONVOLUTION	
5.1	Introduction	96
5.2	Phase-Only Blind Deconvolution: Two Segment Case	96
5.3	Phase-Only Blind Deconvolution: Multisegment Case ..	114
5.4	Conclusions	116

Chapter 6 SOME GENERALIZATIONS OF PHASE-ONLY RECONSTRUCTION

6.1	Introduction	122
6.2	Phase-Only Blind Deconvolution: A Structure Invariant Approach	122
6.3	Notation	123
6.4	Uniqueness Theorems	126
6.5	Some Examples of f -Symmetric Filters	129
6.6	An Application with $f(z) = -z$	131
6.7	Conclusions	133

Chapter 7 CONCLUSIONS

7.1	The work in this Dissertation	135
7.2	Future Work	137

REFERENCES	138
------------------	-----

APPENDIX A	145
------------------	-----

APPENDIX B	147
------------------	-----

APPENDIX C	149
------------------	-----

APPENDIX D	151
------------------	-----

CHAPTER 1

HISTORICAL PERSPECTIVE : DECONVOLUTION

1.1. Introduction

One of the most common problems in the area of signal processing is that of deconvolution. The need to perform deconvolution can appear in quite unexpected places. This chapter will introduce the notion of deconvolution and a brief survey of how the problem of deconvolution has been tackled by various investigators. To pin down what is meant by deconvolution we will begin by its definition.

Consider a signal $x(\cdot)$ which could be either a continuous time or discrete time signal. Let the signal be passed through a linear system H with a system response $h(\tau, t)$, where t is the time variable and τ is the delay variable. The output of the system can be written as

$$y(t) = \int_{\tau} h(\tau, t) x(\tau) d\tau \quad 1.1.1$$

for the continuous case and

$$y(n) = \sum_{\tau} h(\tau, n) x(\tau) \quad 1.1.2$$

for the discrete case. When the system is time invariant equation 1.1.1 becomes

$$y(t) = \int_{\tau} h(t-\tau) x(\tau) d\tau \quad 1.1.3$$

and equation 1.1.2 becomes

$$y(n) = \sum_r h(n-r)x(r) \quad 1.1.4.$$

Equations 1.1.3 and 1.1.4 represent convolution in appropriate time domain and the convolutional operator will be denoted by a " * ". The problem of deconvolution can now be stated as:

Given some information about the system H and output signal y determine the input signal x.

The dual of the above problem is a deconvolution, too, and can be stated as:

Given some information about the input signal x and the output y of the system H, determine the parameters of the system H.

Both above problems are essentially the same in nature. However, they might differ in their computational requirement due to the difference in the number of the parameters to be determined. This problem comes up in various fields and numerous attempts have been made to solve it utilizing to the utmost the knowledge available regarding the problem. In the sequel, a variety of techniques to solve the problem of deconvolution will be presented. Some of these techniques solve more general problems, which under certain circumstances be specialized to deconvolution.

1.2. Notation:

Before proceeding any further we introduce some notation to facilitate subsequent discussion. Most of the cases involve discrete

time sequences unless otherwise specified.

$x(n)$: input signal to a linear system

$y(n)$: output of a linear system

$h(n)$: impulse response of a linear time-invariant system

$X(z)$: $\sum_n x(n)z^{-n}$ = the z-transform of $x(n)$. The z-transform of $y(n)$ and $h(n)$ can be defined, similarly.

$X(\omega)$: the discrete Fourier transform of $x(n)$ = $\sum_n x(n)e^{jn\omega} = X(z)|_{z=e^{j\omega}}$. The discrete Fourier transform of $y(n)$ and $h(n)$ can be defined similarly.

ω : angular frequency variable varying between $-\pi$ and π radians.

f : normalized frequency variable varying between $\frac{-1}{2}$ and $\frac{1}{2}$. Note that $\omega=2\pi f$.

In general the various time series will be assumed to be of finite or semi-infinite extent. For example $x(n)$ may be defined for $n \in [-n_1, n_h]$ where n_1 is finite and n_h may be finite or infinite. With these definitions, the z-transform of $y(n)$ defined in equation 1.1.4 is

$$Y(z) = H(z) X(z) \quad 1.2.1.$$

The Fourier transform of equation 1.1.4 is obtained by setting $z=e^{j\omega}$ in equation 1.2.1. Now, if $h(n)$ and $y(n)$ are known for all n then the first version of deconvolution in section 1 has an exact solution when

$$X(\omega) = \frac{Y(\omega)}{H(\omega)} \quad 1.2.2$$

is defined for all ω . Similarly, the dual deconvolution problem has a solution if

$$H(\omega) = \frac{Y(\omega)}{X(\omega)} \quad 1.2.3$$

is defined for all ω .

In equation 1.2.2, since $Y(\omega)$ is defined for all ω a sufficient condition for existence of equation 1.2.2 is that the function $\frac{1}{H(\omega)}$ be defined for all ω . As a consequence any technique which involves calculation of the reciprocal of a polynomial is said to perform deconvolution or inverse filtering. A similar situation exists with regards to equation 1.2.3.

1.3. Methods to Perform Deconvolution :

Many methods have been proposed to solve the problem of deconvolution but most can be broadly classified under three main categories viz:

- a) Time domain
- b) Frequency domain
- c) Mixed Time - Frequency domain.

This classification is at best arbitrary since the time domain and frequency domain techniques are frequently intimately related. These technique may further be classified as iterative and non-iterative. In what follows, we will present a number of techniques that have developed over the years. Some of the important ones will

be presented in more detail. However no attempt is made to derive the algorithms and the techniques presented here represent only a selection of the numerous available methods and is by no means exhaustive.

One of the earliest references to a " deconvolution " like problem is to be found in a 1795 paper by Prony [1]. Prony's method has been rediscovered repeatedly over the years. One of the more recent was its application to signal processing by Mc Donough[2] and Weiss, et al [3] in 1963. Prony recognized that if a time series $y(n)$ can be represented as a sum of weighted exponential :

$$y(n) = \sum_{r=1}^M \beta_r \mu_r^n \quad \text{for } n \geq 0$$

then the series has to satisfy the following difference equation

$$y(n) + \sum_{k=1}^n \alpha_k y(n-k) = \begin{cases} u(n) & \text{for } n < M \\ 0 & \text{for } n \geq M \end{cases} \quad 1.3.1.$$

Here $u(n)$ corresponds to the initial conditions. The parameters μ_r for $r = 1, 2, \dots, M$ are the roots of the polynomial

$$H(z) = z^M + z^{M-1} \alpha_1 + z^{M-2} \alpha_2 + \dots + \alpha_M.$$

If we take the z -transform of equation 1.3.1 we obtain

$$\sum_r y(r) z^{-r} \sum_r \alpha_r z^{-r} = \sum_{r=0}^{M-1} u(r) z^{-r} \quad 1.3.2.$$

The left hand side of equation 1.3.2 is a product of two polynomials. A comparison the equations 1.2.1 and 1.3.2 shows clearly that to obtain the α 's, we need to perform a deconvolution. Taking the set of equations for $n \geq m$ we need to solve the following system of linear

equations:

$$\sum_{k=1}^n a_k y(n-k) = -y(n) \text{ for } M \leq n \leq 2M-1 \quad 1.3.3.$$

The coefficient matrix corresponding to the above system of equations is known as a Toeplitz matrix and can be solved by a variation of the method proposed by Levinson [4] in 1947. The values of the exponentials derived from equation 1.3.3 are often not very accurate and may give rise to spurious oscillatory terms in $y(n)$ [5].

One of the major deconvolution techniques is the Wiener filtering [6]. Suppose we have an input $y(t)$ and we desire to design a filter $g(t)$ which, when acting on $y(t)$ gives a specified output $x(t)$. Wiener [6] and Boonton[7] have shown that the least-square minimization of the error between the desired and the actual output of the filter results in the following relation

$$R_{xy}(t) = \int_0^{\infty} g(t-u) R_{yy}(u) du, \quad t \geq 0 \quad 1.3.4$$

which $g(\cdot)$ has to satisfy. Here $R_{xy}(\cdot)$ is the crosscorrelation function and $R_{xx}(\cdot)$ and $R_{yy}(\cdot)$ are autocorrelation functions. This is the well-known Wiener-Hopf linear integral equation. Further, if $y(\cdot)$ is the output of a system H with impulse response $h(\cdot)$ for the input $x(\cdot)$ then $g(\cdot)$ is the inverse filter of $h(\cdot)$. Generally $y(\cdot)$ is assumed to be an output with noise $v(\cdot)$, i.e.,

$$y = h * x + v \quad 1.3.5.$$

Assuming the noise to be zero mean and uncorrelated with x , we can

write

$$R_{xy}(t) = h(-t) * R_{xx}(t) \quad 1.3.6.$$

Using equations 1.3.4 ,1.3.5 and 1.3.6 we get the following relation

$$h(-t) * R_{xx}(t) = g(t) * [h(t) * h(-t) * R_{xx}(t) + R_v(t)] \quad 1.3.7.$$

where $R_v(.)$ is the autocorrelation of the noise $v(.)$. Equation 1.3.7 needs to be solved for the inverse filter $g(t)$. Without the restriction of $t \geq 0$ in equation 1.3.4 in the frequency domain this leads to

$$G(\omega) = \frac{\underline{H}(\omega) R_{xx}(\omega)}{|H(\omega)|^2 R_{xx}(\omega) + R_v(\omega)} \quad 1.3.8.$$

where $\underline{H}(\omega)$ is the complex conjugate of $H(\omega)$. However with the above restriction the solution becomes more complicated and can be given by

$$G(s) = \frac{1}{[H(s)H(-s)R_{xx}(s) + R_v(s)]^+} \left[\frac{H(-s)R_{xx}(s)}{[H(s)H(-s)R_{xx}(s) + R_v(s)]^-} \right]^+ \quad 1.3.9.$$

where the "+" superscript indicates a rational function with the poles and zeroes in the left half plane of the Laplace transform variable s -plane and the "-" superscript indicates the poles and zeroes in the right half plane. By replacing s with z and $-s$ with z^{-1} in equation 1.3.9 we obtain the solution to the corresponding discrete version of the problem. Note that $G(s)$ is an inverse of the filter $H(s)$ in the presence of observation noise $v(t)$. If the noise is absent equation 1.3.9 reduces to

$$G(s) = \frac{1}{H(s)} \quad 1.3.10$$

The preceding technique solves the Wiener-Hoff equation in the frequency domain. For continuous time, it is extremely difficult to solve the problem in the time domain. However, in the discrete case the Wiener-Hoff equation

$$R_{xy}(n) = \sum_{m=0}^{\infty} g(m)R_{yy}(n-m) \quad n \geq 0 \quad 1.3.11$$

forms a linear system with a Toeplitz coefficient matrix when $g(m)$ is of finite duration, and can be solved directly or by a fast Levinson recursion to give the inverse filter. One further point to be noted is that the frequency domain solution assumes stationary signals. For nonstationary signals one is forced to resort to time domain solutions because the time to frequency transformation is undefined.

The difficulty in dealing with nonstationary processes in the Wiener filtering technique was overcome by a new formulation due to Kalman [8]. He combined state-space descriptions and the notion of discrete-time innovation to give a complete and elegant solution. Kalman changed the formulation of the problem by giving a model for the signal as an output of a dynamical linear system driven by white noise process instead of using the covariance of the signal process. Specifically, the signal process $z(t)$ was assumed to be described by

$$z(t) = H(t)x(t)$$

$$y(t) = H(t)x(t) + v(t) \quad t \geq t_0 \quad 1.3.12$$

$$\dot{\mathbf{x}}(t) = \mathbf{F}(t)\mathbf{x}(t) + \mathbf{G}(t)\mathbf{u}(t)$$

$$\mathbf{x}(t_0) = \mathbf{x}_0$$

where $\mathbf{x}(\cdot)$ is an n by 1 state vector, $y(\cdot)$ is the observed value of the signal $z(t)$, $v(t)$ is observation noise, and $\mathbf{u}(\cdot)$ is an m by 1 random input, such that

$$E[\mathbf{u}(t)\mathbf{u}(s)^{\tau}] = \mathbf{Q}(t)\delta(t-s) \quad 1.3.13$$

$$E[\mathbf{x}_0\mathbf{x}_0^{\tau}] = \mathbf{\Pi}_0$$

$$E[\mathbf{u}(t)\mathbf{x}_0^{\tau}] = 0 \quad t \geq 0$$

$$E[\mathbf{u}(t)\mathbf{v}(s)^{\tau}] = \mathbf{C}(t)\delta(t-s) .$$

Here $\mathbf{H}(\cdot)$, $\mathbf{F}(\cdot)$, $\mathbf{G}(\cdot)$, $\mathbf{Q}(\cdot)$, $\mathbf{\Pi}_0(\cdot)$ and $\mathbf{C}(\cdot)$ are assumed to be known.

Kalman gave the solution for $z(t)$ in the least-square error sense in an algorithmic form as

$$\underline{z}(t) = \mathbf{H}(t)\underline{x}(t) \quad 1.3.14$$

where the quantities with " $\underline{\quad}$ " represent their estimates and

$$\underline{x}(t) = \mathbf{F}(t)\underline{x}(t) + \mathbf{K}(t)\varepsilon(t)$$

$$\underline{x}(t_0) = 0$$

$$\varepsilon(t) = y(t) - \underline{z}(t) = y(t) - \mathbf{H}(t)\underline{x}(t)$$

$$K(t) = P(t)H(t) + G(t)C(t)$$

and $P(.)$ is an n by n covariance matrix of the error in the state estimates

$$P(t) = E[\tilde{x}(t)\tilde{x}^T(t)] , \quad \tilde{x}(t) = x(t) - \underline{x}(t) . \quad 1.3.15$$

The matrix $P(.)$ can be computed as a unique solution of the nonlinear differential equation

$$\dot{P}(t) = F(t)P(t) + P(t)F(t) - K(t)K(t)^T + G(t)Q(t)G(t) \quad 1.3.16$$

with $P(t_0) = P_0$. Equation 1.3.16 is the matrix version of the Riccati equation [9]. A discrete version of the Kalman filtering method closely parallels the situation described above in equations 1.3.12 to 1.3.16 with some small differences. These are described in an excellent survey paper by Kailath [10].

It should be noted that the processes involved in the preceding equations are not stationary. The Kalman filter is used in a wide variety of fields such as aerospace, seismic, speech and signal processing. We will now briefly look at how the Kalman filter can be used to perform deconvolution. Let $z(t)$ be the output of a system with the impulse response $h(t)$ due input $s(t)$. Let the observed signal be $y(t)$ with an additive noise component $v(t)$. Now we have the following equation:

$$z(t) = h(t) * s(t) \quad 1.3.17$$

$$y(t) = z(t) + v(t).$$

In state variable formulation this is

$$\dot{\mathbf{x}}_1(t) = \mathbf{A}_1(t)\mathbf{x}_1(t) + \mathbf{G}_1(t)s(t) \quad 1.3.18$$

$$y(t) = z(t) + v(t) = \mathbf{H}_1(t)\mathbf{x}_1(t) + v(t).$$

The stochastic process $s(t)$ is assumed to be modeled by passing white noise $w(t)$ through a linear system and can be described by the equations

$$\dot{\mathbf{x}}_2(t) = \mathbf{A}_2(t) \mathbf{x}_2(t) + \mathbf{G}_2(t) w(t) \quad 1.3.19$$

$$s(t) = \mathbf{H}_2(t) \mathbf{x}_2(t)$$

Using equations 1.3.18 and 1.3.19 we have

$$\dot{\mathbf{x}}_1(t) = \mathbf{A}_1(t) \mathbf{x}_1(t) + \mathbf{G}_1(t) \mathbf{H}_2(t) \mathbf{x}_2(t) \quad 1.3.20$$

$$\dot{\mathbf{x}}_2(t) = \mathbf{A}_2(t) \mathbf{x}_2(t) + \mathbf{G}_2(t) w(t)$$

$$y(t) = \mathbf{H}_1(t) \mathbf{x}_1(t) + v(t)$$

Equation 1.3.20 can be rewritten as

$$\dot{\mathbf{x}}(t) = \mathbf{F}(t) \mathbf{x}(t) + \mathbf{G}(t) w(t) \quad 1.3.21$$

$$y(t) = \mathbf{H}(t) \mathbf{x}(t)$$

where

$$\mathbf{F}(t) = \begin{bmatrix} \mathbf{A}_1(t) & \mathbf{G}_1(t)\mathbf{H}_2(t) \\ 0 & \mathbf{A}_2(t) \end{bmatrix}$$

$$G(t) = \begin{bmatrix} 0 \\ G_2(t) \end{bmatrix}, \quad x(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix}$$

$$H(t) = [H_1(t) \quad 0].$$

Equation 1.3.21 is identical in form to equation 1.3.12 and can be used to estimate the states. Then $\underline{s}(t)$ can be determined by

$$\underline{s}(t) = [0 \quad H_2(t)] \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} \quad 1.3.22$$

As equation 1.3.17 shows, we have performed deconvolution to reconstruct $s(t)$ from the noisy observation $y(t)$.

The Kalman filtering approach can be looked upon as a state-variable approach to the determination of inverse filters. However, more recently there have been other attempts to apply state variable techniques to determine inverse filters, especially for multi-input, multi-output cases. Of note are two papers by Silverman 11, 12 and a short paper by Moylan [13]. Moylan, for instance, shows that given a discrete state variable model

$$z(k+1) = A z(k) + B x(k)$$

$$y(k) = C z(k) + D x(k)$$

it is possible to find a minimal order stable inverse provided the matrix

$$\begin{bmatrix} A - \lambda I & B \\ C & D \end{bmatrix}$$

has full rank for all $|\lambda| \geq 1$. This is equivalent to the minimum

phase requirement for single input, single output systems. The algorithm finds the largest order inverse system whose form is a state variable model too. Like the Kalman filter, the method can be applied to time varying systems. The main drawback of this technique is the requirement of computing the rank of various matrices, a difficult task at best. This restricts the order of the original system one can deal with in practice.

The principal difficulty in applying Kalman filtering and Weiner filtering is in obtaining the statistics of the noise $u(t)$ and the dynamics of the system involved. Kalman filters require knowledge of the model which produced the signal in response to white noise input. Further, it requires that the transfer function be rational and that the number of zeros be less than the number of poles. Moreover, in practice one does not use ensemble averages to design the deconvolution filters. In fact, by an implied assumption of ergodicity the time averages are used to perform the computation. Two excellent references on the use of Kalman filters for deconvolution are Baylers et al [14] and Crump [15]. Crump has observed that the deconvolution results obtained using this method are at least as good as those obtained by the conventional frequency domain methods for stationary processes. The real potential of the method lies in its ability to handle nonstationary processes.

While the progress in Kalman filtering was taking place another technique which could be used for inverse filtering was being developed. In 1953 Wardsworth et al [16] and Robinson [17] introduced

linear prediction as a technique for deconvolution. Wardworth et al used predictive deconvolution to process reverberant seismic data. Robinson assumed the input signal to be of stochastic nature and further applied predictive deconvolution to nonstationary data by time gating or time windowing procedure. The nonstationary is assumed to change slowly enough so that time gating procedures can be used. Linear prediction or inverse filtering has been used in many areas including speech processing, econometrics and statistics. Under the name of autoregressive estimation it was already used in the nineteen-twenties to solve curve fitting problems. Linear prediction has been arrived at from many other formulations, such as maximum likelihood [18], spectral flatness [19], and geometric Hilbert Space formulation [20].

We now look briefly at linear prediction. Linear prediction is essentially an all pole modeling of a signal. Let the discrete signal $x(r)$ be available up to time $r=(n-1)$. Then the n th sample is assumed to be expressible as a weighted linear combination of the preceding M samples. The one-step prediction error in the n -th sample is

$$e(n) = x(n) - \sum_{i=1}^M (-a_i) x(n-i) \quad 1.3.23$$

Defining the total error to be

$$\epsilon = \sum_{n=n_0}^{n_1} e^2(n)$$

and performing the minimization of ϵ with respect to the coefficient

a_r gives rise to the following system of linear equations

$$\sum_{i=1}^M a_i C_{ik} = -C_{0k} \quad k = 1, 2, \dots, M \quad 1.3.24$$

where

$$C_{ij} = \sum_{n=n_0}^{n_1} x(n-i) x(n-j) .$$

Equation 1.3.24 forms a normal system of equations. If n_0 and n_1 extended from $-\infty$ to ∞ we would have a Toeplitz system and could use Levinson recursion to solve it efficiently. However, with finite limits the coefficient matrix is not Toeplitz. Kailath et al[21] have developed a technique which solves the system with a Levinson type recursive algorithm. Equation 1.3.23 can be seen in z-transform domain as

$$E(z) = A(z) X(z) \quad 1.3.25$$

which can be rewritten as

$$X(z) = \frac{E(z)}{A(z)} \quad 1.3.26$$

Thus, the sequence x_n can be said to have been generated by an input into a linear system with the transfer function $\frac{1}{A(z)}$. The determination of $A(z)$ thus is equivalent to the determination of an inverse filter.

The solution to equation 1.3.24, when C_{ij} depends only on the difference $(i-j)$, can be shown to be a minimum phase solution. The procedure of minimizing the squared error in one step prediction is

equivalent to a fundamental assumption that the reflector series obtained in the Levinson recursion algorithm is white, so that the autocorrelation of the signal $x(n)$ is proportional to the autocorrelation of the sequence obtained from the coefficients in the Taylor series expansion of $\frac{1}{A(z)}$. While frequently valid, this assumption is not always true. The minimum phase inverse filter is the optimum solution if the original filter is also a minimum phase. When this is not true, the impulse response of the combined inverse and original filter results in a poor oscillatory approximation to an impulse. In addition, there is an implicit assumption of all-pole models for the signal in equation 1.3.23, an assumption which might not be valid. If the signal has zeros and poles, the Linear Prediction method enhances the poles at the expense of the zeros. Finally, the Linear Prediction method is sensitive to the order of inverse or to the number of unknown parameters.

A completely different technique which is characterized by the use of both the time domain and frequency domain is homomorphic deconvolution. Homomorphic signal processing usually involves a sequence called cepstrum or kepsrum, obtained by performing an inverse transform on the natural logarithm of the z transform of a sequence. For the moment we shall ignore the questions of existence, uniqueness, etc. The idea of cepstrum is a classical one appearing in the work of Poisson [22], Schwartz [23], Szego [24] and Kolmogorove [25]. In the nineteen-fifties and -sixties, homomorphic signal processing was applied to geophysical, picture processing and speech

processing fields, among others.

Consider the z transform of equation 1.1.4

$$Y(z) = H(z) X(z). \quad 1.3.27$$

Taking the natural logarithm of the above equation yields

$$\ln Y(z) = \ln H(z) + \ln X(z) \quad 1.3.28$$

provided each term in equation 1.3.27 is well defined. Thus, knowing $H(z)$ and hence $\ln H(z)$, we can obtain $\ln X(z)$ from equation 1.3.28 and then compute $X(z)$. Upon replacing z with $e^{j\omega}$ we get the usual definition of the Fourier transform. Since most of the calculations are done with Fourier transforms, the requirement of a "well-defined" translates into the requirement that the common region of convergence for $\ln Y(z)$, $\ln H(z)$ and $\ln X(z)$ include the unit circle. This requires that none of the functions involved in equation 1.3.27 have zeros on the unit circle. Now, if each of the terms in equation 1.3.28 is expanded in a Laurent series expansion then we have

$$y_n = \underline{h}_n + \underline{x}_n \quad n = 0, \pm 1, \pm 2 \dots \quad 1.3.29$$

where quantities with the "-" are the coefficients in the Laurent series expansion, popularly known as the cepstral coefficients. In certain applications, such as speech, it is well known that the $\underline{h}_n$ and $\underline{x}_n$ are non-zero for different values of n . Typically, $\underline{x}_n$ may exist for lower values of n , and $\underline{h}_n$ for higher ones. Thus, by low-time windowing of y_n we obtain $\underline{x}_n$ and thus x_n . This is also known as blind deconvolution, since it is the location rather than the values

of $\underline{h}_n$ that is important. An example of this has been discussed by Oppenheim [26]. The homomorphic transformation is not an answer to all the problems of deconvolution. The choice of the model in equation 1.3.27 is not necessarily valid in the presence of noise. Moreover the possibility of the Fourier transform having zeros on the unit circle can create a problem in the application of this technique.

As noted earlier, the problem of having zeros on the unit circle can be a stumbling block in performing deconvolution. However, there are other techniques, essentially iterative in nature, discussed by Prost et al[27] and Brigham et al[28] which deal with this situation. In their paper, Prost et al considered a situation where the system to be deconvolved has a zero-valued Fourier transform for some of the frequencies. As noted previously, in such cases the inverse of the system does not exist. However, one can still deconvolve the system under certain circumstances. Primarily we need some a priori knowledge of the input signal $x(n)$. Given that signal $x(n)$ has a support $[a, b]$ the following situation can be used to perform deconvolution:

$$x^{(1)}(n) = y(n)$$

$$x^{(m)}(n) = y(n) - [w(n) x^{(m-1)}(n)] * h(n), \quad m > 1$$

where

$$w(n) = \begin{cases} 1 & n \in [a, b] \\ 0 & n \notin [a, b] \end{cases}$$

and m is the index of iteration.

It can be shown that $x^{(m)}(n)$ converges to $x(n)$ if for any function $\phi(n)$ with a bounded support and such that $\sum_n \phi^2(n) > 1$ the following results hold:

$$\int_{-B}^B [|H(f)|^2 - 2 \operatorname{Re}(H(f)) |\Phi(f)|^2] df < 0$$

where $2B$ is the bandwidth of $H(f)$ and $\Phi(f)$ is the Fourier transform of $\phi(n)$. Sufficient condition for the above to be true is

$$0 < |H(f)|^2 < 2 \operatorname{Re}(H(f))$$

A novel technique has been proposed by Anderson[29] when the original filter is known and further it is nonminimum phase. Briefly, we have

$$Y(z) = H_1(z) H_2(z) X(z)$$

where $H_1(z)$ is the minimum phase and $H_2(z)$ is the maximum phase operator and there are no zeros on the unit circle. Then the sequence $X(z)$ can be obtained in two steps. Compute $X_1(z)$ recursively

$$X_1(z) = \frac{Y(z)}{H_1(z)} \quad 1.3.30$$

Next, compute $X_2(z^{-1})$ recursively as

$$X_2(z^{-1}) = \frac{X_1(z^{-1})}{H_2(z^{-1})} \quad 1.3.31$$

If the initial conditions are not known in equation 1.3.30 they can

be assumed to be zero. In both equations 1.3.30 and 1.3.31 we are deconvolving minimum phase systems. Time reversing the sequence $x_2(n)$ obtained in equation 1.3.31 gives us $x(n)$. If the initial conditions are not known it can be shown that the difference between $x_2(n)$ and $x(-n)$ decays exponentially with increasing n .

One of the problems with this method is the requirement that the original filter be factored into a minimum phase and a maximum phase part. For a large filter this can lead to numerous problems. Also, the presence of zeros can lead to a nondecaying, oscillating error.

1.4. The Ill-Conditioning in Deconvolution

In the previous section, we have seen a number of techniques to perform deconvolution. A legitimate question might be, why so many different techniques to perform such a simple task? Ekstrom[30] and Hunt[31] have shown that numerical deconvolution is inherently ill conditioned. Briefly, to repeat equation 1.2.4 in matrix form,

$$\bar{y} = H \bar{x} \quad 1.4.1$$

where

$$H = \begin{bmatrix} h(0) & 0 & 0 & 0 & 0 \\ h(1) & h(0) & 0 & 0 & 0 \\ . & . & h(0) & 0 & 0 \\ . & . & . & . & 0 \\ . & . & . & . & 0 \\ h(N-P) & h(N-P-1) & . & . & h(0) \end{bmatrix}$$

$$\bar{y} = [y(0), y(1), \dots, y(N)]^T$$

$$\bar{\mathbf{x}} = [\mathbf{x}(0), \mathbf{x}(1), \dots, \mathbf{x}(P)]^T$$

and $N+1$ is the number of observation points and $P+1$ the number of unknowns. Using singular value decomposition, matrix H can be written as

$$H = U \Lambda V^T$$

where U and V^T are square matrices of dimensions N by N and P by P respectively and Λ is the rectangular matrix of singular values of size N by P . The solution for equation 1.4.1 can be written as

$$\bar{\mathbf{x}} = \sum_{i=0}^P \alpha_i \mathbf{v}_i \quad 1.4.2$$

where $\alpha_i = \frac{\bar{\mathbf{y}}^T \mathbf{u}_i}{\lambda_i}$, and $\mathbf{u}_i$ and $\mathbf{v}_i$ are the i -th columns of U and V respectively. This is the solution which would be obtained if we performed the least square minimization of equation error. Now, if there are uncertainties in $\bar{\mathbf{y}}$, the solution along with error can be written as

$$\bar{\mathbf{x}} + \delta \bar{\mathbf{x}} = \sum_{i=0}^P \frac{1}{\lambda_i} [(\bar{\mathbf{y}} + \delta \bar{\mathbf{y}})^T \mathbf{u}_i] \mathbf{v}_i = \sum_{i=0}^P \frac{\bar{\mathbf{y}}^T \mathbf{u}_i}{\lambda_i} \left[1 + \frac{\delta \bar{\mathbf{y}}^T \mathbf{u}_i}{\bar{\mathbf{y}}^T \mathbf{u}_i} \right] \mathbf{v}_i$$

The solution error is

$$\delta \bar{\mathbf{x}} = \sum_{i=0}^P \frac{[\delta \bar{\mathbf{y}}^T \mathbf{u}_i]}{\lambda_i} \mathbf{v}_i$$

The squared norm of error is

$$|\bar{\delta x}|^2 = \sum_i \left| \frac{\bar{\delta y}^T u_i}{\lambda_i} \right|^2 \leq \frac{1}{\lambda_{\min}^2} |\bar{\delta y}|^2 \quad 1.4.3$$

From equations 1.4.1 and 1.4.2

$$|\bar{y}|^2 = \sum_i |\lambda_i|^2 a_i^2 \leq |\lambda_{\max}|^2 \sum_i a_i^2 = |\lambda_{\max}|^2 |\bar{x}|^2$$

Using equations 1.4.3 and 1.4.4

$$\frac{|\bar{\delta x}|^2}{|\bar{x}|^2} \leq \left| \frac{\lambda_{\max}}{\lambda_{\min}} \right|^2 \frac{|\bar{\delta y}|^2}{|\bar{y}|^2}$$

The relative error in $\bar{X}$ is seen to be bounded by the relative error in $\bar{y}$ times the section of maximum to minimum singular value, which is the condition number of H . Now, consider the (i,j) th element of the matrix $H^T H$:

$$\{H^T H\}_{ij} = \sum_{r=0}^{N-P} h(r-i+j) h(r)$$

The matrix $H^T H$ is seen to be a Toeplitz matrix with the first row and column corresponding to the autocorrelation of $h(n)$. It is known [32] that the finite spectra of a Toeplitz matrix is in the range of the Fourier transform of the sequence constructed from its first row and the column. If $H(\omega)$ is the Fourier transform of the analog signal $h(t)$ then the transform of its sampled version $h(n)$ is given by

$$H_s(\omega) = \sum_{r=-\infty}^{\infty} H(\omega + \frac{2\pi r}{T}) \quad 1.4.5$$

where T is the sampling interval.

It follows that the transform of the autocorrelation sequence of $h(n)$ is $|H_s(\omega)|^2$. Thus, if $|H_s(\omega)|^2$ has a maximum M and a minimum m , then $|\lambda_i|^2$ belongs to $[m, M]$ for $i=0, 1, 2, \dots, P$. According to Szego's theorem [32], as the dimension of the matrix $H^T H$ increases the set of the eigenvalues of $H^T H$ can be obtained by evaluating $|H_s(\omega)|^2$ at $(p+1)$ equally spaced points, i.e.,

$$|\lambda_k|^2 \approx |H_s(\omega_k)|^2$$

where $\omega_k = \frac{2\pi k}{PT}$.

A curious effect of this relation is that if $h(t)$ is a low-pass signal then sampling it at a faster rate makes the deconvolution more ill-conditioned. Suppose $|H(\omega)| = 0$ for $|\omega| > \omega_0$. If the sampling interval $T > \frac{2\pi}{\omega_0}$, the adjacent periods in equation 1.4.5 alias more. As T increases, the overlap decreases, $|\lambda|_{\min}^2$ decreases, and $|\lambda|_{\max}^2$ stays essentially constant, giving a more ill conditioned solution. This can be seen in figures 1.1a, 1.1b and 1.1c.

As T' decreases, $m' \rightarrow 0$ and hence $\lambda_{\min} \rightarrow 0$ or the condition number $\frac{|\lambda_{\max}|}{|\lambda_{\min}|} \rightarrow \infty$. On the other hand, if we are already given a sampled sequence $h(n)$ whose spectrum has frequency zeros, then $\lambda_{\min} \approx 0$ and again the condition number is large. This is frequently the situation when $h(t)$ is the impulse response of a system which has transmission zeros.

Figure 1.1(a): Bandlimited Signal Spectrum $|H(\omega)|$

(b): Aliased Signal

(c): Aliased Signal ($T' < T$)

1.5. Aims of This Dissertation

In this chapter, various methods which perform deconvolution have been briefly presented, along with their advantages and disadvantages. This dissertation focuses on the problem of blind deconvolution. The prime motivation for studying approaches to blind deconvolution is the fact that there are instances when one might not know the system function precisely and yet one might need to obtain the input to the system from a knowledge of the output. An example of this is in well logging, where the resistance of various layers of earth down a borehole is measured by a probe. While the characteristics of the probe are determined by its design, there still remains ambiguity on how the probe affects the measured output. It is of interest to develop a technique which essentially bypasses the requirement of precise knowledge of the effect of the probe.

Hayes et al [33] have proposed a new technique called phase-only reconstruction for application to blind deconvolution. We will discuss this technique in depth in Chapter 2 and present refinement on some uniqueness theorems developed by Hayes et al. In Chapter 3 the application of phase-only reconstruction to blind deconvolution will be developed. Some simulation results will be presented to compare various methods.

To make the method efficient, a fast algorithm is developed in Chapter 4. In Chapter 5, we expand on how blind deconvolution can be performed on segmented data. Chapter 6 will discuss some generalizations which enable blind deconvolution of unknown systems with

prescribed symmetries. Some uniqueness theorems are proved and the conditions under which the symmetries can be utilized are discussed. In Chapter 7, we present some concluding remarks and directions for future work.

CHAPTER 2

THEORY OF PHASE-ONLY SIGNAL RECONSTRUCTION

2.1. Introduction

It has been shown by Oppenheim et al [34] that the phase of the Fourier Transform of a signal contains substantial information regarding the signal. Indeed, if we have a two-dimensional signal or a picture then a signal reconstructed from the phase of its Fourier Transform, with its magnitude set to unity everywhere, has clearly recognizable contours and many of the details are preserved. It is natural to ask under what conditions the original signal can be completely recovered from knowledge only of its phase. It is a well known fact [35, 36] that for a minimum phase signal the phase and the log-magnitude are Hilbert transforms of each other. Thus from the phase the log-magnitude can be computed within a scale factor and hence the original signal can be recovered. Hayes et al [33, 34, 37] have shown that under a different set of conditions the phase and the signal are again uniquely related, even if the signal is not minimum phase. In the following sections this relationship will be investigated in detail with a view to its application to blind deconvolution.

2.2. Theory of Signal and Phase Relationships

In the following and all the subsequent discussions, only real finite length sequences will be considered. First, consider the minimum phase sequences. Let $x(n)$ $n=0,1,\dots,M$ be a real minimum phase sequence. Its Fourier Transform is defined as

$$X(\omega) = \sum_{n=0}^M x(n) e^{-jn\omega} \quad 2.2.1$$

In general the summation is taken over all the non-zero values of the sequence $x(n)$. The above equation is a polynomial in $e^{-j\omega}$ and can be factored as

$$X(\omega) = |X(\omega)| \exp(j\Phi(\omega)) = x(0) \prod_{r=1}^M (1 - z_r e^{-j\omega})$$

where $|z_r| < 1$, and $|X(\omega)|$ and $\Phi(\omega)$ are the magnitude and the phase respectively. Taking the complex logarithm of both the sides gives

$$\ln |X(\omega)| + j\Phi(\omega) = \ln(x(0)) + \sum_{r=1}^M \ln(1 - z_r e^{-j\omega}).$$

Using the identity

$$\ln(1-t) = - \sum_{r=1}^{\infty} \frac{t^r}{r}$$

for $|t| < 1$ we get

$$\ln |X(\omega)| + j\Phi(\omega) = \ln x(0) - \sum_{r=1}^M \sum_{n=1}^{\infty} \frac{z_r^n}{n} e^{-jn\omega}$$

$$= \ln x(0) - \sum_{n=1}^{\infty} \frac{\sum_{r=1}^M z_r^n}{n} e^{-jn\omega}$$

$$= \ln x(0) - \sum_{n=1}^{\infty} \frac{S_n}{n} e^{-jn\omega}$$

where $S_n = \sum_{r=1}^M z_r^n$. The terms $\frac{S_n}{n}$ are known as the cepstral coefficients of the signal $x(n)$. Equating the real and imaginary parts

$$\ln|X(\omega)| = \ln|x(0)| - \sum_{n=1}^{\infty} \frac{S_n}{n} \cos(n\omega) \quad 2.2.2a$$

$$\Phi(\omega) = \Phi_{x(0)} + \sum_{n=1}^{\infty} \frac{S_n}{n} \sin(n\omega) \quad 2.2.2b$$

where $\Phi_{x(0)} = 0$ or π for $x(0) \geq 0$ or < 0 respectively. The cepstral coefficients can be found from equation (2.2.2b) by inverse sine transform of $\Phi(\omega)$ and can be used to compute $\ln|X(\omega)|$ via equation (2.2.2a) except for the term $\ln|x(0)|$. Thus from the knowledge of $\Phi(\omega)$ we can compute $\ln|X(\omega)|$ up to a constant and hence compute the sequence $x(n)$, $n=0,1,\dots,M$ up to a constant scale factor.

Now we consider another situation. Let sequence $x(n)$, $n=0,1,\dots,M$ be real finite length sequence which may or may not be minimum phase. Its z -transform is defined to be

$$X(z) = \sum_{n=0}^M x(n) z^{-n}.$$

If there exists a zero z_0 of $X(z)$ such that z_0^{-1} is also a zero of $X(z)$ then $X(z)$ or equivalently the sequence $x(n)$ is said to have a

symmetric factor. If $X(z)$ has no symmetric factor then it can be shown that the phase $\Phi(\omega)$ of $X(\omega)$ and the sequence $x(n)$ are uniquely related. For this we need a lemma. As a preface to the lemma, consider an $(M+1)$ length sequence $g(n)$, $n=0,1,2,\dots,M$. The sequence $g(n)$ is said to be symmetric if $g(n) = g(M-n)$ for all n . The center of symmetry is $n_c = \frac{M}{2}$. For an odd length sequence, n_c is an integer. For an even length sequence, n_c is a fraction reflecting the fact that the center of symmetry lies half way between $\frac{(M-1)}{2}$ and $\frac{(M+1)}{2}$. The statement of the lemma now follows [38].

Lemma 2.1. Let $g(n)$ be a real sequence. Then its Fourier Transform

$$G(e^{j\omega}) = \sum_{n=0}^M g(n)e^{-jn\omega}$$

is said to have linear phase iff $g(n)$ is a symmetric sequence.

The definition of linear phase used here is

$$\text{phase of } G(e^{j\omega}) = -n_c \omega + \Theta(\omega)$$

where $\Theta(\omega)$ is a piecewise constant function taking values 0 or $\pm\pi$. The proof of the above lemma is detailed in Appendix A. The zeros of a symmetric sequence exhibit a reciprocal symmetry. This is true for real zeros at ± 1 which are their own inverses as well. The constant n_c is also known as the group delay of $g(n)$. Thus if an odd length symmetric sequence $g(n)$ is advanced by n_c samples, i.e. a new sequence $\tilde{g}(n) = g(n+n_c)$ is constructed, then the Fourier Transform of this new sequence will have its phase equal to $\Theta(\omega)$. Sequence $\tilde{g}(n)$ is also known as a zero phase sequence. By virtue of Lemma 2.1 any zero

phase sequence is also a symmetric sequence with its center of symmetry at index $n=0$.

The following theorem will show the uniqueness of relationship between a finite real sequence and its phase. Define $\Phi_x(\omega)$ to be the phase of the Fourier Transform of the real sequence $x(n)$ evaluated at frequency ω .

Theorem 2.1: Let $x(n)$, $n=0,1,2,\dots,M$ and $y(n)$, $n=0,1,2,\dots,N$ be two finite length sequences such that $x(n)$ does not have any symmetric factor. If

a) $x(n)$ and $y(n)$ have the same length ($M = N$) or

b) $y(n)$ has no symmetric factor

and further

$$\Phi_x(\omega) = \Phi_y(\omega) \text{ for } 0 < \omega < \pi$$

then there exists a positive constant α such that

$$x(n) = \alpha y(n) \text{ for all } n.$$

If instead,

$$\tan(\Phi_x(\omega)) = \tan(\Phi_y(\omega)) \text{ for } 0 < \omega < \pi$$

then there exists a real constant (positive or negative) α

$$x(n) = \alpha y(n) \text{ for all } n.$$

Proof :

Construct a new sequence $s(n)$ by the discrete convolution

$$s(n) = \sum_{r=0}^M y(r-n) x(r) = y(-n) * x(n) \quad 2.2.3$$

for $n=-N, \dots, -1, 0, 1, \dots, M$. The discrete convolution is symbolised by

"*". Taking the Fourier Transform of the above equation we get

$$S(\omega) = \bar{Y}(\omega)X(\omega) \text{ for all } \omega$$

where $\bar{Y}(\omega)$ is the complex conjugate of $Y(\omega)$. Thus,

$$\text{phase of } \bar{Y}(\omega) = \Phi_y(-\omega) = -\Phi_y(\omega).$$

and the overall phase of $S(\omega)$ is

$$\Phi_s(\omega) = \Phi_x(\omega) - \Phi_y(\omega)$$

for all ω . If $\Phi_x(\omega) = \Phi_y(\omega)$ then

$$\Phi_s(\omega) = 0 \text{ for all } \omega.$$

By Lemma 2.1, the sequence $s(n)$ is a symmetric odd sequence. If, on the other hand

$$\tan(\Phi_y(-\omega)) = \tan(\Phi_y(\omega))$$

then

$$\tan(\Phi_s(\omega)) = \frac{\tan\Phi_x(\omega) - \tan\Phi_y(\omega)}{1 + \tan\Phi_x(\omega) \tan\Phi_y(\omega)} = 0$$

for all ω . Hence

$$\Phi_s(\omega) = 0 \text{ or } \pi$$

for all ω and, again by Lemma 2.1, $s(n)$ is a symmetric odd length sequence. Thus, the z -transform of $s(n)$

$$S(z) = \sum_n s(n) z^{-n}$$

is symmetric in z , i.e. $S(z) = S(z^{-1})$, and

$$S(z) = X(z) Y(z^{-1}) = S(z^{-1}) = X(z^{-1}) Y(z).$$

Now,

$$X(z) \mid S(z^{-1}) = X(z^{-1}) Y(z)$$

where " $\mid$ " denotes "divides". Further, since $X(z)$ has no symmetric factors i.e. $X(z)$ and $X(z^{-1})$ are mutually prime, it follows that

$$X(z) \mid Y(z).$$

If, as in case (a), $x(n)$ and $y(n)$ are of the same length, then $X(z)$ and $Y(z)$ are two polynomials in z^{-1} with the same degree. Hence,

$$X(z) = \alpha Y(z)$$

where α is a real constant.

If, on the other hand, as in case (b), $Y(z)$ has no symmetric factor, then by an argument similar that used to prove $X(z) \mid Y(z)$ it can be shown that $Y(z) \mid X(z)$ and hence

$$X(z) = \alpha Y(z).$$

Thus, in any case,

$$x(n) = \alpha y(n) \text{ for } n = 0, \dots, M.$$

Further,

$$S(\omega) = X(\omega) Y(-\omega) = \alpha |Y(\omega)|^2.$$

If $\Phi_s(\omega) = 0$ for all ω then $S(\omega) > 0$ and $\alpha > 0$. If $\Phi_s(\omega) = \theta(\omega)$ then $S(\omega)$ can be positive or negative and hence α will have the same sign.

Theorem 2.1 proves that if a sequence does not have any symmetric factor (or equivalently any reciprocally located zeros) then it is uniquely related to the phase of its Fourier Transform. Given a phase function it should be possible to compute the corresponding time sequence. However, in most situations only a finite number of phase samples will be available. Thus, to be able to compute the time sequence we must, first, be assured of a similar unique relationship between phase samples and the corresponding time sequence. This is accomplished by the following theorem.

Theorem 2.2: Let $x(n)$, $n=0,1,\dots,M$ be a real discrete sequence of length $(M+1)$ such that it does not have any symmetric factors. Let $y(n)$, $n=0,1,\dots,N$ be another real discrete sequence. Then, if

- a) $y(n)$ is of same length as $x(n)$ or
- b) $y(n)$ has no symmetric factor

and further

$$\Phi_x(\omega_k) = \Phi_y(\omega_k)$$

for $k=1,2,\dots,M$, $0 < \omega_k < \pi$ and $\omega_k \neq \omega_m$ for $k \neq m$ then

$$x(n) = \alpha y(n) \text{ for } n=0,1,\dots,M$$

where α is a positive constant. If, on the other hand,

$$\tan(\Phi_x(\omega_k)) = \tan(\Phi_y(\omega_k))$$

for $k=1,2,\dots,M$, $0 < \omega_k < \pi$ and $\omega_k \neq \omega_m$ for $k \neq m$ then

$$x(n) = \alpha y(n) \text{ for } n=0,1,\dots,M$$

where α is a real (positive or negative) constant.

Proof:

Consider a new sequence $s(n)$ defined as

$$s(n) = \sum_{r=0}^M y(r-n) x(r)$$

for $n=-N\dots 0,1,\dots,M$, and $s(n)$ is zero outside this interval. The Discrete Fourier Transform of $s(n)$ is

$$S(\omega_k) = X(\omega_k) \bar{Y}(\omega_k)$$

where $0 < \omega_k < \pi$, $k=1,2,\dots,M$, ω_k are distinct frequencies and $\bar{Y}(\omega_k)$ is the complex conjugate of $Y(\omega_k)$. Then, as in Theorem 2.1, in either the event that $\Phi_x(\omega_k) = \Phi_y(\omega_k)$ or that $\tan(\Phi_x(\omega_k)) = \tan(\Phi_y(\omega_k))$ it can be shown that $S(\omega_k)$ is a real valued function at the frequencies ω_k . Consequently,

$$\text{Im}(S(\omega_k)) = \sum_{n=-M}^N s(n) \sin(n\omega_k) = 0.$$

Rearranging the above equation

$$\sum_{n=1}^{\max(M,N)} (s(n) - s(-n)) \sin(n\omega_k) = 0$$

for $k=1,2,\dots,M$. Define the following vector

$$V_n = [\sin(n\omega_1) \sin(n\omega_2) \dots \sin(n\omega_M)]^T.$$

for $n=1,2,\dots,\max(M,N)$. Then the set of vectors $\{V_n\}$ is linearly independent and hence

$$s(n)-s(-n)=0, \text{ for } n=1,2,\dots,\max(M,N).$$

or $s(n)$ is a symmetric sequence. Following the argument of Theorem 2.1 we can prove that

$$x(n) = \alpha y(n)$$

for all n and appropriate α .

2.3. A Conceptual Technique for Computing a Signal from its Phase

Let $x(n)$, $n=0,1,\dots$ be any real sequence. Its Fourier Transform can be written as

$$X(\omega) = \frac{K e^{-jN\omega} \prod_{r=1}^n (1 - a_r e^{-j\omega}) \prod_{r=1}^m (1 - b_r e^{j\omega})}{\prod_{r=1}^p (1 - c_r e^{-j\omega}) \prod_{r=1}^q (1 - d_r e^{j\omega})} \quad 2.3.1$$

where $|a_r|, |b_r|, |c_r|$ and $|d_r|$ are less than 1. The quantities a_r and c_r are the zeros and poles respectively, inside the unit circle and b_r^{-1} and d_r^{-1} are the zeros and poles, respectively, outside the unit circle. Assuming, without loss of generality, $K > 0$ and taking the complex logarithm of equation 2.3.1

$$\begin{aligned}
\ln(X(\omega)) &= \ln K - jN\omega + \sum_{r=1}^n \ln(1 - a_r e^{-j\omega}) + \sum_{r=1}^m \ln(1 - b_r e^{j\omega}) \\
&\quad - \sum_{r=1}^p \ln(1 - c_r e^{-j\omega}) - \sum_{r=1}^q \ln(1 - d_r e^{j\omega}). \quad 2.3.2
\end{aligned}$$

$$= \ln K - jN\omega + \sum_{r=1}^{\infty} \frac{S_r}{r} \cos r\omega - j \sum_{r=1}^{\infty} \frac{G_r}{r} \sin r\omega$$

where

$$S_r = \sum_{i=1}^n a_i^r + \sum_{i=1}^m b_i^r - \sum_{i=1}^p c_i^r - \sum_{i=1}^q d_i^r$$

and

$$G_r = \sum_{i=1}^n a_i^r - \sum_{i=1}^m b_i^r - \sum_{i=1}^p c_i^r + \sum_{i=1}^q d_i^r.$$

Let $\frac{S_r}{r}$ and $\frac{G_r}{r}$ be denoted the real and the imaginary cepstral coefficients, respectively. It is interesting to note that if there is a pair of reciprocally located zeros or poles the corresponding powers cancel in the expression of G_r . Hence, the imaginary cepstral coefficient G_r does not contain any information about symmetric factors.

Let the sequence $x(n)$ be of finite length then in equation 2.3.1 $c_r = d_r = 0$ and the denominator is unity. If the unwrapped phase $\Phi_x(\omega)$ is known, then it can be expanded in a sine expansion as

$$\Phi_x(\omega) = -N\omega - \sum_{r=1}^{\infty} \frac{\tilde{G}_r}{r} \sin(r\omega)$$

where $\tilde{G}_r = \sum_{i=1}^n a_i^r - \sum_{i=1}^m b_i^r$. Assuming that the terms $\frac{\tilde{G}_r}{r}$ are the cepstral

coefficients of a minimum phase sequence $\tilde{x}(n)$ or equivalently setting the real and imaginary cepstral coefficient to be equal, we can compute $\tilde{x}(n)$ as seen in Section 2.2. Note that it is assumed that the value of N is known independently or can somehow be determined. The Fourier Transform of $\tilde{x}(n)$ is

$$\tilde{X}(\omega) = \tilde{K} \frac{\prod_{i=1}^n (1 - \tilde{a}_i e^{-j\omega})}{\prod_{i=1}^p (1 - \tilde{c}_i e^{-j\omega})}$$

where $\tilde{K}$ is an undetermined constant.

To see how the new minimum phase sequence $\tilde{x}(n)$ relates to $x(n)$ we compare $\tilde{G}_r$ with S_r and G_r of equation 2.3.2. The equating of $\frac{\tilde{G}_r}{r}$ to the real cepstral coefficient $\frac{\tilde{S}_r}{r}$ is the same as assuming that

$$\tilde{a}_i = a_i \text{ and } \tilde{c}_i = b_i$$

which in turn implies that

$$\tilde{X}(\omega) = \tilde{K} \frac{\prod_{i=1}^n (1 - a_i e^{-j\omega})}{\prod_{i=1}^p (1 - b_i e^{-j\omega})}$$

It is important to observe that in the equation above, the phase of a signal treats the zeros outside of the unit circle as poles at the reciprocal location inside the unit circle. This is the crucial property on which Phase-Only reconstruction is based. If the original finite length sequence $x(n)$ has its Fourier Transform written as

equation 2.3.1

$$X(\omega) = X_i(\omega) X_o(\omega)$$

where

$$X_i(\omega) = \prod_{i=1}^n (1 - a_i e^{-j\omega})$$

$$X_o(\omega) = \prod_{i=1}^m (1 - b_i e^{j\omega})$$

then

$$\tilde{X}(\omega) = \tilde{K} \frac{X_i(\omega)}{X_o(-\omega)}.$$

Assuming $\tilde{K} = 1$, the generalized Prony's method can be used [39] to solve for $X_i(\omega)$ and $X_o(\omega)$ separately and hence obtain $x(n)$ up to a constant.

There are two reasons why the method outlined above should not be used for recovering a sequence from its phase. Firstly, in most cases the phase is not available in an unwrapped form. Phase unwrapping is well known for its difficulty [40]. Secondly, the cepstral coefficients involve the powers of the zeros of the sequence and those zeros with magnitude smaller than unity die out rapidly, leading to numerical problems of recovering them. In next and subsequent sections we present other methods more suitable for computational purposes.

2.4. Mathematical Relation Between a Sequence and its Phase

In the previous section it was shown that it is possible to develop a constructive but not a practical way to compute a signal from the phase of its Fourier Transform under certain conditions. In this section, we develop and investigate a practical alternative relationship for computing the signal from the phase.

Consider a real sequence $x(n)$, $n=0,1,\dots$ whose Fourier Transform is

$$X(\omega) = \sum_n x(n) e^{-jn\omega}. \quad 2.4.1$$

We assume that the z-transform of $x(n)$ does not have any zeros at reciprocal locations or on the unit circle, i.e. $x(n)$ does not have any symmetric factors. Then

$$\frac{X(\omega)}{X(-\omega)} = \frac{\sum_n x(n) e^{-jn\omega}}{\sum_n x(n) e^{jn\omega}} = \frac{M_x(\omega) e^{j\Phi_x(\omega)}}{M_x(\omega) e^{-j\Phi_x(\omega)}} \quad 2.4.2$$

where $M_x(\omega)$ and $\Phi_x(\omega)$ are the magnitude and phase of $X(\omega)$ respectively. Assume $x(0)$ to be unity. After canceling $M_x(\omega)$ and cross-multiplying and rearranging the result we get

$$\sum_n x(n) e^{-j(n\omega + \Phi_x(\omega))} = \sum_n x(n) e^{j(n\omega + \Phi_x(\omega))} \quad 2.4.3$$

The above equation is equivalent to

$$\sum_{n \geq 1} x(n) \sin(n\omega + \Phi_x(\omega)) = -\sin(\Phi_x(\omega)) \quad 2.4.4$$

where $x(0) = 1$. If $\Phi_x(\omega)$ is known for M distinct values of ω between 0 and π , then assuming the length of sequence $x(n)$ to be $(M+1)$, equation 2.4.4 gives the following system of linear equations in M unknowns

$$\sum_{n=1}^M x(n) \sin(n\omega_k + \Phi_x(\omega_k)) = -\sin(\Phi_x(\omega_k)) \quad 2.4.5$$

where $k=1,2,\dots,M$. The values of $\omega = 0$ or π do not give any additional information for real sequences $x(n)$ since equation 2.4.5 reduces to the trivial equality $0 = 0$.

For future reference, the coefficient matrix of the above system of equations will be called "sine matrix". This system can be solved exactly for M unknowns by any standard technique.

The system of equations in 2.4.5 is derived from frequency domain considerations. In what follows, it will be shown that it is possible to reformulate the problem from time domain considerations. This gives rise to an interesting system of equations which is capable of solution in a very efficient manner.

We start from equation 2.4.3. Let $e^{j\Phi_x(\omega)}$ be expanded in a Fourier series

$$e^{j\Phi_x(\omega)} = \sum_{n=-\infty}^{\infty} \phi(n) e^{jn\omega}. \quad 2.4.6$$

Taking the inverse Fourier Transform of equation 2.4.3 and using equation 2.4.6 we get

$$\sum_{r=0}^M \phi(-n+r)x(r) = \sum_{r=0}^M \phi(n+r)x(r)$$

for all integers n . The normalization $x(0) = 1$ yields

$$\sum_{r=1}^M \{\phi(-n+r) - \phi(n+r)\}x(r) = \phi(n) - \phi(-n) \quad 2.4.7$$

for all n . Above system of equation for $n=1,2,\dots,M$ can be solved for M unknowns. This system of equations has a coefficient matrix which can be represented as a sum or difference of a Toeplitz and a Hankel matrix and will be denoted as a TH (Toeplitz plus or minus Hankel) matrix. A Toeplitz matrix is a matrix whose (m,n) -th element depends on the difference $(m-n)$ whereas the (m,n) -th element of a Hankel matrix depends on the sum $(m+n)$ only. It can be shown [41, 42] that any system of equations with the coefficient matrix representable as a TH matrix can be converted to a block-Toeplitz matrix which can be solved very efficiently [43] by a block-Levinson type recursion algorithm. This will be presented in detail in Chapter 4.

The expression in equation 2.4.6 assumes that the quantities $\phi(n)$ are known somehow. However, if inverse DFT is used to compute these quantities, what we actually get is the aliased version

$$\bar{\phi}(n) = \sum_{r=-\infty}^{\infty} \phi(n+Lr)$$

where L is the length of DFT. However, the equation 2.4.7 still holds with $\phi(n)$ replaced by $\bar{\phi}(n)$ as long as $L > 2M$.

As an illustration of the theory in this section an example of signal reconstruction from knowledge of only the phase was attempted

on the signal in Figure 2.1. The wrapped phase used in the calculation is shown in Figure 2.2 and the reconstructed signal is in Figure 2.3.

Going back to equation 2.4.5, it is certainly possible to have more phase samples than the number of unknowns involved. To utilize this additional information efficiently, especially if there is uncertainty in these values, one could use a least-square error criterion to obtain the solution. This will be dealt with in next section.

2.5. Signal Reconstruction - Least Squared Error Criterion

Consider an overdetermined version of the system of equations in 2.4.5

$$\sum_{n=1}^M x(n) \sin(n\omega_k + \Phi_x(\omega_k)) = -\sin(\Phi_x(\omega_k)) \quad 2.5.1$$

for $k=1,2,\dots,N-1$ and $N-1 > M$. The above system of equations can be written as

$$V \bar{x} = \bar{b}$$

where V is the sine coefficient matrix and $\bar{x}$ is the vector of M unknowns

$$\bar{x} = [x(1) \ x(2) \ \dots x(M)]^T$$

and

$$\{\bar{b}\}_k = -\sin(\Phi_x(\omega_k)).$$

Figure 2.1: Original Signal $x(n]$

Figure 2.2: Phase of $X(\omega)$

Figure 2.3: Reconstructed Signal $x(n)$

To obtain the solution which minimizes the equation error $\bar{b} - V \bar{x}$ the following system of normal equations needs to be solved

$$V^T V \bar{x} = V^T \bar{b}. \quad 2.5.2$$

It is of interest to see the form which the matrix $V^T V$ takes. It can be shown that the resulting form of the equation can be represented as a sum of a Toeplitz and a Hankel matrix i.e. TH type which, as seen before, can be solved efficiently.

The (m,n) -th element of the normal matrix $V^T V$ is

$$\begin{aligned} \sum_{k=1}^{N-1} \sin(m\omega_k + \Phi_x(\omega_k)) \sin(n\omega_k + \Phi_x(\omega_k)) &= \frac{1}{2} \sum_{k=1}^{N-1} \cos((m-n)\omega_k) \\ &- \frac{1}{2} \sum_{k=1}^{N-1} \cos((m+n)\omega_k + 2\Phi_x(\omega_k)). \end{aligned} \quad 2.5.3$$

The first summation on the right of equation 2.5.3 is independent of the phase $\Phi_x(\omega_k)$ and, further, for a given set of ω_k 's, it depends only on the difference between m and n . Similarly, the second term, for a given set of ω_k 's and the corresponding phase values $\Phi_x(\omega_k)$ depends only on the sum of m and n . Denoting these two terms by $t(m-n)$ and $h(m+n)$ respectively, the (m,n) -th term of $V^T V$ is

$$\{ V^T V \}_{m,n} = t(m-n) - h(m+n). \quad 2.5.4$$

Thus, $V^T V$ is seen to be a TH type matrix.

The right hand side of equation 2.5.2 is

$$\{ V^T \bar{b} \}_m = - \sum_{k=1}^{N-1} \sin(m\omega_k + \Phi_x(\omega_k)) \sin(\Phi_x(\omega_k)) \quad 2.5.5$$

which, after comparison with 2.5.3 and 2.5.4, can be written as

$$\{V^{\tau\bar{b}}\}_m = -t(m) + h(m). \quad 2.5.6$$

In most signal processing application the Discrete Fourier Transform (DFT) is computed by Fast Fourier Transform (FFT) and consequently the phase is available at uniformly spaced values of ω . Consider a situation when $\omega_k = \frac{2\pi k}{2N}$ with $2N$ being the length of FFT. Using the fact that

$$\Phi_x(0) = 0 \text{ or } \pi, \quad \Phi_x(\omega_N) = \Phi_x(\pi) = 0 \text{ or } \pi$$

and

$$\Phi_x(-\omega_k) = \Phi_x(\omega_{2N-k})$$

for any integer r , $0 \leq r < 2N$,

$$\begin{aligned} \sum_{k=1}^{N-1} \cos(r\omega_k + \Phi_x(\omega_k)) &= \frac{1}{2} \sum_{k=0}^{2N-1} e^{j(r\omega_k + 2\Phi_x(\omega_k))} - \frac{1}{2} - \frac{1}{2} e^{j(r\omega_N + 2\Phi_x(\omega_N))} \\ &= -\frac{1}{2} \{ 1 + (-1)^r \} + N \cdot \frac{1}{2N} \sum_{k=0}^{2N-1} e^{jr\omega_k} e^{j(2\Phi_x(\omega_k))}. \end{aligned}$$

Note that the summation on the right hand side of the above equation is an inverse DFT of a conjugate symmetric sequence and hence a real quantity, which will be denoted $\eta(r)$. Then,

$$\sum_{k=1}^{N-1} \cos(r\omega_k + 2\Phi_x(\omega_k)) = \begin{cases} N \cdot \eta(r) - 1 & \text{for } r \text{ even} \\ N \cdot \eta(r) & \text{for } r \text{ odd} \end{cases} \quad 2.5.7$$

where

$$\eta(r) = \frac{1}{2N} \sum_{k=0}^{2N-1} e^{j r \omega_k} e^{j(2\Phi_x(\omega_k))}.$$

Similarly, for any r , $0 \leq r < 2N$,

$$\sum_{k=1}^{N-1} \cos(r\omega_k) = \begin{cases} N-1 & \text{for } r=0 \\ -1 & \text{for } r \text{ even } \neq 0 \\ 0 & \text{for } r \text{ odd.} \end{cases}$$

Now, consider any two integers m and n . Since $(m+n) = (m-n) + 2n$ it follows that

$$(m-n) \text{ is even (odd) iff } (m+n) \text{ is even (odd) .}$$

Using the above relationships along with equation 2.5.8 we get

$$\{V^T V\}_{m,n} = \begin{cases} N - N \cdot \eta(2m) & m=n \\ - N \cdot \eta(m+n) & m \neq n \end{cases} \quad 2.5.8$$

and similarly with equation 2.5.9 we get

$$\{V^T \bar{b}\}_m = \begin{cases} N - N \cdot \eta(0) & m=0 \\ - N \cdot \eta(m) & m \neq 0 \end{cases} \quad 2.5.9$$

Comparison of equations 2.5.4 and 2.5.6 with equations 2.5.8 and 2.5.9 respectively yields the identification

$$t(m) = \delta(m) = \begin{cases} 1 & \text{for } m=0 \\ 0 & \text{for } m \neq 0 \end{cases}$$

and

$$h(m) = \eta(m) \text{ for all } m$$

where we have divided equations 2.5.8 and 2.5.9 by N . The function $\delta(\cdot)$ is the Kronecker delta function. Thus we need to solve the following system of equations

$$\sum_{n=1}^M \{ \delta(m-n) - \eta(m+n) \} x(n) = \eta(m) \quad 2.5.10$$

for $m=1,2,\dots,M$. In the above system of equations, which is of the TH type, the terms $\eta(m)$ can be obtained by inverse FFT. Then 2.5.10 can be solved by the block-Toeplitz matrix inversion technique.

Instead of minimizing the squared norm of the equation error of 2.5.1, which was derived from frequency domain considerations, we can minimize the norm of the equation error of 2.4.7. This derivation is detailed in Appendix B. By Parseval's Theorem, the solution obtained in either case is the same. However, it is interesting to observe that the system of equations obtained in both cases are identical except for a scale constant.

It is further interesting to observe that equation 2.5.10 is reminiscent of the time domain formulation equation 2.4.7. Indeed, it is possible to obtain 2.5.10 more directly by rewriting 2.4.3 in the form

$$X(\omega) = e^{j2\Phi_x(\omega)} x(-\omega). \quad 2.5.11$$

Let

$$e^{j2\Phi_x(\omega_k)} = \sum_{n=0}^{2N-1} \eta(n) e^{-jn\omega_k}$$

where $\omega_k = \frac{2\pi k}{2N}$ for $k=0,1,\dots,(2N-1)$, then $\eta(n)$ is as defined in equation 2.5.7. Taking the inverse DFT of 2.5.11 we get

$$x(m) = \sum_{n=0}^M \eta(n+m) x(n) \quad 2.5.12$$

for $m=1,2,\dots,N$. Using $x(0) = 1$, 2.5.12 becomes

$$x(m) - \sum_{n=1}^M \eta(n+m)x(n) = \eta(m)$$

for $m=1,2,\dots,M$, which is identical to the system of equations obtained from 2.5.10.

2.6. Utilization of Known Section of Sequence

Suppose that a part of the sequence $x(n)$ is known. Then this knowledge can be used to reduce the amount of computation needed to solve for the rest of the sequence. Let $x(0), x(1), \dots, x(q-1)$ be the known portion and $x(q), x(q+1), \dots, x(M)$ be the unknown portion. Then equation 2.4.5 becomes

$$\sum_{n=q}^M x(n) \sin(n\omega_k + \Phi_x(\omega_k)) = - \sum_{n=0}^{q-1} x(n) \sin(n\omega_k + \Phi_x(\omega_k)) \quad 2.6.1$$

and equation 2.4.7 becomes

$$\sum_{r=q}^M \{\phi(-k+r) - \phi(k+r)\}x(r) = - \sum_{r=0}^{q-1} \{\phi(-k+r) - \phi(k+r)\}x(r) \quad 2.6.2$$

for $k=1,2,\dots,(M-q+1)$. The above equation is of the TH type. Equation 2.5.12 can be similarly rearranged to solve for only the unknown portion of the sequence and this also yields a TH type system. Alternatively, a shifted version of sequence $x(n)$ can be considered. For this equation 2.4.3 can be rewritten as

$$\{X_q(\omega) + e^{-jq\omega} \underline{X}(\omega)\} e^{-j\Phi_x(\omega)} = \{X_q(-\omega) + e^{jq\omega} \underline{X}(-\omega)\} e^{j\Phi_x(\omega)}$$

where

$$X_q(\omega) = \sum_{n=0}^{q-1} x(n) e^{-jn\omega}$$

and

$$\underline{X}(\omega) = \sum_{n=q}^M x(n) e^{-j(n-q)\omega}.$$

Rearranging the above equation so that the known quantities are on the right hand side and taking the inverse DFT we get the system of equations in 2.6.2.

In all the previous discussion, it has been assumed that the sequence $x(n)$ which is to be reconstructed from the phase has a finite support i.e. is of finite length. The problem can be viewed slightly differently viz. that we are trying to compute a finite number of unknowns. This leads to an interesting variation of the known segment problem dealt with earlier in this section.

Consider a sequence $x(n)$ with a semi-infinite support. Let

$$x(n) \begin{cases} = \text{constant} & -\infty < n \leq 0 \\ \neq \text{constant} & 1 \leq n \leq M \end{cases}$$

then we can write { cf. 2.4.2 }

$$\frac{X(\omega)}{X(-\omega)} = \frac{\frac{1}{1 - e^{j\omega}} + \sum_{n=1}^M x(n) e^{-jn\omega}}{\frac{1}{1 - e^{-j\omega}} + \sum_{n=1}^M x(n) e^{jn\omega}} = \frac{M_x(\omega) e^{j\Phi_x(\omega)}}{M_x(\omega) e^{-j\Phi_x(\omega)}} \quad 2.6.3$$

The above ratio is well defined for all values of ω except $\omega = 0$. However, it has been noted earlier in Section 2.4 that $\omega = 0$ does not provide any information. Thus, equation 2.6.3 is well defined for all

the values of ω of interest and hence can be rewritten as

$$\sum_{n=1}^M x(n) \sin(n\omega_k + \Phi_x(\omega_k)) = - \operatorname{Im} \frac{e^{-j\Phi_x(\omega_k)}}{1 - e^{j\omega_k}} \quad 2.6.4$$

$$= - \frac{\cos(\frac{\omega_k}{2} + \Phi_x(\omega_k))}{2 \sin(\frac{\omega_k}{2})}.$$

The system of equations obtained from 2.6.4 has the same form of coefficient matrix as the sine matrix of 2.4.5. The time domain version of 2.6.4 is a TH system similar to that of 2.4.7, except for the right hand side, which is slightly different and can be obtained by the inverse DFT of the sequence $p(k)$ where

$$p(k) = \begin{cases} 0 & \text{for } k=0 \\ - \frac{\cos(\frac{\omega_k}{2} + \Phi_x(\omega_k))}{2 \sin(\frac{\omega_k}{2})} & \text{for } k \neq 0 \end{cases}.$$

2.7. Reconstruction Error Due to Phase Uncertainties

Consider a situation where the signal $x(n)$ is to be from a phase term which contains an error function $\Phi_e(\omega)$. Let the solution due to $\Phi_x(\omega)$ be $x(n)$ and the one due to the correct phase $\Phi_x(\omega) - \Phi_e(\omega)$ be $\tilde{x}(n)$. Then, $\tilde{x}(n)$ satisfies the equation

$$e^{-j(\Phi_x(\omega) - \Phi_e(\omega))} \tilde{X}(\omega) = e^{j(\Phi_x(\omega) - \Phi_e(\omega))} \tilde{X}(-\omega).$$

Comparing the equation above with equation 2.4.3

$$X(\omega) = \frac{1}{\mu} e^{j\Phi_\varepsilon(\omega)} \tilde{X}(\omega)$$

where

$$\mu = \frac{1}{2\pi} \int_{-\pi}^{\pi} \tilde{X}(\omega) e^{j\Phi_\varepsilon(\omega)} d\omega.$$

The constant μ ensures that $x(0) = 1$. The squared error norm is

$$\begin{aligned} \sum_n (x(n) - \tilde{x}(n))^2 &= \frac{1}{2\pi} \int_{-\pi}^{\pi} |X(\omega) - \tilde{X}(\omega)|^2 d\omega \\ &= \frac{1}{2\pi} \int_{-\pi}^{\pi} |\tilde{X}(\omega)|^2 \left| 1 - \frac{1}{\mu} e^{j\Phi_\varepsilon(\omega)} \right|^2 d\omega \\ &= \frac{1}{2\pi} \int_{-\pi}^{\pi} |\tilde{X}(\omega)|^2 \left(1 + \frac{1}{\mu^2} - \frac{2}{\mu} \cos(\Phi_\varepsilon(\omega)) \right) d\omega \\ &\leq \frac{1}{2\pi} \int_{-\pi}^{\pi} |\tilde{X}(\omega)|^2 \left(1 + \frac{1}{\mu^2} + \frac{2}{\mu} \left(\frac{\Phi_\varepsilon(\omega)^2}{2} - 1 \right) \right) d\omega \\ &= \frac{1}{2\pi} \int_{-\pi}^{\pi} |\tilde{X}(\omega)|^2 \left\{ \left(1 - \frac{1}{\mu} \right)^2 + \frac{\Phi_\varepsilon(\omega)^2}{\mu} \right\} d\omega \\ &\leq \frac{1}{2\pi} \int_{-\pi}^{\pi} |\tilde{X}(\omega)|^2 d\omega \left\{ \left(1 - \frac{1}{\mu} \right)^2 + \frac{\max \Phi_\varepsilon(\omega)^2}{\mu} \right\} \\ &= \left(\sum_n \tilde{x}(n)^2 \right) \left\{ \left(1 - \frac{1}{\mu} \right)^2 + \frac{\max \Phi_\varepsilon(\omega)^2}{\mu} \right\} \end{aligned}$$

The relative error bound is

$$\frac{\sum_n \{x(n) - \tilde{x}(n)\}^2}{\sum_n \tilde{x}(n)^2} \leq \left\{ \left(1 - \frac{1}{\mu}\right)^2 + \frac{\max_{\omega} \Phi_{\varepsilon}(\omega)^2}{\mu} \right\} \quad 2.7.1$$

Now by assumption

$$1 = \frac{1}{2\pi} \int_{-\pi}^{\pi} \tilde{X}(\omega) d\omega.$$

Using the definition of μ

$$|\mu - 1| = \frac{1}{2\pi} \left| \int_{-\pi}^{\pi} \tilde{X}(\omega) (e^{j\Phi_{\varepsilon}(\omega)} - 1) d\omega \right|.$$

By Cauchy-Schwartz inequality

$$\begin{aligned} |\mu - 1|^2 &\leq \left\{ \frac{1}{2\pi} \int_{-\pi}^{\pi} |\tilde{X}(\omega)|^2 d\omega \right\} \left\{ \frac{1}{2\pi} \int_{-\pi}^{\pi} |e^{j\Phi_{\varepsilon}(\omega)} - 1|^2 d\omega \right\} \\ &\leq \left\{ \sum_n \tilde{x}(n)^2 \right\} \left\{ \frac{1}{2\pi} \int_{-\pi}^{\pi} 4 \sin^2 \left(\frac{\Phi_{\varepsilon}(\omega)}{2} \right) d\omega \right\} \\ &\leq \left\{ \sum_n \tilde{x}(n)^2 \right\} \left\{ \max_{\omega} \Phi_{\varepsilon}(\omega)^2 \right\}. \end{aligned}$$

Using the inequality above in 2.7.1 we get

$$\frac{\sum_n \{x(n) - \tilde{x}(n)\}^2}{\sum_n \tilde{x}(n)^2} \leq \beta \quad 2.7.2$$

where

$$\beta = \left\{ \max_{\omega} \Phi_{\varepsilon}(\omega)^2 \right\} \left\{ \frac{1}{\mu} + \frac{1}{\mu^2} \sum_n \tilde{x}(n)^2 \right\}.$$

On the other hand the Triangular inequality

$$\begin{aligned}
 \sum_n (x(n) - \tilde{x}(n))^2 &= \frac{1}{2\pi} \cdot \int_{-\pi}^{\pi} |X(\omega) - \tilde{X}(\omega)|^2 d\omega & 2.7.3 \\
 &\geq \frac{1}{2\pi} \cdot \int_{-\pi}^{\pi} (|X(\omega)| - |\tilde{X}(\omega)|)^2 d\omega \\
 &= \frac{1}{2\pi} \cdot \int_{-\pi}^{\pi} \left(\frac{|\tilde{X}(\omega)|}{|\mu|} - |\tilde{X}(\omega)| \right)^2 d\omega \\
 &= \frac{1}{2\pi} \cdot \left(\frac{1}{|\mu|} - 1 \right)^2 \int_{-\pi}^{\pi} |\tilde{X}(\omega)|^2 d\omega \\
 &= \left(\frac{1}{|\mu|} - 1 \right)^2 \cdot \sum_n \tilde{x}^2(n)
 \end{aligned}$$

Hence

$$\frac{\sum_n (x(n) - \tilde{x}(n))^2}{\sum_n \tilde{x}(n)^2} \geq \left(\frac{1}{|\mu|} - 1 \right)^2 = \alpha \quad 2.7.4$$

The ratio in equations 2.7.2 and 2.7.4 is the reciprocal of signal energy to error energy. As a figure of merit we will use the ratio of signal energy to error energy which will be denoted as SER. Equations 2.7.2 and 2.7.4 imply that

$$\frac{1}{\alpha} \geq \text{SER} \geq \frac{1}{\beta}.$$

To test the consistency of the above bounds, the signal in the section (2.2) had its phase contaminated by a uniformly distributed noise and the original signal was reconstructed. The plot in

Figure 2.4 shows how SER and the bounds in decibels vary with respect to the largest phase error value.

2.8. Summary of Results in this Chapter

This chapter has discussed in detail the relationships between a signal and its phase. The conditions under which this unique relationship allows computation of the signal from its phase were investigated. It was found that if a signal does not have any symmetric factors then the signal can be reconstructed uniquely.

The theory in this chapter was applied to an example in which the signal was reconstructed from its phase only. It was seen that the reconstructed signal was identical to the original signal.

In the next chapter we present how the theory developed in this chapter can be applied to blind-deconvolution.

Figure 2.4: Reconstruction Error in $x(n)$
Due to Given Max. Phase Error

CHAPTER 3

BLIND DECONVOLUTION USING PHASE-ONLY RECONSTRUCTION

3.1. Introduction

In chapter 2, we discussed the problem of how to relate a sequence and its phase uniquely. Various conditions under which this is possible were presented. In this chapter we will exploit these conditions and the unique relationships to perform blind deconvolution.

3.2. Blind Deconvolution

Let $x(n), n=0,1,\dots,M$ be an input to a filter H with impulse response $h(n)$. Let $y(n)$ be the output of H as depicted in the figure 3.1.

Figure 3.1

The output is given by

$$y(n) = \sum_r h(r) x(n-r) \quad 3.2.1$$

where the summation is over the temporal support of $h(n)$. The problem

of blind deconvolution can be stated as:

Given the output $y(n)$ and the restrictions on $h(n)$ reconstruct the input $x(n)$.

Normally, the impulse response $h(n)$ is known a priori and therefore can be used to deconvolve directly. This has been discussed in chapter 1. When $h(n)$ is not known precisely except for some restrictions we are "blind" with regard to H , hence the term blind deconvolution. To use Phase-Only reconstruction, we consider the restrictions that H be a zero phase or a linear phase filter. Further, let the z -transform of $h(n)$ have no symmetric factors. The Fourier transform of equation 3.2.1 is

$$Y(\omega) = Y(z) \Big|_{z=e^{j\omega}} = H(\omega) X(\omega) \quad 3.2.2$$

Let H be a zero phase filter. Then its impulse response is symmetric and its transform is strictly real.

$$\begin{aligned} H(\omega) &= \sum_{r=-L}^L h(n) e^{-jn\omega} \\ &= h(0) + \sum_{r=1}^L h(n) (e^{-jn\omega} + e^{jn\omega}) \\ &= h(0) + 2 \sum_{r=1}^L h(n) \cos(n\omega) \end{aligned}$$

Thus, equation 3.2.2 can be rewritten

$$M_y(\omega) = M_h(\omega) M_x(\omega)$$

$$\Phi_y(\omega) = \Phi_h(\omega) + \Phi_x(\omega)$$

where $M_y(\omega)$, $M_h(\omega)$, $M_x(\omega)$ and $\Phi_y(\omega)$, $\Phi_h(\omega)$ and $\Phi_x(\omega)$ are the magnitudes and phases of $Y(\omega)$, $H(\omega)$ and $X(\omega)$ respectively at the angular frequency ω . Since $H(\omega)$ is strictly real its phase $\Phi_h(\omega) = 0$ or π for all ω . Thus,

$$\tan(\Phi_y(\omega)) = \tan(\Phi_h(\omega) + \Phi_x(\omega)) = \tan(\Phi_x(\omega)) \quad 3.2.3$$

Since the sequence $x(n)$ is assumed to satisfy the conditions of Theorems 1 and 2 (Chapter 2), and further, since equation 3.2.3 gives us the knowledge of $x(n)$, we can use the methods developed in Chapter 2 to reconstruct the input $x(n)$. Following this is a brief description of the algorithm to compute the input $x(n)$.

Algorithm 1:

Step 1: Compute the Fourier transform

$$Y(\omega) = \sum_n y(n) e^{-jn\omega}$$

Step 2: Compute the phase function

$$\Phi_y(\omega) = \text{phase angle of } Y(\omega)$$

Step 3: Expand $e^{j\Phi_y(\omega)}$ by inverse Fourier Transform in term of powers of $e^{-j\omega}$

$$e^{j\Phi_y(\omega)} = \sum_n \phi(n) e^{-jn\omega}$$

Step 4: Solve for $x(1), x(2), \dots, x(M)$ using equation 2.6.2

$$\sum_{r=1}^M [\phi(-n+r) - \phi(n+r)] x(r) = \phi(n) - \phi(-n),$$

for $n=1,2,\dots,M$.

If a minimum squared error solution is desired, the series of steps above can be modified as follows:

Algorithm 1a:

Step 3a: Expand $e^{j2\Phi_y(\omega)}$ in Fourier series

$$e^{j2\Phi_y(\omega)} = \sum_n \eta(n) e^{-jn\omega}$$

Step 4a: Solve for $x(1), x(2), \dots, x(M)$ using equation 2.5.10

$$\sum_{r=1}^M \{ \delta(n-r) - \eta(n+r) \} x(r) = \eta(n)$$

for $n=1,2,\dots,M$.

3.3. Magnitude Weighted Phase-Only Reconstruction

Until now, we have proceeded by finding an all-pass sequence corresponding to the phase $\Phi_y(\omega)$ (for example, steps 3 and 3a in section 3.2). The approach outlined in the preceding section assumes that the phase $\Phi_y(\omega)$ is known accurately for all ω . This need not be true when the magnitude $|Y(\omega)|$ is small and there are errors in computation of $Y(\omega)$. Figures 3.2a and 3.2b show that for smaller magnitude $M_y(\omega)$ there is greater likelihood of errors in $\Phi_y(\omega)$ for the same error in $Y(\omega)$. One approach to this problem is to have each

Figure 3.2a: Phase Error Due to Error in $Y(\omega_0)$

Figure 3.2b: Phase Error Due to Error in $Y(\omega_1)$

equation (in 2.4.3) involving $\Phi(\omega)$ weighed by the corresponding $M_y(\omega)$. This reduces the contribution of those phase terms which have greater possibility of error.

Recall equation 2.4.3

$$\sum_{n=0}^M x(n) e^{-j(n\omega + \Phi_y(\omega))} = \sum_{n=0}^M x(n) e^{j(n\omega + \Phi_y(\omega))} \quad 3.3.1$$

where $\Phi_x(\omega)$ is replaced by the known phase function $\Phi_y(\omega)$. Multiplying both sides by $M_y(\omega)$.

$$M_y(\omega) e^{-j\Phi_y(\omega)} \sum_{n=0}^M x(n) e^{-jn\omega} = M_y(\omega) e^{j\Phi_y(\omega)} \sum_{n=0}^M x(n) e^{jn\omega}$$

which can be rewritten as

$$Y(-\omega) X(\omega) = Y(\omega) X(-\omega) \quad 3.3.2$$

Taking inverse Fourier transform of 3.3.2

$$\sum_{r=0}^M y(-r+n) x(r) = \sum_{r=0}^M y(r+n) x(r)$$

Using the normalization $x(0) = 1$ the system of equations to be solved is

$$\sum_{r=1}^M [y(-n+r) - y(n+r)] x(r) = y(n) - y(-n) \quad 3.3.3$$

for $n=1,2,\dots,M$. It should be observed that equation 3.3.3 uses the known values of $y(n)$ directly and thus avoids introducing phase errors at an intermediate stage.

It is of interest to view the derivation of equation 3.3.2 from a different viewpoint. Consider equation 3.2.2

$$Y(\omega) = H(\omega) X(\omega)$$

$H(\omega)$ is assumed to have zero phase, i.e. $H(\omega) = H(-\omega)$ Construct a new function

$$Y(\omega) X(-\omega) = H(\omega) X(\omega) X(-\omega) \quad 3.3.4$$

$$= H(\omega) X(-\omega) X(\omega)$$

$$= H(-\omega) X(-\omega) X(\omega)$$

$$= Y(-\omega) X(\omega)$$

Thus equation 3.3.2 has implicit in it the fact that $H(\omega)$ is a zero phase filter.

To distinguish the Phase-Only Blind Deconvolution presented in the previous section from the Magnitude Weighted Phase-Only Blind Deconvolution, the former will be called Unit Magnitude Phase-Only Deconvolution.

3.4. Blind Deconvolution of Linear Phase Filters Using Phase-Only Reconstruction

So far, in the preceding discussion the deconvolution of only a zero phase filter has been considered. In actual situations, we expect the filter $H(\omega)$ to be a linear phase filter and thus we need to reformulate the problem of deconvolution to take this into

account.

3.4.1. Linear Phase Odd Length Filters

Consider an impulse response of an odd length symmetric filter $h(n)$, $n=0,1,\dots,2L$. The length $2L+1$ is assumed to be known. The z -transform of $h(n)$ is

$$\begin{aligned} H(z) &= \sum_{n=0}^{2L} h(n) z^{-n} \\ &= \sum_{n=-L}^L h(n+L) z^{-n-L} \\ &= z^{-L} \sum_{n=-L}^L h(n+L) z^{-n} = z^{-L} \tilde{H}(z) \end{aligned}$$

Since $h(n+L) = h(-n+L)$, $\tilde{H}(z)$ is a symmetric polynomial, i.e.
 $\tilde{H}(z) = \tilde{H}(z^{-1})$

$$\begin{aligned} H(z^{-1}) &= z^L \tilde{H}(z^{-1}) = z^L \tilde{H}(z) \\ &= z^{2L} z^{-L} \tilde{H}(z) = z^{2L} H(z). \end{aligned}$$

Hence,

$$H(-\omega) = H(z^{-1}) \Big|_{z=e^{j\omega}} = e^{j2L\omega} H(\omega)$$

Following the approach of equation 3.2.4, let

$$\begin{aligned} Y(\omega) X(-\omega) &= H(\omega) X(\omega) X(-\omega) \\ &= e^{-j2L\omega} H(-\omega) X(-\omega) X(\omega) \end{aligned}$$

$$= e^{-j2L\omega} Y(-\omega) X(\omega)$$

Thus

$$e^{jL\omega} Y(\omega) X(-\omega) = e^{-jL\omega} Y(-\omega) X(\omega)$$

The time domain version of the above equation is

$$\sum_{r=1}^M [y(-n+r+L) - y(n+r+L)] x(r) = y(n+L) - y(-n+L) \quad 3.4.1$$

for $n=1, 2, \dots, M$.

3.4.2. Linear Phase Even Length Filters

Consider the impulse response of an even length symmetric filter

$$h(n), n=0, 1, \dots, 2L-1 \quad \text{with} \quad h(n) = h(2L-1-n)$$

The z-transform of $h(n)$ is

$$\begin{aligned} H(z) &= \sum_{n=0}^{2L-1} h(n) z^{-n} \\ &= \sum_{n=0}^{2L-1} h(2L-1-n) z^{-n} \\ &= z^{-2L+1} \sum_{n=0}^{2L-1} h(2L-1-n) z^{2L-1-n} \\ &= z^{-(2L-1)} \sum_{n=0}^{2L-1} h(n) z^n = z^{-(2L-1)} H(z^{-1}) \end{aligned}$$

Therefore

$$H(\omega) = H(z) \Big|_{z=e^{j\omega}} = z^{-(2L-1)} H(z^{-1}) \Big|_{z=e^{j\omega}} = e^{-j(2L-1)\omega} H(-\omega)$$

Again using the approach of equation 3.3.4, let

$$Y(\omega) = H(\omega) X(\omega), \quad \text{where} \quad H(\omega) = e^{-j(2L-1)\omega} H(-\omega)$$

Then

$$\begin{aligned} Y(\omega) X(-\omega) &= H(\omega) X(\omega) X(-\omega) \\ &= e^{-j(2L-1)\omega} H(-\omega) X(-\omega) X(\omega) \\ &= e^{-j(2L-1)\omega} Y(-\omega) X(\omega) \end{aligned}$$

or

$$e^{jL\omega} Y(\omega) X(-\omega) = e^{-j(L-1)\omega} Y(-\omega) X(\omega)$$

The time domain version of the above equation is

$$\sum_{r=1}^M [y(-n+r+L) - y(n+r+L-1)] x(r) = y(n+L-1) - y(-n+L) \quad 3.4.2$$

for $n=1,2,\dots,M$.

It is interesting to note that though both equation 3.4.1 and 3.4.2 have a Toeplitz-plus-Hankel structure they are not identical. In equation 3.4.2 the Hankel part of the system is delayed by one sample. This is due to the fact that the center of symmetry for even length $h(n)$ is at a half sample. The effect of advancing the Toeplitz part by a half sample and delaying the Hankel part by the same amount can be effectively achieved by delaying the Hankel part by one sample and keeping the Toeplitz part fixed.

In this section, we have shown how to deal with a linear phase even length filter. Since the mathematical operations involved are simple but somewhat messy, we will deal with linear phase odd length filters in all subsequent discussions in this and future chapters.

3.5. Uncertainties of Filter Length and Signal Length

In all our previous discussions, we have assumed that the support of the input signal or the length of the input signal and the location of the first nonzero sample are known. The knowledge of the location of the first nonzero sample is equivalent to knowing the filter to be either zero phase or a known linear phase. In a real situation, this might not be true. In this section we discuss how the above problem can be tackled.

Let $x(n)$ be a real sequence of length M , convolved with symmetric filter H of length $2L-1$. If we know the value of L but not the value of M then any of the previous techniques starting with $M=1, 2, \dots$ and each time checking the error on the equations not used until it goes below a threshold or the coefficient matrix becomes singular.

When the values of L and M are both unknown the problem becomes more difficult. Equation 2.5.10 provides a possible approach to the solution of this problem. Since this equation is true for all n choose $n \leq -M_0 < 0$ then

$$\sum_{n=1}^M \eta(m+n) x(n) = -\eta(m)$$

for $m=-M_0, -(M_0+1), \dots, -(M_0+M+1)$. Since $x(n)$ satisfy any block of M

equations we can choose M_0 to have any value greater than zero. This is a Toeplitz system of equations and can be solved by Levinson recursion. Now, suppose $\Phi(\omega)$ the phase function has a linear phase term due to the $H(\omega)$ Then we can write

$$\Phi_y(\omega) = \Phi_x(\omega) - \omega L$$

or
$$e^{j\Phi_y(\omega)} = e^{j\Phi_x(\omega)} \cdot e^{-j\omega L}$$

Let
$$e^{j2\Phi_x(\omega)} = \sum_n \eta_x(n) e^{-jn\omega}$$

then
$$e^{j2\Phi_y(\omega)} = \sum_n \eta_x(n-2L) e^{-jn\omega}$$

and the system of equations to be solved becomes .

$$\sum_{n=1}^M \eta_x(-m+n-2L)x(n) = -\eta_x(-m-2L)$$

for $m=1,2,\dots,M$. This can be rewritten as

$$\sum_{n=1}^M \eta_x(-m+n)x(n) = -\eta_x(-m) \quad 3.5.1$$

for $m = -2L-1, -2L, -2L+1, \dots, -2L-M$. But this is equivalent to solving for $x(n)$ using the block of M equations with offset of $M_0 = 2L$.

Again, we can use the Levinson recursion to solve equation 3.5.1. Since the Levinson recursion finds the solution for all values of M starting from $M=1$, in its intermediate steps, we can keep a check on the prediction error or the determinant of the matrix and

when either of them reach the value zero or a certain threshold, the recursion is stopped and the corresponding value of M and the solution $x(n)$ can be assumed to be the desired solution. This follows from the fact that since a unique solution exists the coefficient matrix cannot be singular for any value of M equal to the length of $x(n)$, but for values of M greater than this length, the columns are linearly dependent.

This procedure, however, is not numerically attractive. The value of $\eta_x(n)$ can be shown to be

$$\eta_x(-n) = \sum_{|z_k| > 1} \frac{X(z_k^{-1})}{z_k^{X'(z_k)}} z_k^{-n} \quad \text{for } n > 0$$

where z_k are the zeros of $X(z)$ and $X'(z)$ is the derivative of $X(z)$ with respect to z . When n is large $\eta(-n)$ becomes small, and thus for large but unknown values of L the results can be unreliable.

3.6. Reconstruction Error Due to Measurement Error

So far it has been assumed that the measured values of the output $y(n)$ are accurately known. In reality, some measurement noise will always be present. In this section, we will consider this noise to be modeled as additive zero mean white Gaussian noise $v(n)$ with standard deviation σ . The measured output can be written in frequency domain

$$Y(\omega) = H(\omega) X(\omega) + N(\omega) \quad 3.6.1$$

where N is the noise spectrum. Equation 3.6.1 can be rewritten as

$$Y(\omega) = H(\omega) \left[X(\omega) + \frac{N(\omega)}{H(\omega)} \right] = H(\omega) \tilde{X}(\omega)$$

where $H(\omega)$ is real. The use of phase-only reconstruction yields a finite length approximation to $\tilde{x}(n)$ which might be written as

$$\tilde{x}(n) = x(n) + v(n) * g(n)$$

where $g(n)$ is the convolutional inverse of $h(n)$. The expected value is

$$E(\tilde{x}(n)) = E(x(n)) + E(v(n)) * g(n) = E(x(n)) = x(n)$$

and the variance of error in $\tilde{x}(n)$ is

$$\sigma_x^2 = E(|\tilde{x}(n) - x(n)|^2) \quad 3.6.2$$

$$= \sum_{k=-\infty}^{\infty} \sum_{r=-\infty}^{\infty} g(n-k) g(n-r) E(v(k)v(r))$$

$$= \sigma^2 \sum_{r=-\infty}^{\infty} g^2(n-r) = \sigma^2 \sum_{r=-\infty}^{\infty} g^2(r)$$

$$= \frac{\sigma^2}{2\pi} \int_{-\pi}^{\pi} |G(\omega)|^2 d\omega = \frac{\sigma^2}{2\pi} \int_{-\pi}^{\pi} \frac{1}{|H(\omega)|^2} d\omega$$

which follows from Parseval's Theorem. Thus, the variance in error due to noise is proportional to σ^2 .

3.7. Examples of Phase-Only Blind Deconvolution

In this chapter we have discussed the application of various techniques of Phase-Only signal reconstruction to Blind Deconvolution. We will now present an example of Phase-Only blind deconvolution. A synthetic signal $x(n)$ was passed through a filter with

impulse response $h(n)$ to yield $y(n)$. Figures 3.3 and 3.4 show signal $x(n)$ and $h(n)$ respectively, and figure 3.5 shows the output $y(n)$. The filter impulse response $h(n)$ has 41 samples and its center of symmetry is at $n=20$. Using the facts that $h(n)$ is symmetric and $y(n)$ is known, $x(n)$ was computed by solving the system of linear equations of equation 2.6.1 (algorithm 1a). Figure 3.6 shows the result of this reconstruction. The original and reconstructed signals are identical except for the scale factor. The signal $x(n)$ was also reconstructed using the least squared technique (algorithm 1b). The plot of this $x(n)$ is indistinguishable from the original signal.

Next we assume that the length of the filter $h(n)$ is not known and use the technique of section 3.5 to recompute $x(n)$. The result (figure 3.7) is again indistinguishable from the original. To investigate how Magnitude weighted Phase-Only Blind Deconvolution compares with Unit Magnitude Phase-Only Blind Deconvolution when the filter has unit circle zeros white Gaussian noise of varying noise power was added to the output signal $y(n)$. The signal $x(n)$ (figure 3.8) chosen for this purpose was a modulated Gaussian pulse with normalized frequency of 0.1 and bandwidth 0.02. The filter chosen was a 15 point linear phase equiripple filter with passband 0.0 to 0.25 and stopband 0.35 to 0.5. The filter had three zeros in the stopband. This allowed the bulk of the signal spectrum unchanged while adding zeros on the output spectrum. Noise of varying power was now added to the resulting output signal. The signal $x(n)$ was now reconstructed and the resulting SER was computed. This is plotted in figure 3.9. The

Figure 3.3: Signal $x[n]$

Figure 3.4: Linear Phase Filter $h(n)$

Figure 3.5: Output $y(n)=x(n)*h(n)$

Figure 3.6: Magnitude Weighted Reconstruction of $x(n)$

Figure 3.7: Reconstruction of $x(n)$ using Toeplitz Structure

Figure 3.8: Modulated Gaussian

PLEASE NOTE:

**Duplicate page numbers. Text
follows. Filmed as received.**

University Microfilms International

Figure 3.9: SER, v/s $\sqrt{\text{Noise Variance}}$

Magnitude Weighted reconstruction gives an SER which about 10 db. better thus proving its superior performance over Unit Magnitude reconstruction. Also, we note that the plots are substantially linear which indicates that the output SER is proportional to input SER and confirms the conclusion of equation 3.6.2.

3.8. Conclusions

In this chapter the application of Phase-Only signal reconstruction to Blind Deconvolution was described. The previous work by Hayes et.al. [33] involved the solution of equation 2.4.5 in an iterative fashion. We have introduced the time domain versions of the problem and in particular we have developed a new Magnitude Weighted signal reconstruction technique which avoids actual computation of phase. These time domain versions lead to systems of equations which can be solved by a very efficient algorithm to be presented in the next chapter.

We have also presented some examples of blind deconvolution and compared the Magnitude weighted Phase-Only Blind Deconvolution with Unit Magnitude Phase-Only Blind Deconvolution. The results show that the Magnitude weighted version performs better in presence of additive noise. In addition we have presented a novel version of the problem which requires solution of a conventional Toeplitz system of equations using the Levinson recursion and which does not require the knowledge of the length of the filter.

CHAPTER 4

EFFICIENT ALGORITHM TO SOLVE TOEPLITZ-PLUS-HANKEL SYSTEM OF EQUATIONS

4.1. Introduction

In chapters 2 and 3, the problem of Phase-only reconstruction has been converted into a problem of solving a system of linear equations. The coefficient matrix of this new system of equations is expressible as a sum of a Toeplitz matrix and a Hankel matrix. Similar systems of equations arise in other problems as well [44-46]. In this chapter we present a technique which converts above problem into an equivalent block-Toeplitz matrix inversion problem, which can be solved using block-Levinson recursion [43]. This extends the savings obtained by Levinson recursion [4] for inversion of Toeplitz matrix to the problems involving the inversion of matrices which are the sum of Toeplitz and Hankel matrices.

Further reduction in computation can be obtained by using the symmetries inherent in a Toeplitz plus Hankel matrix. The resulting algorithm has a storage requirement proportional to n , the number of unknowns and the number of multiplications to the square of n . To facilitate subsequent discussion we begin with definition of some terms and the notations.

4.2. Notation

Let A be any matrix and p be a vector and the matrix

$$J = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

is an n by n operator which performs a "flip" or a reversal operation.

A^T = transpose of matrix A ($a_{ij} \Leftrightarrow a_{ji}$)

A^π = cross transpose of matrix A around the main cross diagonal ($a_{ij} \Leftrightarrow a_{n-i-j, n-1-i}$). If $A^\pi = A$, then A is said to be persymmetric.

p_+ = vector p .

$p_- = J p$ = vector p reordered in reverse index sequence.

T = Toeplitz matrix with $\{T\}_{ij} = t(i-j)$.

H = Toeplitz matrix with $\{H\}_{ij} = h(i+j-n+1)$.

Note that $J^2 = I$ = identity and $J A J = A^{T\pi} = A^{\pi T}$. The operations of $(.)^T$ and $(.)^\pi$ commute. Further, if B is any other matrix then

$$(A B)^{T\pi} = J A B J = J A J \cdot J B J = A^{T\pi} B^{T\pi}$$

T is always persymmetric and H is always symmetric. Also we define a $2n$ by $2n$ "interleaving" operator Q such that

$$\{Q\}_{ij} = \begin{cases} 1 & \text{for } i = 2r, j = r, r=0,1,\dots,n-1 \\ 1 & \text{for } i = 2r+1, j = n+r, r=0,1,\dots,n-1 \\ 0 & \text{for all other } (i,j) \text{ pairs} \end{cases}$$

It is simple to show that $Q^T Q = Q Q^T = I$. If we operate on a vector p with matrix Q then Q simply interleaves the sequences

$$p_r \text{ and } p_{n+r}, r = 0,1,\dots,n-1.$$

In matrix notation this is

$$Q [p_0 p_1 \dots p_n p_{n+1} \dots p_{2n-1}]^T = [p_0 p_n p_1 p_{n+1} \dots p_{n-1} p_{2n-1}]^T.$$

4.3. Conversion of a Toeplitz-Plus-Hankel Operator to a Block-Toeplitz Operator

Consider a problem

$$(T + H) p = b \quad 4.3.1$$

where T and H are n by n Toeplitz and Hankel matrices, respectively, p is the vector to be determined and b is a known vector. Rewrite (4.3.1) as

$$T p + H p = b. \quad 4.3.2$$

Using the notation in section 4.2 equation 4.3.2 can be written in two different forms:

$$T p + H J \cdot J p = b \quad 4.3.3a$$

$$J T J \cdot J p + J H p = J b. \quad 4.3.3b$$

Since T is persymmetric ($T^\pi = T$) it follows that and equation 4.3.3a and b become

$$T p_+ + (HJ) p_- = b_+ \quad 4.3.4a$$

$$T^\tau p_- + (JH) p_+ = b_- \quad 4.3.4b$$

Equations 4.3.4a and b can now be written in a block form

$$\begin{bmatrix} T & HJ \\ JH & T^\tau \end{bmatrix} \begin{bmatrix} p_+ \\ p_- \end{bmatrix} = \begin{bmatrix} b_+ \\ b_- \end{bmatrix}. \quad 4.3.5$$

Since H is a Hankel matrix $H^\tau = H$ and $(HJ)^\tau = J^\tau H^\tau = JH$. Denoting HJ by T_H , 4.3.5 becomes

$$\begin{bmatrix} T & T_H \\ T_H^\tau & T^\tau \end{bmatrix} \begin{bmatrix} p_+ \\ p_- \end{bmatrix} = \begin{bmatrix} b_+ \\ b_- \end{bmatrix}. \quad 4.3.6$$

We now make a crucial observation, that T_H is a Toeplitz matrix. Thus, each block matrix in equation 4.3.6 is a Toeplitz matrix. The coefficient matrix in equation 4.3.6 is a $2n$ by $2n$ matrix. The known and unknown vectors are of length $2n$. Using the interleaving operator Q on equation 4.3.6

$$Q \cdot \begin{bmatrix} T & T_H \\ T_H^\tau & T^\tau \end{bmatrix} \cdot Q^\tau \cdot Q \cdot \begin{bmatrix} p_+ \\ p_- \end{bmatrix} = Q \cdot \begin{bmatrix} b_+ \\ b_- \end{bmatrix}. \quad 4.3.7$$

If we now define a set of 2 by 2 matrices

$$R(i, j) = \begin{bmatrix} t(i, j) & h(i, j) \\ h(j, i) & t(j, i) \end{bmatrix} \quad 4.3.8$$

where $t(i, j) = \{T\}_{ij}$ and $h(i, j) = \{T_H\}_{ij}$, it can be seen that .

equation 4.3.7 gives a new coefficient matrix, in which the elements are 2 by 2 matrices similar to those defined in equation 4.3.8. Since T and T_H are Toeplitz matrices

$$t(i,j) = t_{i-j} \quad \text{and} \quad h(i,j) = h_{i-j}.$$

Hence,

$$R(i,j) = \begin{bmatrix} t_{i-j} & h_{i-j} \\ h_{j-i} & t_{j-i} \end{bmatrix} = R_{i-j}.$$

Using the above notation for $R(i,j)$, 4.3.7 can be written as

$$\begin{bmatrix} R_0 & R_{-1} & R_{-2} & \cdot & \cdot & \cdot & R_{-n+1} \\ R_1 & R_0 & R_{-1} & \cdot & \cdot & \cdot & R_{-n+2} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ R_{n-1} & \cdot & \cdot & \cdot & \cdot & \cdot & R_0 \end{bmatrix} \begin{bmatrix} \bar{p}_0 \\ \bar{p}_1 \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \bar{p}_{n-1} \end{bmatrix} = \begin{bmatrix} \bar{b}_0 \\ \bar{b}_1 \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \bar{b}_{n-1} \end{bmatrix} \quad 4.3.9$$

where

$$\bar{p}_i = \begin{bmatrix} p_i \\ p_{n-1-i} \end{bmatrix} \quad \text{and} \quad \bar{b}_i = \begin{bmatrix} b_i \\ b_{n-1-i} \end{bmatrix}.$$

Equation 4.3.9 has a block-Toeplitz coefficient matrix and can be solved by a block-Levinson algorithm as shown by Akaike[43]

4.4. Block-Levinson Algorithm

In this section a regular block-Levinson algorithm to solve equation 4.3.9 is presented. All the lower case letters represent

length-2 vectors, and all the capital letters represent 2 by 2 matrices.

$$1) \mathbf{x}_0 = \mathbf{y}_0 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \mathbf{p}_0 = \mathbf{p}_1 = \mathbf{R}_0^{-1} \bar{\mathbf{b}}_0, \quad \mathbf{V}_x = \mathbf{V}_y = \mathbf{R}_0$$

2) For $i=1, 2, \dots, (n-1)$; do

$$1) \mathbf{E}_x = \sum_{j=0}^{i-1} \mathbf{R}_{i-j} \mathbf{X}_j, \quad \mathbf{E}_y = \sum_{j=1}^i \mathbf{R}_{-j} \mathbf{Y}_{i-j}$$

$$2) \bar{\mathbf{e}}_p = \sum_{j=0}^{i-1} \mathbf{R}_{i-j} \bar{\mathbf{p}}_j$$

$$3) \mathbf{B}_x = \mathbf{V}_y^{-1} \mathbf{E}_x, \quad \mathbf{B}_y = \mathbf{V}_x^{-1} \mathbf{E}_y$$

$$4) \begin{bmatrix} \mathbf{I} \\ \mathbf{X}_1 \\ \mathbf{X}_2 \\ \cdot \\ \cdot \\ \cdot \\ \mathbf{X}_i \end{bmatrix} \leftarrow \begin{bmatrix} \mathbf{I} \\ \mathbf{X}_1 \\ \mathbf{X}_2 \\ \cdot \\ \cdot \\ \mathbf{X}_{i-1} \\ 0 \end{bmatrix} - \begin{bmatrix} 0 \\ \mathbf{Y}_{i-1} \\ \cdot \\ \cdot \\ \cdot \\ \mathbf{Y}_1 \\ \mathbf{I} \end{bmatrix} \cdot \mathbf{B}_x; \quad \begin{bmatrix} \mathbf{Y}_i \\ \mathbf{Y}_{i-1} \\ \cdot \\ \cdot \\ \cdot \\ \mathbf{Y}_1 \\ \mathbf{I} \end{bmatrix} \leftarrow \begin{bmatrix} 0 \\ \mathbf{Y}_{i-1} \\ \cdot \\ \cdot \\ \cdot \\ \mathbf{Y}_1 \\ \mathbf{I} \end{bmatrix} - \begin{bmatrix} \mathbf{I} \\ \mathbf{X}_1 \\ \mathbf{X}_2 \\ \cdot \\ \cdot \\ \mathbf{X}_{i-1} \\ 0 \end{bmatrix} \cdot \mathbf{B}_y$$

$$\mathbf{V}_x \leftarrow \mathbf{V}_x - \mathbf{E}_x \mathbf{B}_x; \quad \mathbf{V}_y \leftarrow \mathbf{V}_y - \mathbf{E}_y \mathbf{B}_y$$

$$5) \bar{\mathbf{g}} = \mathbf{V}_y^{-1} (\bar{\mathbf{b}}_i - \bar{\mathbf{e}}_p)$$

$$6) \mathbf{P}_{i+1} = \begin{bmatrix} \bar{p}_0 \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \bar{p}_{i-1} \\ \bar{p}_i \end{bmatrix} \leftarrow \begin{bmatrix} \bar{p}_0 \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ \bar{p}_{i-1} \\ 0 \end{bmatrix} + \begin{bmatrix} y_i \\ y_{i-1} \\ \cdot \\ \cdot \\ \cdot \\ y_1 \\ \mathbf{I} \end{bmatrix} \cdot \bar{g}$$

7) next i

3) rearrange $\mathbf{P} = (p_0 \ p_1 \ p_2 \ \dots \ p_{2n-1})^T$

$$p_r \leftarrow p_{2r}, \quad r = 0, 1, \dots, (n-1).$$

At the completion of step 3, the first half of the rearranged vector $\mathbf{P}_n$ gives the solution to equation 4.3.1. In the algorithm, it is assumed that there is a unique solution to equation 4.3.1, i.e. $(T+H)$ is nonsingular. Also, in steps (2.3) and (2.5) it is assumed that the necessary inverses of V_x and V_y exist. This is true if the corresponding block-submatrices are nonsingular. This can be shown as follows:

In step 2, at the k -th substep, let the auxiliary vectors be denoted as

$$\{ \mathbf{I}, (y_1^k)^T, (y_2^k)^T, \dots, (y_k^k)^T \}^T$$

and

$$\{ \mathbf{I}, (X_1^k)^\tau, (X_2^k)^\tau, \dots, (X_k^k)^\tau \}^\tau$$

and matrices V_x and V_y as V_x^k and V_y^k . Then, at the end of the k -th substep the following relationships hold.

$$\begin{bmatrix} \mathbf{I} & & & & & \\ X_1^1 & \mathbf{I} & & & & \\ X_2^2 & X_1^2 & \mathbf{I} & & & 0 \\ X_3^3 & X_2^3 & X_1^3 & \mathbf{I} & & \\ \cdot & \cdot & \cdot & \cdot & \mathbf{I} & \\ \cdot & \cdot & \cdot & \cdot & \cdot & \mathbf{I} \\ X_k^k & X_{k-1}^k & X_{k-2}^k & \cdot & \cdot & \cdot & \mathbf{I} \end{bmatrix} \begin{bmatrix} R_0 & R_{-1} & \cdot & R_{-k} \\ R_1 & R_0 & \cdot & R_{-k+1} \\ R_2 & R_1 & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \\ R_k & R_{k-1} & \cdot & R_0 \end{bmatrix}$$

$$= \begin{bmatrix} V_x^0 & S_1^0 & S_2^0 & \cdot & S_k^0 \\ & V_x^1 & S_2^1 & \cdot & S_k^1 \\ & & V_x^2 & \cdot & \cdot \\ & & & \cdot & \cdot \\ & & & & \cdot \\ & & 0 & & \cdot \\ & & & & V_k^k \end{bmatrix} \quad 4.4.1$$

where $S_j^i = \sum_{r=0}^i X_r^i R_{(i-j)-r}$. For our purpose, the S matrices do not play any significant role. Denote the matrix

$$\begin{bmatrix} R_0 & R_{-1} & \cdot & \cdot & R_{-k} \\ R_1 & R_0 & \cdot & \cdot & R_{-k+1} \\ R_2 & R_1 & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ R_k & R_{k-1} & \cdot & \cdot & R_0 \end{bmatrix} = \mathbf{R}_k$$

Then, from equation 4.4.1

$$\det(\mathbf{R}_k) = \prod_{i=0}^k \det(V_x^i) \quad 4.4.3a$$

and similarly it can be shown that

$$\det(\mathbf{R}_k) = \prod_{i=0}^k \det(V_y^i) \quad 4.4.3b$$

However, it can be shown that [47]

$$\det(\mathbf{R}_k) = \det(T_k + H_k) \det(T_k - H_k) \quad 4.4.4$$

where T_k is the leading block-submatrix along the main diagonal and H_k is the leading block-submatrix along the main cross diagonal. Clearly, if $\det(V_x^k) = 0$ then $\det(T_k + H_k) = 0$ or $\det(T_k - H_k) = 0$ or both are zero. Assuming that $\det(T_k + H_k) \neq 0$ for all k the invertibility of $\mathbf{R}_k$ depends on whether $\det(T_k - H_k)$ is zero or not. We shall now prescribe an approach which in general should make the invertibility of $\mathbf{R}_k$ independent of $\det(T_k - H_k)$.

Let D_n be an n by n matrix such that it is both Toeplitz and Hankel. Then equation 4.3.1 can be written as

$$(T' + H') p = b$$

4.4.5

where $T' = T + D_n$ and $H' = H - D_n$ are Toeplitz and Hankel matrices, respectively. Clearly, we are still solving the same problem. One simple way of generating a matrix D_n which is both Toeplitz and Hankel is to generate a Toeplitz matrix with the vector

$$\{d_{-n+1}, d_{-n+2}, \dots, d_{-1}, d_0, d_1, \dots, d_{n-1}\}^T$$

where

$$d_{2r} = d_{-2r} = d_0, d_{2r-1} = d_{-2r+1} = d_1, r=1, 2, \dots, \text{integer}[\frac{n}{2}]$$

and where $\{D_n\} = d_{i-j}$. Here d_0 and d_1 can be any two numbers. At the k -th substep in step 2 of the algorithm, let U_k be the leading $(k+1)$ by $(k+1)$ submatrix along the main diagonal of D_n and W_k be the leading $(k+1)$ by $(k+1)$ submatrix along the main cross diagonal of D_n as shown below

$$D_n = \begin{bmatrix} d_0 & d_1 & d_0 & d_1 & \cdot & \cdot \\ d_1 & d_0 & d_1 & d_0 & \cdot & \cdot \\ d_0 & d_1 & d_0 & d_1 & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \end{bmatrix} = \begin{bmatrix} U_k & \cdot \\ \cdot & \cdot \end{bmatrix} = \begin{bmatrix} \cdot & V_k \\ \cdot & \cdot \end{bmatrix}.$$

Both U_k and V_k have the same property as D_n , viz. each of them is both Toeplitz and Hankel. If $|d_0| \neq |d_1|$ and if the number of unknowns n is odd, then for $k=0, 2, 4, \dots$

$$U_k = W_k, \quad U_{k+1} \neq W_{k+1}$$

If $|d_0| \neq |d_1|$ and if n is even, then for $k=0, 2, 4, \dots$

$$U_k \neq W_k, \quad U_{k+1} = W_{k+1}$$

If $d_0 = d_1$ then for $k=0, 1, 2, \dots$

$$U_k = W_k$$

If $d_0 = -d_1$ then for $k=0, 1, 2, \dots$

$$U_k = \pm W_k$$

the sign being dependent on the values of n and k . Equation 4.4.4 now becomes

$$\det(\mathbb{R}'_k) = \det(T'_k + H'_k) \det(T'_k - H'_k)$$

where $T'_k = T_k + U_k$ and $H'_k = H_k - W_k$. Even if $T_k - H_k$ is singular, addition of an arbitrary matrix $U_k + W_k$ will in general make it nonsingular. For the same reason, $T_k + H_k$ which has been assumed to be nonsingular, will remain so. This in turn implies that in general $\det(\mathbb{R}'_k) \neq 0$ which ensures that in equations 4.4.3a and b

$$\prod_{i=0}^k \det(V'_x{}^i) = \prod_{i=0}^k \det(V'_y{}^i) \neq 0$$

Hence, $V'_x{}^k$ and $V'_y{}^k$ are invertible.

If during execution of the algorithm, we encounter V_x^k and V_y^k with determinant zero, we need to restart the algorithm from the beginning with modified T and H . A convenient way of avoiding this is to start with modified T and H and solve the problem, regardless of whether the unmodified problem is known to be solvable or not. One

simple choice of matrix D_n is to use $d_0 = t_0$ and $d_1 = 0$. This modified procedure has been implemented with excellent results.

4.5. Use of Symmetries to Reduce Computation

Write 4.4.9 as

$$\mathbf{R}_n \mathbf{P} = \mathbf{B} \quad 4.5.1$$

where $\mathbf{R}_n$ is the block-Toeplitz coefficient matrix. We show that $\mathbf{R}_n^{\tau\pi} = \mathbf{R}_n$. Since $\mathbf{R}_n^{\tau\pi} = \mathbf{J}_{2n} \mathbf{R}_n \mathbf{J}_{2n}$, the (i,j) -th block element of $\mathbf{R}_n^{\tau\pi}$ is

$$\{\mathbf{R}_n^{\tau\pi}\}_{ij} = \mathbf{J}_2 \mathbf{R}_{(n-1-i, n-1-j)} \mathbf{J}_2 = \mathbf{J}_2 \mathbf{R}_{j-i} \mathbf{J}_2 = \mathbf{R}_{j-i}^{\tau\pi} \quad 4.5.2$$

But,

$$\mathbf{R}_{j-i} = \begin{bmatrix} t_{j-i} & h_{j-i} \\ h_{i-j} & t_{i-j} \end{bmatrix}$$

hence

$$\mathbf{R}_{j-i}^{\tau\pi} = \begin{bmatrix} t_{i-j} & h_{i-j} \\ h_{j-i} & t_{j-i} \end{bmatrix} = \mathbf{R}_{i-j}$$

Thus, $\{\mathbf{R}_n^{\tau\pi}\}_{j-i} = \{\mathbf{R}_n\}_{i-j}$. Also, note that $\mathbf{J}_{2n} \mathbf{P} = \mathbf{P}$ and $\mathbf{J}_{2n} \mathbf{B} = \mathbf{B}$.

Using the above symmetries, it can be shown in the algorithm of section 4.4 that the following relations hold (see Appendix C):

$$a) \quad \mathbf{E}_y = \mathbf{E}_x^{\tau\pi}$$

$$b) \quad \mathbf{B}_y = \mathbf{B}_x^{\tau\pi}$$

$$c) \quad Y_r = X_r^{\tau\pi}, \quad r=0,1,\dots,i$$

$$d) \quad V_y = V_x^{\tau\pi}$$

These relations lead to a simplified algorithm, requiring less computation and storage. The new algorithm is:

$$1) \quad x_0 = y_0 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad p_0 = \underline{p}_1 = R_0^{-1} \bar{b}_0, \quad V_x = R_0$$

2) For $i=1,2, \dots, (n-1)$; do

$$1) \quad E_x = \sum_{j=0}^{i-1} R_{i-j} X_j$$

$$2) \quad \bar{e}_p = \sum_{j=0}^{i-1} R_{i-j} \bar{p}_j$$

$$3) \quad B_x = \{V_x^{\tau\pi}\}^{-1} E_x$$

$$4) \quad \begin{bmatrix} I \\ X_1 \\ X_2 \\ \cdot \\ \cdot \\ \cdot \\ X_i \end{bmatrix} \longleftarrow \begin{bmatrix} I \\ X_1 \\ X_2 \\ \cdot \\ \cdot \\ X_{i-1} \\ 0 \end{bmatrix} - \begin{bmatrix} 0 \\ X_{i-1}^{\tau\pi} \\ \cdot \\ \cdot \\ \cdot \\ X_1^{\tau\pi} \\ I \end{bmatrix} \cdot B_x$$

$$V_x \longleftarrow V_x - E_x^{\tau\pi} B_x,$$

$$5) \quad \bar{g} = \{V_x^{\tau\pi}\}^{-1} (\bar{b}_i - \bar{e}_p)$$

$$6) \mathbf{P}_{i+1} = \begin{bmatrix} \bar{p}_0 \\ . \\ . \\ . \\ . \\ \bar{p}_{i-1} \\ \bar{p}_i \end{bmatrix} \leftarrow \begin{bmatrix} \bar{p}_0 \\ . \\ . \\ . \\ . \\ \bar{p}_{i-1} \\ 0 \end{bmatrix} + \begin{bmatrix} X_i^{\tau\pi} \\ X_{i-1}^{\tau\pi} \\ . \\ . \\ . \\ X_1^{\tau\pi} \\ \mathbf{I} \end{bmatrix} \cdot \bar{g}$$

7) next i

3) rearrange $\mathbf{P} = (p_0 \ p_1 \ p_2 \ \dots \ p_{2n-1})^T$

$p_r \leftarrow p_{2r}, \ r = 0, 1, \dots, (n-1).$

The remarks following the algorithm in section 4.4 hold here too.

4.6. General Discussion

The coefficient matrix in 4.3.5 belongs to a more general class of matrices called the centrosymmetric matrices. Given two matrices A and B we can form a centrosymmetric matrix as

$$\begin{bmatrix} A & BJ \\ JB & JAJ \end{bmatrix}$$

Good [47] has shown that finding the inverse of the above matrix is equivalent to finding inverses of two smaller matrices, viz. (A+B) and (A-B). In this chapter we have taken the opposite view and created a larger centrosymmetric matrix which, because of the special

properties of T and H , can be inverted more efficiently than inverting $(T+H)$ directly.

4.7. Conclusion

The method described in section 4. has been used to solve a system of equations with 255 unknowns on a PDP 11/55 in double precision. The executive time required was of the order of one minute. For purposes of comparison, a simplified version of the same problem (where matrix T turns out to be identity) was solved by the Conjugate Gradient (CG) method [48] for symmetric matrices. The CG method required six major iterations to solve the problem with comparable accuracy and the execution time was of the order of four hours. The CG method was chosen for comparison since, like the method described in this chapter, it does not require explicit storage of the coefficient matrix. Moreover the total array size required is comparable in each case. The results are not surprising considering that the CG method has the computational complexity of $O(n^3)$.

CHAPTER 5

APPLICATION OF PHASE-ONLY RECONSTRUCTION TO BLIND DECONVOLUTION OF SEGMENTED DATA

5.1. Introduction

Chapters 2 and 3 have discussed the basis for Phase-Only Blind Deconvolution. It is clear from equation 2.4.4 or equation 3.3.2 that it is necessary to use the entire length of output signal $y(\cdot)$ to be able to use Phase-Only reconstruction for Blind Deconvolution.

When the sequence $y(\cdot)$ is too long, it may be difficult to reconstruct $x(\cdot)$ all at once. Clearly, it would help to be able to reconstruct $x(\cdot)$ from the known segments of $y(\cdot)$ [49].

5.2. Phase-Only Blind Deconvolution: Two Segment Case

For purposes of discussion we consider the case when we can split signal $y(n)$ into two distinct segments. Let $x(n)$, $n=0, 1, \dots, M$ be a long input sequence with no symmetric factor. Split $x(n)$ into two segments $\tilde{x}_1(n)$ and $\tilde{x}_2(n)$ such that

for $n = 0, 1, \dots, M_1$

$$\tilde{x}_1(n) = x(n) \text{ and } \tilde{x}_2(n) = 0$$

and for $M_1 < n \leq M$

$$\tilde{x}_1(n) = 0 \text{ and } \tilde{x}_2(n) = x(n)$$

Then

$$x(n) = \tilde{x}_1(n) + \tilde{x}_2(n)$$

for all n . The corresponding results of convolution with a symmetric odd length sequence $h(n)$ (i.e. zero phase) is

$$\begin{aligned} y(n) &= \sum_r h(n-r) x(r) \\ &= \sum_r h(n-r) (\tilde{x}_1(r) + \tilde{x}_2(r)) \\ &= \sum_r h(n-r) \tilde{x}_1(r) + \sum_r h(n-r) \tilde{x}_2(r) \\ &= \tilde{y}_1(n) + \tilde{y}_2(n) \end{aligned}$$

where $\tilde{y}_1(n)$ and $\tilde{y}_2(n)$ represent the first and second summations respectively on the right hand side. Figure 5.1a shows $\tilde{x}_1(n)$ and $\tilde{x}_2(n)$ and Figure 5.1b shows $\tilde{y}_1(n)$, $\tilde{y}_2(n)$ and $y(n)$. Sequences $\tilde{y}_1(n)$ and $\tilde{y}_2(n)$ overlap in the interval $[M_1-L, M_1+L]$. Since only $y(n)$, the sum of $\tilde{y}_1(n)$ and $\tilde{y}_2(n)$, is available, segmenting it at $n = M_1$ will not give $\tilde{y}_1(n)$ and $\tilde{y}_2(n)$. Let $y(n)$ be segmented into two parts such that for $-L \leq n \leq M_1$

$$y_1(n) = y(n) \text{ and } y_2(n) = 0$$

and for $M_1 < n \leq M + L$

$$y_1(n) = 0 \text{ and } y_2(n) = y(n).$$

Figure 5.1(a): Input Segment $\tilde{x}_1(n)$ and $\tilde{x}_2(n)$

(b): Output Segment $\tilde{y}_1(n)$ and $\tilde{y}_2(n)$

It follows that

$$y(n) = y_1(n) + y_2(n) = \tilde{y}_1(n) + \tilde{y}_2(n).$$

$$\text{Let} \quad y_1(n) = \tilde{y}_1(n) + e(n) \quad 5.2.1a$$

$$\text{then} \quad y_2(n) = \tilde{y}_2(n) - e(n) \quad 5.2.1b$$

$$\text{where} \quad e(n) = y_1(n) - \tilde{y}_1(n)$$

$$= y(n) - \tilde{y}_1(n) = \tilde{y}_2(n) \quad M_1 - L \leq n \leq M_1$$

$$= 0 - \tilde{y}_1(n) = -\tilde{y}_1(n) \quad M_1 < n \leq M_1 + L$$

Using the definitions of $\tilde{y}_1(n)$ and $\tilde{y}_2(n)$

$$y_1(n) = \sum_r h(n-r) \tilde{x}_1(r) + e(r)$$

$$y_2(n) = \sum_r h(n-r) \tilde{x}_2(r) - e(r).$$

In the frequency domain, the above equations become

$$Y_1(\omega) = H(\omega)\tilde{X}_1(\omega) + E(\omega)$$

$$Y_2(\omega) = H(\omega)\tilde{X}_2(\omega) - E(\omega)$$

which can be rewritten as

$$Y_1(\omega) = H(\omega) \left[\tilde{X}_1(\omega) + \frac{E(\omega)}{H(\omega)} \right] = H(\omega)X_1(\omega) \quad 5.2.2a$$

$$Y_2(\omega) = H(\omega) \left[\tilde{X}_2(\omega) - \frac{E(\omega)}{H(\omega)} \right] = H(\omega)X_2(\omega) \quad 5.2.2b$$

where

$$X_1(\omega) = \tilde{X}_1(\omega) + \frac{E(\omega)}{H(\omega)}$$

$$X_2(\omega) = \tilde{X}_2(\omega) - \frac{E(\omega)}{H(\omega)}$$

Equations 5.2.2a and b are of the same form as equation 3.2.2. We assume that a finite length approximation of $X_1(\omega)$ and $X_2(\omega)$ satisfy the requirements of Theorem 1 in Chapter 2. Then we can use Phase-Only reconstruction to compute $X_1(\omega)$ and $X_2(\omega)$. In the time domain

$$x_1(n) = \tilde{x}_1(n) + e(n) * g(n) \quad 5.2.3a$$

$$x_2(n) = \tilde{x}_2(n) - e(n) * g(n) \quad 5.2.3b$$

where $g(n)$ is the noncasual convolutional inverse of $h(n)$ and "*" denotes convolution.

According to the definitions of $X_1(\omega)$ and $X_2(\omega)$ in equations 5.2.2a and b, $x_1(n)$ and $x_2(n)$ must be infinite length sequences (due to the presence of $\frac{E(\omega)}{H(\omega)}$). The theorems in Chapter 2 for Phase-Only reconstruction require that the sequences to be calculated be of finite length. Thus, it is necessary to know what kinds of errors are made when the finite length versions of $x_1(n)$ and $x_2(n)$ are computed. Let $G(\omega)$ be the Discrete Fourier Transform of $g(n)$, the finite length noncasual stable convolutional inverse of $h(n)$. Since $H(\omega)$ is real, we assume $G(\omega)$ is real, too. Then it is always possible to find a $g(n)$ such that

$$|1 - H(\omega)G(\omega)| < \varepsilon \quad \text{for all } |\omega| < \pi$$

for any given $\varepsilon < 0$. Let $S_1(\omega)$ denote an approximation of $X_1(\omega)$. Let " $_$ " denote complex conjugation.

Lemma 5.1: Given $\varepsilon > 0$ however small, there exists a $\delta > 0$ such that

$$|X_1(\omega) - S_1(\omega)| < \delta \quad \text{for all } \omega$$

if and only if

$$|\operatorname{Im}(Y_1(\omega)\underline{S}_1(\omega))| < \varepsilon$$

Proof:

Consider the forward implication. Since $X_1(\omega)$ satisfies

$$Y_1(\omega) \underline{X}_1(\omega) = \underline{Y}_1(\omega) X_1(\omega).$$

We have

$$\begin{aligned} \operatorname{Im}(Y_1(\omega) \underline{S}_1(\omega)) &= \frac{1}{2j} \{Y_1(\omega)\underline{S}_1(\omega) - \underline{Y}_1(\omega)S_1(\omega)\} \\ &= \frac{1}{2j} \{Y_1(\omega)[\underline{S}_1(\omega) - \underline{X}_1(\omega)] - \underline{Y}_1(\omega)[S_1(\omega) - X_1(\omega)]\} \end{aligned}$$

Hence

$$|\operatorname{Im}(Y_1(\omega) \underline{S}_1(\omega))| \leq \frac{1}{2} \cdot 2|Y_1(\omega)| |S_1(\omega) - X_1(\omega)| \leq \varepsilon$$

where $\varepsilon = \max_{\omega} |Y_1(\omega)| \delta$. This proves the forward implication. Now consider the reverse implication.

$$|\operatorname{Im}(Y_1(\omega) \underline{S}_1(\omega))| < \varepsilon \implies |Y_1(\omega)\underline{S}_1(\omega) - \underline{Y}_1(\omega)S_1(\omega)| < 2\varepsilon \quad 5.2.4$$

Let $S_1(\omega)$ be of form

$$S_1(\omega) = \tilde{X}_1(\omega) + E(\omega)G(\omega)$$

Using $Y_1(\omega) = H(\omega)\tilde{X}_1(\omega) + E(\omega)$

$$\begin{aligned} |Y_1(\omega)\underline{S}_1(\omega) - \underline{Y}_1(\omega)S_1(\omega)| &= |(H(\omega)\tilde{X}_1(\omega) + E(\omega))(\tilde{X}_1(\omega) + \underline{E}(\omega)G(\omega)) \\ &\quad - (H(\omega)\tilde{X}_1(\omega) + \underline{E}(\omega))(\tilde{X}_1(\omega) + E(\omega)G(\omega))| \\ &= |E(\omega)\tilde{X}_1(\omega) - \underline{E}(\omega)\tilde{X}_1(\omega)| |1 - H(\omega)G(\omega)| < 2\varepsilon. \end{aligned}$$

Let the length of $G(\omega)$ be such that ε_1 satisfies

$$\begin{aligned} 2\varepsilon &= \max_{\omega} |E(\omega)\tilde{X}_1(\omega) - \underline{E}(\omega)\tilde{X}_1(\omega)| \max_{\omega} |1 - H(\omega)G(\omega)| \\ &= \max_{\omega} |E(\omega)\tilde{X}_1(\omega) - \underline{E}(\omega)\tilde{X}_1(\omega)| \varepsilon_1 \end{aligned}$$

Now

$$\begin{aligned} |X_1(\omega) - S(\omega)| &= \left| \frac{E(\omega)}{H(\omega)} - E(\omega)G(\omega) \right| \\ &= \left| \frac{E(\omega)}{H(\omega)} \right| |1 - H(\omega)G(\omega)| \\ &\leq \frac{\max_{\omega} |E(\omega)|}{\min_{\omega} |H(\omega)|} \varepsilon_1 = \delta \end{aligned}$$

where

$$\delta = \frac{2\varepsilon \max_{\omega} |E(\omega)|}{\min_{\omega} |H(\omega)| \max_{\omega} |E(\omega)\tilde{X}_1(\omega) - \underline{E}(\omega)\tilde{X}_1(\omega)|}$$

Next we note that $\max_{\omega} |H(\omega)| \neq 0$ since $H(\omega)$ has no zeros on the unit

circle. Further

$$\max_{\omega} |E(\omega) \tilde{X}_1(\omega) - \underline{E}(\omega) \tilde{X}_1(\omega)| \neq 0$$

since it would imply that the phase of $\tilde{X}_1(\omega)$ is equal to the phase of $E(\omega)$ for all ω and hence $E(\omega) = K \tilde{X}_1(\omega)$ where K is a constant. This is not possible, since it would require that

$$Y_1(\omega) = H(\omega) \tilde{X}_1(\omega) + E(\omega) = (H(\omega) + K) \tilde{X}_1(\omega)$$

be divisible by a symmetric factor, which is contrary to assumption.

QED

The preceding lemma implies that by choosing a good approximation of $\frac{1}{H(\omega)}$ we would be able to achieve as close an approximation to $X_1(\omega)$ as desired. One way of improving the approximation is to choose a long length sequence $g(n)$. Computing a long length approximation to $x_1(n)$ can achieve this. It can be shown (Lemma 5.2) that $g(n)$ is bounded by an exponentially decaying function of index n .

Lemma 5.2: Let z_r belong to the set of zeros of $H(z)$ and let

$$\mu = \max_{r, |z_r| < 1} |z_r|. \text{ Assume } g(n) \text{ to be a noncasual, stable convolutional inverse of } h(n). \text{ Then}$$

$$|g(n)| \leq K \mu^{|n|}$$

for all n and where K is a constant. The proof for the above lemma is presented in Appendix D.

Using the above lemma, we will show that the error term $e(n) * g(n)$ is also bounded by an exponentially decaying function of index n . We can now derive the bound on the error term in equations

5.2.3a and b.

$$\begin{aligned}
 \varepsilon(n) &= |e(n) * g(n)| = \left| \sum_{r=M_1-L}^{M_1+L} e(r) g(n-r) \right| \\
 &= \left| \sum_{r=-L}^L e(r+M_1) g(n-r-M_1) \right| \\
 &\leq \sum_{r=-L}^L |e(r+M_1)| K \mu^{|n-r-M_1|}
 \end{aligned}$$

For $n > M_1+L$

$$\begin{aligned}
 \varepsilon(n) &\leq \sum_{r=-L}^L |e(r+M_1)| K \mu^{n-r-M_1} \\
 &= \mu^{n-M_1} K \sum_{r=-L}^L |e(r+M_1)| \mu^{-r} \\
 &= K_1 \mu^{n-M_1}
 \end{aligned}$$

where $K_1 = K \sum_{r=-L}^L |e(r+M_1)| \mu^{-r}$. Similarly, for $n < M_1-L$

$$\varepsilon(n) \leq K_2 \mu^{M_1-n}$$

where $K_2 = K \sum_{r=-L}^L |e(r+M_1)| \mu^r$. Finally for $M_1-L \leq n \leq M_1+L$

$$\varepsilon(n) \leq K \mu^{n-M_1} \sum_{r=-L}^{n-1-M_1} |e(r+M_1)| \mu^{-r} + K \mu^{M_1-n} \sum_{r=n-M_1}^L |e(r+M_1)| \mu^r$$

$$\begin{aligned}
& < K \mu^{n-M_1} \sum_{r=-L}^L |e(r+M_1)| \mu^{-r} + K \mu^{M_1-n} \sum_{r=-L}^L |e(r+M_1)| \mu^r \\
& = K_1 \mu^{n-M_1} + K_2 \mu^{M_1-n} \\
& \leq K_1 \mu^{-L} + K_2 \mu^{-L} = (K_1 + K_2) \mu^{-L}
\end{aligned}$$

Thus, the error term is bounded in the middle (for $|n - M_1| \leq L$) and decays exponentially outside this interval (for $|n - M_1|$ large). Thus,

$$x_1(n) \approx \tilde{x}_1(n) \quad \text{and} \quad x_2(n) \approx \tilde{x}_2(n)$$

Depending on the values of M_1 and n above, approximations can be as close as desired. This is depicted in Figure 5.2.

For $n \leq n_0$, let the error be less than a desired threshold δ_0 . We can set $\delta_0 \approx K_2 \mu^{-n_0}$. For purposes of computation we can assume that $n \leq n_0$

$$x_1(n) = \tilde{x}_1(n).$$

As a consequence of the error bounds, to achieve a particular value δ_0 for the error as $\mu \rightarrow 1$, we need to choose M_1 such that $(M_1 - n_0)$ is large. When μ is very close to unity, M_1 will become inordinately large, making it almost impossible to extract a "good" section $\tilde{x}_1(n)$ from the computed section $x_1(n)$. As a corollary, when $\mu = 1$, we need to use an infinite length data segment. If the original signal $x(n)$

Figure 5.2(a): Actual Input $\tilde{x}_1(n)$

(b): Computed $x_1(n)$

(c): Error $x_1(n) - \tilde{x}_1(n)$

was long but of finite length, this implies that we have to use the full length of output $y(n)$, although its length is also finite. This is borne out by the simulation results as shown in the graph in Figure 5.3.

In Figure 5.3 we have plotted the values of the error length $M_1 - n_0$ required to achieve $\delta_0 = 10^{-3}$ versus the $(-\ln \mu)^{-1}$. The filter used in the problem had all its zeros fixed except a set of four symmetrically placed zeros, whose M value was varied from $\mu = 0.3$ to 0.95. The curve joining the points is close to a straight line, indicating the validity of the assumption $\delta_0 = K\mu^{-n_0}$.

It is interesting to note that in equations 5.2.3a and b, if we could compute both $x_1(n)$ and $x_2(n)$, then their point by point addition should be able to cancel the errors exactly. This fact has been verified by actual simulation for a signal segmented into two pieces as shown in Figures 5.4 to 5.7. Figure 5.4 is an original 229 points long signal which was passed through a filter to give $y(n)$ (Figure 5.5). The output $y(n)$ was segmented into two pieces at $M_1 = 150$ and the two respective inputs $x_1(n)$ (Figure 5.6) and $x_2(n)$ (Figure 5.7) were computed. It is clear that a substantial portion of $x_1(n)$ is similar to the corresponding portion of $\tilde{x}_1(n)$. Over the length of $n_0 = 75$, the signal energy to error energy ratio (SER) was 67 db. The signal $x_2(n)$ behaves in the same way. The two computed inputs $x_1(n)$ and $x_2(n)$ were added point-by-point (Figure 5.8) and an SER of 66db was achieved.

Figure 5.3: Plot of Error Length v/s $[-\log(\mu)]^{-1}$

Figure 5.4: Signal $x[n]$

Figure 5.5: Output Signal $y(n)$

Figure 5.6: Magnitude Weighted Reconstruction of $x_1(n)$

Figure 5.7: Magnitude Weighted Reconstruction of $x_2(n)$

Figure 5.8: Sum of $x_1(n]$ and $x_2(n]$

5.3. Phase-Only Blind Deconvolution: Multisegment Case

The two-segment signal reconstruction case discussed in section 5.2 can be used to design a procedure to recursively compute the full input $x(n)$, from multi-segment output $y(n)$, when $\mu < 1$.

Assume that we are able to handle a segment of length M which is long enough to be able to yield a reasonable long segment of "good" input reconstruction. For our purpose, we assume that the length of "good" segment is $\frac{M}{2}$. The given output $y(n)$ is segmented into $\frac{M}{2}$ point long sections, labeled $s_1, s_2, \dots$, etc. The corresponding inputs are denoted by $d_1, d_2, \dots$, etc. We start by choosing two pieces s_1 and s_2 (which put together are of length $M+L$), and compute an input of length $\frac{3M}{2}$. Out of this we retain a segment d_1 of length $\frac{M}{2}$.

Now, the input sequence is shifted by length $\frac{M}{2}$ so that the segment s_1 is entirely in the negative time domain. Using the segments s_1, s_2 , and s_3 (where s_1 being in the negative time domain is wrapped around modulo the length of DFT during the calculation of phase) and the known segment d_1 , we compute another length $\frac{M}{2}$ segment of input. Out of this we retain the length $\frac{M}{2}$ segment and denote it as d_2 . Next time, s_1 and s_2 are wrapped in the negative time domain. Using d_1 and d_2 , d_3 is computed. The process continues until the data runs out. The problem of computing an unknown portion of segment using the data $y(n)$ and the known portion of input has been dealt with in Section 2.7. Figure 5.9 shows a pictorial representation of the above algorithm.

Figure 5.9: Phase-Only Blind Deconvolution — Segmentation technique

By way of illustration, the same output $y(n)$ as in Section 5.2 was run through the multisegment algorithm of this section. A segment of length 175 was used to start the algorithm. From the computed output of length 255, a section of length 50 samples was retained for d_1 . Here δ_0 had a value 0.001. Using the retained segment d_1 as a known segment, the rest of the input signal was built up recursively until the end of the data. The reconstituted input (Figure 5.10) has an SER of 70 db over the full input.

The previous example involves enhancement of the higher frequency components as seen in Figure 5.5. To investigate the behavior when the higher frequency terms are attenuated a signal $x(n)$ (Figure 5.11) was passed through a lowpass filter to give $y(n)$ (Figure 5.12). The signal $x(n)$ contained two narrow band components, one centered at 0.1 and the other at 0.3. The reconstruction of $x(n)$ using the segments of $y(n)$ is plotted in Figure 5.13. The starting segment was 150 points long and in each iteration 25 points were retained. The error in the reconstructed signal (Figure 5.14) was 51.3 db.

5.4. Conclusions

In this chapter we introduced a segmentation technique for Phase-Only Blind Deconvolution. This enables us to compute the full length of a long input without the benefit of precise knowledge of the blurring function $h(n)$. Some results of actual application are presented along with some bounds. We see that an excellent reconstruction of the input signal is achieved.

Figure 5.10: Signal Reconstruction of $x(n)$ by Segmentation

Figure 5.11: Signal $x[n]$

Figure 5.12: Signal $y(n)$ (filtered Output)

Figure 5.13: Signal Reconstruction of $x(n)$ by Segmentation

Figure 5.14: Error in Signal Reconstruction

CHAPTER 6

SOME GENERALIZATIONS OF PHASE-ONLY RECONSTRUCTION

6.1. Introduction

In the previous chapters we have discussed in detail the theory of Phase-Only reconstruction and its application to blind deconvolution. The filters involved are assumed to be symmetric or linear phase. A logical adjunct question arises: is it possible to allow some kind of generalization so that one can perform blind deconvolution in a manner analogous to Phase-Only blind deconvolution? In this chapter we present the theory which indicates that such an extension is possible and allows us to define new kinds of structures for the filters which are to be deconvolved.

6.2. Phase-Only Blind Deconvolution: A Structure Invariant Approach

Consider a signal $x(n)$, $n=0, 1, \dots, M$ with no symmetric filter, as an input to a filter H with a symmetric filter response $h(n)$, $n=0, \pm 1, \pm 2, \dots, \pm L$, and its output $y(n)$, $n=-L, \dots, 0, 1, \dots, L+M$. Then, the z transform of the output is

$$Y(z) = H(z) X(z) \quad 6.1$$

Since $h(n)$ is symmetric $H(z) = H(z^{-1})$. Define a map $f: \mathbb{C} \rightarrow \mathbb{C}$ ($\mathbb{C}$ = complex plane) such that $f(z) = z^{-1}$. Then $H(z)$ is invariant under the map f on its argument. Thus, if there exists a signal $x_1(n)$,

$n=0, 1, \dots, M$, with no symmetric factor, such that

$$X_1(z) Y(z^{-1}) = X_1(z^{-1}) Y(z)$$

then

$$X_1(z) = A X(z)$$

where A is a real constant. This has been proved in chapter 2. In subsequent sections we seek to generalize the notion of invariant filter $H(z)$ under the symmetry condition. However, as a preliminary to this objective we need to define some notation.

6.3. Notation

Let f be a map

$$f: \mathbb{C} \rightarrow \mathbb{C}$$

such that f is analytic almost everywhere except perhaps at the origin. An example of such a map is $f(z) = z^{-1}$.

f-symmetric polynomial

Let $H(z)$ be a polynomial in z and z^{-1} such that

$$H(f(z)) = U(z) H(z)$$

where $U(z)$ is a fixed polynomial independent of the coefficients of z in $H(z)$. $H(z)$ is said to be an f -symmetric polynomial. An example of an f -symmetric polynomial under the map $f(z) = z^{-1}$ is an even length polynomial

$$H(z) = h_0 + h_1 z^{-1} + \dots + h_L z^{-L} + h_{L+1} z^{-L-1} + \dots + \dots + h_{2L+1} z^{-2L-1}$$

such that

$$h_0 = h_{2L+1}, h_1 = h_{2L}, \dots h_L = h_{L+1}$$

Then

$$\begin{aligned} H(f(z)) &= H(z^{-1}) = \sum_{r=0}^{2L+1} h_r z^r \\ &= z^{2L+1} \sum_{r=0}^{2L+1} h_r z^{-r} = z^{2L+1} H(z) \end{aligned}$$

Thus, in this case $U(z) = z^{2L+1}$. In all the applications $U(z)$ will be assumed to be constant or a power of z .

f-symmetric factor

Let $x(n)$, $n=0, 1, \dots, M$ be a signal such that its Z-transform $X(z) = \sum_n x(n) z^{-n}$ can be factored into two factors $X(z) = X_1(z) X_2(z)$ such that $X_1(z)$ is f-symmetric. Then $x(n)$ is said to have an f-symmetric factor.

relatively prime

Two polynomials $X_1(z)$ and $X_2(z)$ are said to be relatively prime if there does not exist a polynomial $X_3(z)$ such that

$$X_3(z) \mid X_1(z) \text{ and } X_3(z) \mid X_2(z)$$

This is denoted by

$$X_3(z) = (X_1(z), X_2(z)) = \text{constant}$$

If $X_1(z)$ and $X_2(z)$ are not relatively prime then

$$X_3(z) = (X_1(z), X_2(z)) \neq \text{constant}.$$

relatively f-prime

Two polynomials $X_1(z)$ and $X_2(z)$ are said to be relatively f-prime if there does not exist an f-symmetric polynomial $X_3(z)$ such that

$$X_3(z) \mid X_1(z) \text{ and } X_3(z) \mid X_2(z)$$

This is denoted by

$$X_3(z) = (X_1(z), X_2(z))_f = \text{constant}$$

Note that this does not imply that $X_1(z)$ and $X_2(z)$ are necessarily relatively prime.

orbit of a zero

Let z_0 be a zero of an f-symmetric polynomial $H(z)$. Then z_0 , $f(z_0)$, $f(f(z_0)) = (f)^2(z_0)$, $f(f(f(z_0))) = (f)^3(z_0)$, etc. are all zeros of $H(z)$. Since $H(z)$ has a finite number of zeros, there exists an integer n such that $(f)^n(z_0) = (z_0)$. The sequence $z_0, (f)^2(z_0), \dots, (f)^{(n-1)}(z_0)$ is the orbit of z_0 , and is of length n . This orbit under map f will be denoted $\text{Or}(z_0, f)$. Each member of the $\text{Or}(z_0, f)$ is a zero of the same multiplicity as z_0 .

6.4. Uniqueness Theorems

We need the following two lemmas to prove the uniqueness theorems.

Lemma 6.1

Given any two real nontrivial polynomials $X(z)$ and $Y(z)$, their greatest common divisor $g(z)$ is a real polynomial.

Proof: Since the greatest common divisor $g(z)$ is computed by the Euclidean algorithm, it is possible to write

$$g(z) = p_1(z) X(z) + p_2(z) Y(z)$$

where $p_1(z)$ and $p_2(z)$ are real polynomials. Thus $g(z)$ is a real polynomial.

Lemma 6.2:

Let a given polynomial $X(z)$ be factorizable into two polynomials $X(z) = X_1(z) X_2(z)$. Then if $X(z)$ and $X_1(z)$ are real polynomials, it implies that $X_2(z)$ is a real polynomial.

Theorem 6.1

Let $x(n)$, $n=0, 1, \dots, M$ and $y(n)$ be two nontrivial finite length sequences such that:

- (1) the greatest common divisor (gcd) of $X(z)$ and $X(f(z))$ is not divisible by any nontrivial real coefficient polynomial

(2) $x(n)$ and $y(n)$ have the same support

OR

(3) the gcd of $Y(z)$ and $Y(f(z))$ is not divisible by any nontrivial real coefficient polynomial;

(4) $X(z) Y(f(z)) = U(z) X(f(z)) Y(z)$, $U(z) = cz^{n_0}$, for some positive integer n_0 .

Then there exists a real constant A such that

$$X(z) = A Y(z)$$

Proof:

The $(X(z), U(z))$ is constant (condition 1), hence $X(z) \mid X(f(z)) Y(z)$ (condition 4). Let $X(z) = X_1(z) X_2(z)$ such that $X_1(z)$ is the greatest common divisor of $X(z)$ and $Y(z)$ and $X_2(z) \mid X(f(z))$. Since $X(z)$ and $Y(z)$ are real polynomials, by Lemma 6.1 $X_1(z)$ is a real polynomial. Furthermore, since $X(z)$ and $X_1(z)$ are real polynomials, by Lemma 6.2 $X_2(z)$ is a real polynomial, and $X_2(z) \mid X(f(z))$. This is a contradiction of condition 1 above. Hence $X_2(z)$ is a constant and $X(z) \mid Y(z)$. Now, if condition 2 holds, then $X(z)$ and $Y(z)$ have the same highest and the lowest powers of z^{-1} , and hence $X(z) = A Y(z)$ where A is a real constant. On the other hand, if condition 3 holds, then $Y(z) \mid X(z)$ and hence $X(z) = A Y(z)$ where A is a real constant.

Theorem 6.2:

Let $Y(z)$ be a real polynomial such that it can be factored as

$$Y(z) = H(z) X(z)$$

where $H(z)$ is the largest nontrivial f -symmetric factor of $Y(z)$ and the degree of polynomial $X(z)$ is $\deg(X)$. Then if another real polynomial $X_1(z)$ has $\deg(X_1) = \deg(X)$ and

$$X_1(z) Y(f(z)) = U(z) X_1(f(z)) Y(z)$$

and the greatest common divisor of $X_1(z)$ and $X_1(f(z))$ has no real polynomial factors then

$$X_1(z) = A X(z)$$

where A is a real coefficient.

Proof:

Since $H(z)$ is an f -symmetric factor

$$H(f(z)) = U(z) H(z)$$

Further

$$X_1(z) Y(f(z)) = X_1(z) H(f(z)) X(f(z))$$

$$= U(z) X_1(z) X(f(z)) H(z)$$

and

$$X_1(f(z)) Y(z) = X_1(f(z)) X(z) H(z)$$

Then, since the right hand side of the two expressions above have the polynomial $H(z)$ as the common factor

$$X_1(z) Y(f(z)) = U(z) X_1(f(z)) Y(z)$$

iff

$$X_1(z) X(f(z)) = U(z) X_1(f(z)) X(z)$$

But, by Theorem 6.2, this implies

$$X_1(z) = A X(z)$$

where A is a real constant.

Theorem 6.2 allows us to perform blind deconvolution to obtain the original signal $x(n)$ without the knowledge of filter $H(z)$, except for the fact that $H(z)$ is invariant under the map f .

6.5. Some Examples of f -Symmetric Filters

Phase-Only blind deconvolution is characterized by the map $f_1(z) = z^{-1}$. Here $H(z)$ is a symmetric or a linear phase polynomial. Figure 6.1 shows the kinds of zero placement which $f_1(z)$ allows. Other possible examples are

$$f_2(z) = -z$$

Any filter $H(z)$ with only even or only odd powers of z^{-1} is f -symmetric. Such f -symmetric filters are symmetric around the origin and have their zeros symmetrically placed around the real and imaginary axes (Figure 6.2). An interesting sidelight is that it is possible to scale the zeros, if there are any on the unit circle,

Figure 6.1: $f_1(z) = z^{-1}$

Figure 6.2: $f_2(z) = -z$

without loss of symmetry.

A third example is

$$f_3(z) = -z^{-1}$$

This is essentially a combination of $f_1(z)$ and $f_2(z)$ (Figure 6.3). The previous three cases involve maps which are real. However, it is possible to consider complex maps. For example,

$$f_4(z) = e^{\frac{j 2\pi}{3}} z$$

will deconvolve filters with zeros placed 120 degrees apart on a circle of fixed radius (Figure 6.4).

A more complex map is

$$f_s(z) = s_0 z$$

where s_0 is a root of $s^{2n+1} - 1 = 0$. Then a filter such as $1 + z^{-2n-1}$ is symmetric under map f_s . This filter can be used for band compression. The reduced band signal can now be sent out at a lower rate and then the original signal can be recovered.

6.6. An Application with $f(z) = -z$

In this section we will present a small example to indicate the validity of the generalizations presented in the preceding section.

Let

$$H(z) = 1 + 4z^{-2} + 3z^{-4}$$

Figure 6.3: $f_3(z) = -z^{-1}$

Figure 6.4: $f_4(z) = e^{i\frac{2\pi}{3}} z$

$$S(z) = 1 + 2z^{-1} + 2z^{-2}$$

Then

$$Y(z) = H(z) S(z) = 1 + 2z^{-1} + 6z^{-2} + 8z^{-3} + 11z^{-4} + 6z^{-5} + 6z^{-6}$$

Here we use $f(z) = -z$ and $U(z) = 1$. To reconstruct $S(z)$ without using the knowledge of $H(z)$, we set up the following equality

$$X(z) Y(-z) = X(-z) Y(z)$$

where $X(z)$ is a real polynomial of degree 2. Then the time domain version of the above equality is

$$\sum_{r=0}^2 ((-1)^{n-r} - (-1)^r) y(n-r) x(r) = 0$$

for all n . For even n the above equations are identically zero. For odd n we get

$$-2 \sum_{r=0}^2 y(n-r) x(r) (-1)^r = 0$$

Using $n=1$ and 3 and $x(0) = 1$, we get the system of equations

$$\begin{bmatrix} 2 & -1 & 0 \\ 8 & -6 & 2 \end{bmatrix} \begin{bmatrix} 1 \\ x(1) \\ x(2) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

which gives $x(1)=2$ and $x(2)=2$, as expected.

6.7. Conclusions

In this chapter, we have presented a generalization of Phase-Only deconvolution. It is shown that there are other kinds of symmetries which can be utilized to reconstruct the original signal.

Naturally, the map under which the filter $H(z)$ is invariant has to be known beforehand, along with $U(z)$. However, this would be a part of the design of $H(z)$, or it could arise from the basic nature of $H(z)$. For example, in well logging, resistivity measurement is done by probes which are inherently symmetric around their centers. Further work needs to be done to see how this generalization can be used for practical applications.

CHAPTER 7

CONCLUSIONS

7.1. The work in this Dissertation

Chapter 1 of this dissertation has presented a brief review of the state of the art on the problem of deconvolution. The difficulties inherent in the problem have also been discussed. Following this, Chapter 2 has considered the concept of phase-only reconstruction. Theorems discussing the conditions under which this technique yields unique results have been presented. Various formulations of arriving at a system of linear equations have been outlined. A few plots of the actual examples have been shown.

Chapter 3 discusses the concept of applying phase-only reconstruction to blind deconvolution. In this chapter, a new idea of magnitude-weighted phase-only blind deconvolution is presented. It is shown that for most problems the magnitude-weighted version of the problem gives a solution as good as the one obtained by "ordinary" constant magnitude phase-only reconstruction, yet it requires less calculation, and input does not even need the computation of either Fourier transform of the phase. When the signal to be recovered has been passed through a filter, with zeros near or on the unit circle, magnitude-weighted phase-only blind deconvolution gives a superior performance. The example presented shows that it outperforms the constant magnitude approach by approximately 8 db. for all noisy signals

with additive noise with the standard deviation ranging from 10^{-6} to 10^{-3} . Thus, for narrow band signals, the magnitude-weighted scheme is recommended.

In Chapter 4, we present a new, fast algorithm to solve the system of linear equations which comes out of the time domain formulation of phase-only blind deconvolution. This system is representable as the sum of a Toeplitz and a Hankel matrix, and can be converted to a block-Toeplitz problem, which is solved by block-Levinson algorithms. The symmetries inherent in the problem allow the amount of computation to be cut by almost fifty per cent. A new way of eliminating singular matrices has been proposed.

The previous formulations have required the full length of output of a system to be available before phase-only blind deconvolution can be applied. This restricts its usefulness in handling long data. In Chapter 5, a segmentation technique has been developed. It is shown that with an additional constraint of a symmetric filter (viz. no unit circle zeros), segmentation of data is allowed. The segmented phase-only blind deconvolution technique has given excellent results, as shown in Figures 5.6, 5.7 and 5.8.

Finally, in Chapter 6, we present a generalization of phase-only blind deconvolution. It is seen that there exist a variety of symmetries under which a filter can be deconvolved without actual knowledge of the values of the filter coefficients.

7.2. Future Work

One major problem with the segmentation technique of Chapter 5 is the requirement that the filter have no zeros whose magnitude approaches unity. This is a fundamental limitation of all deconvolution schemes. Further work needs to be done in this area.

The generalizations presented in Chapter 6 are not exhaustive. Only a basic framework for the generalization of symmetries has been presented. More work needs to be done on generalization, and, in particular, on the kinds of maps which will allow new kinds of symmetries. It is this author's hope that both the work presented here and the potential future development suggested by it will make useful contributions to the various fields of Digital Signal Processing.

REFERENCES

- [1] Prony, R., "Essai Experimental et Analytique sur les Loix de la Dilatabilite des Fluides Elastiques et sur Celles de la Force Expansive de la Vapeur de l'eau et la vapeur de L'Alcool, A Differentes Temperatures," J. L'Ecole Polytech. 1(2), pp.24-76, Paris (1795).
- [2] McDonough, R. N., "Matched Exponent for the representation of Signals," Ph.D. Dissertation, Dept. of Electrical Engineering, The John Hopkins University (1963).
- [3] Weiss, L. and , R. N. McDonough, "Prony's Method, Z-transform and Pade Approximation ," SIAM Review 5(2), pp.145-149 (April 1963).
- [4] Levinson, N., "The Weiner RMS (root mean square) Error Criterion in Filter Design and Prediction," J. Math. and Phys. 25, pp.261-278 (1947).
- [5] Hildebrand, F. B., Introduction to Numerical Analysis, McGraw-Hill Inc. (1974).
- [6] Wiener, N., "The extrapolation, interpolation and smoothing of stationary time series with engineering application," MIT DIC Contract., No.6037, National Defence Council, Section D2, Cambridge, Mass. (1942). Reprinted 1949, Wiley, New York, N.Y.
- [7] Boonton, R. C., "An optimization theory for time varying systems with nonstationary inputs," Proceedings IRE 40, pp.977-981 (1952).

- [8] Kalman, R. E., "New Methods of Wiener Filtering Theory," Proc. 1st Symp. Engineering Applications of Random Function Theory and Probability, pp.270-388, Wiley, New York (1963).
- [9] Francesco, Count Ricatti Jacopo, "Animadversiones in aequationes differentiales secundi gradus," Actorum Erunditorum quae Lipsiae publicantur Suppl.8, pp.66-73 (1724).
- [10] Kailath, T., "A view of Three Decades of Linear Filtering Theory," IEEE Trans. Info. Theory IT-20, pp.146-181 (March 1974).
- [11] Silverman, L. M., "Properties and application of inverse systems," IEEE Trans. Automatic Control AC-13, pp.436-437 (August 1968).
- [12] Silverman, L. M., "Inversion of Multivariable Linear Systems," IEEE Trans. Automatic Control AC-14(3), pp.270-278 (June 1969).
- [13] Moylan, P. J., "Stable Inversion of Linear Systems," IEEE Trans. Automatic Control (February 1977).
- [14] Bayless, J. W. and Brigham, E. O., "Application of the Kalman Filter to Continuous Signal Restoration," Geophysics 35(1), pp.2-23 (1970).
- [15] Crump, N. D., "A Kalman Filter Approach to the Deconvolution of Seismic Signals," Geophysics 39(1), pp.1-13 (1974).
- [16] Wardsworth, G. P., Robinson, E. A., Bryan, J. G., and Hurley, P. M., "Detection of Reflection on Seismic records by Linear Operators," Geophysics 18, pp.539-586.

- [17] Robinson, E. A., "Predictive Decomposition of Time Series with Applications to seismic exploration," Ph.D. Dissertation, MIT (1954).
- [18] Itakura, F. and Saito, S., "A Statistical Method for Estimation of Speech Spectral Density and Formant Frequencies," Electron. Commun. Japan 53-A, pp.36-43 (1970).
- [19] Gray, Jr., A. H. and Markel, J. D., "A Spectral Flatness Measure for studying the Autocorrelation Method of Linear Prediction of Speech Analysis," IEEE Transactions on Acoustics, Speech and Signal Processing ASSP-22, pp.207-217 (1974).
- [20] Narasimha, M. J., Shenoi, K., and Peterson, A. M., "A Hilbert Space approach to Linear Predictive Analysis of Speech Signals," Tech. Rep.3606-10, Radio Science Lab., Stanford Electronics Lab., Stanford University, California (1974).
- [21] Morf, M., Dickinson, B., Kailath, T., and Vieira, A., "Efficient Solution of Covariance Equations for Linear Prediction," IEEE Transactions on Acoustics, Speech and Signal Processing ASSP-25(5), pp.429-433 (Oct. 1977).
- [22] Poisson, S. D., "Sur la distribution de la chaleur dans les corps solides," J. Ec. R. Polytech. I(19), pp.1-162 (1823).
- [23] Schwarz, H. A., "Zur Integrator der partillen Differentialgleichung," Z. Reine Angewandte. Math., pp.218-254 (1872).

- [24] Szego, G., "Ein Grenzwertsatz Uber die Toeplitzchen Determinantes einer reell positiv Funktion," Math. Ann. 76, pp.490-803 (1915).
- [25] Kolmogorov, A. N., "Sur l'interpolation et extrapolation des suites stationnaires," C. R. Acad. Sci. 208, pp.2043-2068, Paris (1959).
- [26] Oppenheim, A. V., "Speech Analysis-Synthesis System Based on Homomorphic Filtering," Journal of Acoust. Soc. of Am. 45(2), pp.458-465 (February 1969).
- [27] Prost, R. and Goulte, R., "Deconvolution when the convolution kernel has no inverse," IEEE Trans. on ASSP ASSP-25(6), pp.542-549 (December 1977).
- [28] Brigham, E. O., Smith, H. W., Bostick, F. X., and Dueterhoeft, W. C., "An Iterative Technique for Determining Inverse Filters," IEEE Trans. on Geoscience Electronics GE-6(2), pp.86-96 (May 1968).
- [29] Anderson, B. O., "Practical System Inversion without Impulse Response Approximation," IEEE Trans. on Circuits and Systems CAS-26(11), pp.965-967 (November 1979).
- [30] Ekstrom, M. P., "A spectral characterization of the ill-conditioning in numerical deconvolution," IEEE Trans. on Audio and Electro-Acoustics AU-21(4) (August 1973).

- [31] Hunt, B. R., "A Theorem on the Difficulty of Numerical Deconvolution," IEEE Trans. on Audio Electroacoustics **AU-20**, pp.94-95 (March 1972).
- [32] Grenander, W. and Szego, G., Toeplitz forms and their applications, University of California Press, Berkeley and Los Angeles (1958).
- [33] Hayes, M. H., Lim, J. S., and Oppenheim, A. V., "Signal Reconstruction from Phase or Magnitude," 1979-64, Lincoln Laboratory, M.I.T. (Sept. 1979).
- [34] Hayes, M. H., Lim, J. S., and Oppenheim, A. V., "Phase-Only Reconstruction," ICASSP80 Proceedings 2, pp.437-440 (April 1980).
- [35] Oppenheim, A. V. and Schaffer, R. W., in Digital Signal Processing, Prentice-Hall, Englewood Cliffs, N.J. (1975).
- [36] Oppenheim, A. V., Lim, J. S., Kopec, G., and Pohlig, S. C., "Phase in Speech and Pictures," ICASSP-79 Record, pp.632-637, 1979 IEEE International Conference on Acoustic, Speech Signal Processing, Washington, D.C. (April 1979).
- [37] Hayes, M. H., Lim, J. S., and Oppenheim, A. V., "Signal Reconstruction from Phase or Magnitude," IEEE Transactions on Acoustics, Speech and Signal Processing **ASSP-28(6)**, pp.672-680 (December 1980).

- [38] Rabiner, L. R. and Gold, B., Theory and Application of Digital Signal Processing, Prentice-Hall, Englewood Cliffs, NJ (1975).
- [39] Burrus, C. S. and Parks, T. W., "Time Domain Design of Recursive Digital Filters," IEEE Trans. on Audio Electroacoust. **AU-18**, pp.137-141 (June 1970).
- [40] Tribolet, J. M., "A New Phase Unwrapping Algorithm," IEEE Transactions on Acoustics, Speech and Signal Processing **ASSP-25(2)**, pp.170-177 (April 1977).
- [41] Merchant, G. A. and Parks, T. W., Efficient solution of Toeplitz-plus-Hankel Coefficient Matrix System of Equations, Rice Tech Report 8003 (Mar, 1980).
- [42] Merchant, G. A. and Parks, T. W., "Efficient solution of Toeplitz-plus-Hankel Coefficient Matrix System of Equations," IEEE Transactions on Acoustics, Speech and Signal Processing **ASSP-30(1)**, pp.40-44 (February 1982).
- [43] Akaike, H., "Block Toeplitz Matrix Inversion," SIAM J. APPL. MATH. **24(2)**, pp.234-241 (March 1973).
- [44] Gragg, W. B. and Johnson, G. D., "The Laurent-Pade Table," Proceedings of IFIP CONGRESS 1974, pp.632-637, North-Holland Publishing Co. (1974).
- [45] Aki, K. and Richards, P. G., Quantitative Seismology Theory and Methods, W. H. Freeman and Co., San Fransisco (1980).

- [46] Dugre', J., Beex, A. A. L., and Scharf, L. L., "Generating Covariance Sequences and Calculation of Quantization and Rounding Error Variances in Digital Filters," IEEE Transactions on Acoustics, Speech and Signal Processing ASSP-28(1), pp.102-104 (February 1980).
- [47] Good, I. J., "The Inverse of a Centrosymmetric Matrix," Technometrics 12(4), pp.925-928 (November 1970).
- [48] Hestenes, M. R. and Stiefel, E., "Methods of Conjugate Gradient for solving linear systems," Journal of Res. of Nat. Bureau of Std. 49(6), pp.407-436 (December 1952).
- [49] Schafer, R. W., "Echo removal by discrete generalized linear filtering," Ph.D Dissertation, Dept. of Elec.Eng., M.I.T. (1968).

Appendix A

Proof:

Let $g(n)$ be an odd length sequence of length $M+1$. Then $g(0) \neq 0$ and $g(M) \neq 0$. Let $g(n)$ be a linear phase sequence. The Fourier transform of $g(n)$ is

$$G(\omega) = |G(\omega)| e^{j\Phi_g(\omega)} = \sum_{n=0}^M g(n) e^{-jn\omega}.$$

Let $\Phi_g(\omega) = -\alpha\omega + \Theta(\omega)$, where $\Theta(\omega) = 0$ or π . The equation above can be rewritten as

$$e^{j\alpha\omega} G(\omega) = |G(\omega)| e^{j\Theta(\omega)} = \sum_{n=0}^M g(n) e^{-j(n-\alpha)\omega}. \quad A.1$$

Since $e^{j\Theta(\omega)} = \pm 1$ equation A.1 is real for all ω or

$$\sum_{n=0}^M g(n) \sin(n-\alpha)\omega = 0.$$

Substituting $m=n-\alpha$ and assuming α to be an integer

$$\sum_{m=-\alpha}^{M-\alpha} g(\alpha+m) \sin(m\omega) = 0$$

or

$$\sum_{m=1}^{M-\alpha} g(\alpha+m) \sin(m\omega) - \sum_{m=1}^{\alpha} g(\alpha-m) \sin(m\omega) = 0 \quad A.2$$

Define a new sequence

$$\tilde{g}(m) = \begin{cases} g(\alpha+m) - g(\alpha-m) & 1 \leq m \leq \alpha \\ g(\alpha+m) & \alpha < m \leq M-\alpha \end{cases}$$

Equation A.2 becomes

$$\sum_{m=1}^{M-\alpha} \tilde{g}(m) \sin(m\omega) = 0$$

Since the set of functions

$$\{\sin(m\omega), m=1,2,\dots\}$$

is a linearly independent set $\tilde{g}(m) = 0$ for all m i.e.

$$g(\alpha + m) = g(\alpha - m) \quad 1 \leq m \leq \alpha$$

$$g(\alpha + m) = 0 \quad \alpha < m \leq M - \alpha$$

Since $g(M) \neq 0$ we need

$$\alpha \geq M - \alpha \quad \text{or} \quad \alpha \geq \frac{M}{2}.$$

On the other hand if $\alpha > \frac{M}{2}$ or $2\alpha \geq M+1$ then since $g(2\alpha) = g(0)$ and $g(n)$ is 0 for all $n \geq M+1$ we have $g(0) = 0$ which is contrary to assumption. Consequently,

$$\alpha = \frac{M}{2} \quad \text{and} \quad g(n) = g(M-n), \text{ for } n=0,1,\dots,M.$$

The converse assertion can be proved trivially and the proof will not be presented.

Appendix B

Starting from equation 2.4.7, we derive the normal system of equations. The (r_1, r_2) th term of the coefficient matrix is

$$\begin{aligned}
 &= \sum_{n=-\infty}^{\infty} [\phi(-n + r_1) - \phi(n + r_1)][\phi(-n + r_2) - \phi(n + r_2)] \quad \text{B.1} \\
 &= \sum_{n=-\infty}^{\infty} \phi(-n + r_1) \phi(-n + r_2) + \sum_{n=-\infty}^{\infty} \phi(n + r_1) \phi(n + r_2) \\
 &\quad - \sum_{n=-\infty}^{\infty} \phi(-n + r_1) \phi(n + r_2) - \sum_{n=-\infty}^{\infty} \phi(n + r_1) \phi(-n + r_2) .
 \end{aligned}$$

By substituting $-n$ for n in the first and third summations on the right, it can be seen that they are equal to the second and fourth summations respectively. Thus, the (r_1, r_2) th element is

$$(r_1, r_2)\text{th element} \quad \text{B.2}$$

$$\begin{aligned}
 &= 2 \left[\sum_{n=-\infty}^{\infty} \phi(n + r_1) \phi(n + r_2) - \sum_{n=-\infty}^{\infty} \phi(n + r_1) \phi(-n + r_2) \right] \\
 &= 2 \left[\sum_{n=-\infty}^{\infty} \phi(n) \phi(n - (r_1 - r_2)) - \sum_{n=-\infty}^{\infty} \phi(n) \phi(-n + (r_1 + r_2)) \right]
 \end{aligned}$$

Now, from equation 2.4.6

$$1 = e^{j\Phi_x(\omega)} e^{-j\Phi_x(\omega)} = \left(\sum_{n=-\infty}^{\infty} \phi(n) e^{-jn\omega} \right) \left(\sum_{n=-\infty}^{\infty} \phi(n) e^{jn\omega} \right)$$

and

$$e^{j2\Phi_x(\omega)} = \left(\sum_{n=-\infty}^{\infty} \phi(n) e^{-jn\omega} \right)^2$$

Hence

$$\sum_{r=-\infty}^{\infty} \phi(r) \phi(r-n) = \delta(n) \quad \text{B.3a}$$

$$\sum_{r=-\infty}^{\infty} \phi(r) \phi(n-r) = \tilde{\eta}(n) \quad \text{B.3b}$$

where $\delta(n)$ is the delta function, and $\tilde{\eta}(n)$ is the inverse Fourier transform of $e^{j2\Phi_x(\omega)}$. From equations B.3a and b,

$$(r_1, r_2)\text{th element} = 2[\delta(r_1 - r_2) - \tilde{\eta}(r_1 + r_2)] .$$

Similarly, the r_1 -th constant term is $2(\tilde{\eta}(r_1) - \delta(r_1))$. Thus, the normal system of equations, after division by 2, is

$$\sum_{r_2} [\delta(r_1 - r_2) - \tilde{\eta}(r_1 + r_2)] x(r_2) = \tilde{\eta}(r_1) - \delta(r_1)$$

for all r_1 . This is similar to the system of equations in 2.5.10 for $r_1 > 0$. If we follow the procedure outlined in this appendix for an aliased version of the system of equations in 2.4.7 (obtained when the calculation are done using DFTs) we get a system of equation identical to 2.5.10.

Appendix C

Since $I^{\tau\pi} = I$ and $R_0^{\tau\pi} = R_0$, using the notation of section 2.4.2 we have

$$\{Y_0^0\}^{\tau\pi} = Y_0^0 = I \quad \text{and} \quad \{V_y^0\}^{\tau\pi} = V_x^0 = R_0.$$

Following will be proved using induction.

- a) $E_y^k = \{E_x^k\}^{\tau\pi}$
- b) $B_y^k = \{B_x^k\}^{\tau\pi}$
- c) $Y_r^k = \{X_r^k\}^{\tau\pi}$
- d) $V_y^k = \{V_x^k\}^{\tau\pi}$

Assume that a), b), c) and d) are true for all integers up to $k-1$. In

this appendix $J = J_2 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$.

$$\text{i) } E_x^k = \sum_{j=0}^{k-1} R_{k-j} X_j^{k-1}$$

$$\begin{aligned} \text{Then } \{E_x^k\}^{\tau\pi} &= J E_x^k J = \sum_{j=0}^{k-1} J R_{k-j} J \quad J X_j^{k-1} J \\ &= \sum_{j=0}^{k-1} R_{k-j} Y_j^{k-1} \\ &= \sum_{j=1}^k R_{-j} Y_{i-j}^{k-1} = E_y^k. \end{aligned}$$

$$\text{ii) } B_x^k = \{V_y^{k-1}\}^{-1} E_x^k$$

$$\text{Then } \{B_x^k\}^{\tau\pi} = J B_x^k J = \{J V_y^{k-1} J\}^{-1} J E_x^k J$$

Using the results (i) and (d) $\{B_x^k\}^{\tau\pi} = \{V_x^{k-1}\}^{-1} E_y^k = B_y^k$.

$$\text{iii) } X_r^k = X_r^{k-1} - Y_{k-r}^{k-1} B_x^k \quad \text{for } r=0,1,\dots,k.$$

where $Y_k^{k-1} = 0$

$$\text{Then } \{X_r^k\}^{\tau\pi} = J X_r^k J$$

$$= J X_r^{k-1} J - J Y_{k-r}^{k-1} J J B_x^k J$$

$$\text{Using (ii) and (c) } \{X_r^k\}^{\tau\pi} = Y_r^{k-1} - X_{k-r}^{k-1} B_y^k = Y_r^k$$

$$\text{iv) } V_x^k = V_x^{k-1} - E_y^k B_x^k$$

$$\text{Then } \{V_x^k\}^{\tau\pi} = J V_x^k J$$

$$= J V_x^{k-1} J - J E_y^k J J B_x^k J$$

$$= V_y^{k-1} - E_x^k B_y^k$$

(by i and ii.)

$$= V_y^k$$

Appendix D

Let $H(z) = H_1(z) H_1(z^{-1})$ where the zeros of $H_1(z)$ are all inside the unit circle, i.e.

$$H_1(z) = \prod_{i=1}^r (1 - z_i z^{-1})$$

and $|z_i| < 1$ for all i . Since $H_1(z)$ is a minimum phase function, we can compute its inverse

$$\begin{aligned} G_1(z) &= \frac{1}{H_1(z)} = \frac{1}{\prod_{i=1}^r (1 - z_i z^{-1})} \\ &= \sum_{i=1}^r \frac{1}{z_i H_1'(z_i) (1 - z_i z^{-1})} \end{aligned}$$

where we have expanded $G_1(z)$ in a partial fraction expansion. Hence,

$$g_1(n) = \begin{cases} \sum_{i=1}^r \frac{z_i^{n-1}}{H_1'(z_i)} & n \geq 0 \\ 0 & n < 0 \end{cases}$$

Since $G(z) = G_1(z) G_1(z^{-1})$

$$g(m) = \sum_{n=0}^{\infty} g_1(n) g_1(n+m)$$

For $m \geq 0$

$$\begin{aligned}
 g(m) &= \sum_{i=1}^r \sum_{j=1}^r \frac{1}{H'(z_i)H'(z_j)} \sum_{n=0}^{\infty} z_i^{n-1} z_j^{n+m-1} \\
 &= \sum_{i=1}^r \sum_{j=1}^r \frac{z_i^{-1} z_j^{m-1}}{H'(z_i) H'(z_j) (1-z_i z_j)} \\
 &= \sum_{j=1}^r \left(\sum_{i=1}^r \frac{z_i^{-1} z_j^{-1}}{H'(z_i) H'(z_j) (1-z_i z_j)} \right) z_j^m \\
 &= \sum_{j=1}^r \alpha_j z_j^m
 \end{aligned}$$

where

$$\alpha_j = \sum_{i=1}^r \frac{z_i^{-1} z_j^{-1}}{H'(z_i) H'(z_j) (1-z_i z_j)}$$

Similarly, it can be shown that for $m > 0$

$$g(-m) = \sum_{j=1}^r \alpha_j z_j^m = g(m)$$

or

$$g(m) = \sum_{j=1}^r \alpha_j z_j^{|m|}$$

for all m . Now, let

$$\mu = \max_j |z_j|$$

then

$$|g(m)| \leq \left(\sum_{j=1}^r |\alpha_j| \right) \max_j |z_j|^m$$

$$= \left(\sum_{j=1}^r |\alpha_j| \right) \mu^m$$

$$= K\mu^m$$

where

$$K = \sum_{j=1}^r |\alpha_j| \leq \sum_{j=1}^r \sum_{i=1}^r \frac{|z_i|^{-1} |z_i|^{-1}}{|H'(z_i)| |H'(z_j)| |1-z_i z_j|}.$$