INFORMATION TO USERS

This reproduction was made from a copy of a document sent to us for microfilming. While the most advanced technology has been used to photograph and reproduce this document, the quality of the reproduction is heavily dependent upon the quality of the material submitted.

The following explanation of techniques is provided to help clarify 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 complete continuity.

2. When an image on the film is obliterated with a round black mark, it is an indication of either blurred copy because of movement during exposure, duplicate copy, or copyrighted materials that should not have been filmed. For blurred pages, a good image of the page can be found in the adjacent frame. If copyrighted materials were deleted, a target note will appear listing the pages in the adjacent frame.

3. When a map, drawing or chart, etc., is part of the material being photographed, a definite method of “sectioning” the material has been followed. 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 illustrations that cannot be satisfactorily reproduced by xerographic means, photographic prints can be purchased at additional cost and inserted into your xerographic copy. These prints are available upon request from the Dissertations Customer Services Department.

5. Some pages in any document may have indistinct print. In all cases the best available copy has been filmed.
Bouktache, Essaid

ANALYSIS OF AN ADAPTIVE ANTENNA ARRAY WITH INTERMEDIATE-FREQUENCY WEIGHTING PARTIALLY IMPLEMENTED BY DIGITAL PROCESSING

The Ohio State University

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

In all cases this material has been filmed in the best possible way from the available copy. Problems encountered with this document have been identified here with a check mark √.

1. Glossy photographs or pages ______
2. Colored illustrations, paper or print ______
3. Photographs with dark background ______
4. Illustrations are poor copy ______
5. Pages with black marks, not original copy ______
6. Print shows through as there is text on both sides of page ______
7. Indistinct, broken or small print on several pages /√
8. Print exceeds margin requirements ______
9. Tightly bound copy with print lost in spine ______
10. Computer printout pages with indistinct print ______
11. Page(s) ______ lacking when material received, and not available from school or author.
12. Page(s) ______ seem to be missing in numbering only as text follows.
13. Two pages numbered ______. Text follows.
14. Curling and wrinkled pages ______
15. Dissertation contains pages with print at a slant, filmed as received ______
16. Other ____________________________

University
Microfilms
International
ANALYSIS OF AN ADAPTIVE ANTENNA ARRAY
WITH INTERMEDIATE-FREQUENCY WEIGHTING PARTIALLY IMPLEMENTED
BY DIGITAL PROCESSING

DISSERTATION

Presented in Partial Fulfillment of the Requirements for
the Degree Doctor of Philosophy in the Graduate
School of The Ohio State University

By

Essaid Bouktache, B.S.E.E., M.Sc.

* * * * *

The Ohio State University

1985

Reading Committee:
Professor D.T. Davis
Professor A.A. Ksienski
Professor F.D. Garber

Approved By

Adviser

Department of Electrical Engineering
DEDICATION

To the memory of my parents
ACKNOWLEDGMENTS

I would like to express my sincere appreciation to my graduate adviser, Professor D. T. Davis for his encouragements, guidance and suggestions throughout this work. I wish to thank Dr. W. G. Swarner to whom I owe a debt of gratitude for suggesting this topic, and for sharing his experience in this subject through fruitful discussions.
VITA

August 30, 1949
Born - Boghni(G-K), Algeria

1974
Engineer's Diploma (B.S.E.E),
Polytechnic School of Algiers, Algeria

1974-1977
Research Associate, CNI (Center of Computer
Science), Algeria

1980
M.Sc., The Ohio State University,
Columbus, Ohio

FIELDS OF STUDY

Major Field: Electrical Engineering

Studies in Communications: Professors D.T. Davis and C.E. Warren

Studies in Digital Systems: Professor F. Ozguner

Studies in Control Theory: Professor U. Ozguner

Studies in Mathematics: Professor S. Drobot
# TABLE OF CONTENTS

**DEDICATION** ii  
**ACKNOWLEDGMENTS** iii  
**VITA** iv  
**LIST OF FIGURES** vii  

## CHAPTERS  

<table>
<thead>
<tr>
<th>I</th>
<th>INTRODUCTION</th>
</tr>
</thead>
</table>
| 1.1 | HISTORICAL BACKGROUND | 1  
| 1.2 | PURPOSE OF THIS STUDY | 3  
| 1.3 | ORGANIZATION | 5  

| II | SIGNAL PROCESSING REQUIREMENTS FOR THE I.F. FEEDBACK LOOP | 7  
|---|------------------------------------------------|---|  
| 2.1 | INTRODUCTION | 7  
| 2.2 | THE LMS ALGORITHM | 7  
| 2.3 | SOME DEFICIENCIES ASSOCIATED WITH THE BASEBAND LOOP | 13  
| 2.4 | INTRODUCTION TO THE I.F. LOOP | 16  
| 2.5 | PARTIAL DIGITIZATION OF THE I.F. LOOP | 19  

| III | THE LMS ALGORITHM APPLIED TO THE I.F. ARRAY | 21  
|---|---------------------------------|---|  
| 3.1 | INTRODUCTION | 21  
| 3.2 | THE I.F. ALGORITHM | 22  
| 3.3 | TRANSIENT BEHAVIOR AND SPEED OF RESPONSE | 35  
| 3.4 | EXAMPLE OF A TWO-ELEMENT ARRAY | 40  
| 3.4.1 | Reference Signal with Both Sidebands | 43  
| 3.4.2 | Reference Signal with the Lower Sideband Only | 46  
| 3.4.3 | Reference Signal with the Upper Sideband Only | 51  

| IV | CONDITION FOR THE I.F. ARRAY PERFORMANCE | 58  
|---|--------------------------------------|---|  
| 4.1 | INTRODUCTION | 58  
| 4.2 | EXAMPLE OF A TWO-ELEMENT ARRAY | 60  
| 4.3 | ARRAY PERFORMANCE VERSUS LEAST-MEAN-SQUARE ERROR | 67  
| 1. | Reference signal with both sidebands | 68  
| 2. | Reference signal with the lower (or upper) sideband only | 70  

v
**LIST OF FIGURES**

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Block Diagram of an N-Element LMS Adaptive Array</td>
<td>8</td>
</tr>
<tr>
<td>2.2</td>
<td>Correlator Feedback Loop for the LMS Adaptive Array</td>
<td>12</td>
</tr>
<tr>
<td>2.3</td>
<td>Transconductance Baseband Weight Multiplier Gain versus Positive Control Voltage</td>
<td>15</td>
</tr>
<tr>
<td>2.4</td>
<td>Self-Excited I.F. Adaptive Array Feedback Controller</td>
<td>18</td>
</tr>
<tr>
<td>3.1</td>
<td>General Configuration for the LMS I.F. Adaptive Array</td>
<td>24</td>
</tr>
<tr>
<td>3.2</td>
<td>Basic Feedback Structure for the LMS Array with I.F. Weights</td>
<td>36</td>
</tr>
<tr>
<td>3.3</td>
<td>Scenario for a Two-Element Array Showing Input Signal Arrival</td>
<td>42</td>
</tr>
<tr>
<td>3.4</td>
<td>Inphase and Quadrature Components for a Two-Element Array: θ_d = 30°; Reference Signal with Upper and Lower Sidebands</td>
<td>47</td>
</tr>
<tr>
<td>3.5</td>
<td>Inphase and Quadrature Components for a Two-Element Array: θ_d = 60°; Reference Signal with Upper and Lower Sidebands</td>
<td>48</td>
</tr>
<tr>
<td>3.6</td>
<td>Inphase and Quadrature Components for a Two-Element Array: θ_d = 30°; Reference Signal with Lower Sideband</td>
<td>52</td>
</tr>
<tr>
<td>3.7</td>
<td>Inphase and Quadrature Components for a Two-Element Array: θ_d = 60°; Reference Signal with Lower Sideband</td>
<td>53</td>
</tr>
<tr>
<td>3.8</td>
<td>Inphase and Quadrature Components for a Two-Element Array: θ_d = 30°; Reference Signal with Upper Sideband</td>
<td>55</td>
</tr>
<tr>
<td>3.9</td>
<td>Inphase and Quadrature Components for a Two-Element Array: θ_d = 60° Reference Signal with Upper Sideband</td>
<td>56</td>
</tr>
<tr>
<td>4.1</td>
<td>A Two-Element LMS I.F. Array: Desired and Interference Signal Arrivals</td>
<td>61</td>
</tr>
<tr>
<td>4.2</td>
<td>Error Signal for a Two-Element LMS I.F. Array with no Output Filter</td>
<td>73</td>
</tr>
<tr>
<td>4.3</td>
<td>Error Signal for a Two-Element LMS I.F. Array with Output Filter</td>
<td>74</td>
</tr>
<tr>
<td>4.4</td>
<td>LMS I.F. Array Configuration with Output Band-Pass Filter</td>
<td>76</td>
</tr>
</tbody>
</table>
Figure

5.1 Theoretical Output Signal-to-Interference Ratio versus Upper Sideband Attenuation \( n \) for a Two-Element Array. 101

5.2 Inphase and Quadrature Components for a Simulated Two-Element LMS I.F. Array with Output Filter. 102

6.1 The LMS I.F. Array Loop Showing Group Delays \( t_1 \) and \( t_2 \). 105

6.2 Inphase and Quadrature Components of a Two-Element I.F. Array with no Group Delays; \( t_1 = 0, t_2 = 0 \). 110

6.3 Effect of Group Delay \( t_1 \) on the Inphase and Quadrature Components of a Two-Element I.F. Array; \( t_1 = 80 \mu s, t_2 = 0 \). 111

6.4 Effect of Group Delay \( t_2 \) on the Inphase and Quadrature Components of a Two-Element I.F. Array; \( t_1 = 0, t_2 = 50 \mu s \). 112

6.5 Effect of Group Delay \( t_1 \) on Output Signal-to-Interference Ratio for a Two-Element I.F. Array with \( t_2 = 0 \). 113

6.6 Effect of Group Delay \( t_2 \) on Output Signal-to-Interference Ratio for a Two-Element I.F. Array with \( t_1 = 0 \). 114

6.7 Characteristic Root \( D_1 \) for a Two-Element LMS I.F. Array versus Group Delays \( t_1 \) and \( t_2 \). 126

6.8 Characteristic Root \( D_1^* \) for a Two-Element LMS I.F. Array versus Group Delays \( t_1 \) and \( t_2 \). 127

6.9 Characteristic Root \( D_2 \) for a Two-Element LMS I.F. Array versus Group Delays \( t_1 \) and \( t_2 \). 128

6.10 Characteristic Root \( D_2^* \) for a Two-Element LMS I.F. Array versus Group Delays \( t_1 \) and \( t_2 \). 129

6.11 Characteristic Roots for a Two-Element LMS I.F. Array versus Loop Gain \( G \) with \( t_1 = t_2 = 0 \). 131

7.1 The LMS I.F. Array Loop Showing Time-Delay Compensators \( t_3 \) and \( t_4 \). 139

7.2 A Two-Element LMS I.F. Array: \( t_1 = t_2 = 0 \mu s \). 144

7.3 A Two-Element LMS I.F. Array: \( t_1 = 12.5 \mu s \). 146

7.4 A Two-Element LMS I.F. Array: \( t_2 = 0; t_3 = t_1 = 12.5 \mu s \). 147

7.5 A Two-Element LMS I.F. Array: \( t_3 = t_1 = 12.5 \mu s; t_4 = t_1 = 12.5 \mu s \). 148
7.6 A Two-Element LMS I.F. Array: \( t_3 = t_1 = 20 \mu s; \ t_1 = 2T_0 = 20\mu s \). 149

7.7 A Two-Element LMS I.F. Array: \( t_2 = 5 \mu s \). 151

7.8 A Two-Element LMS I.F. Array: \( t_2 = 5\mu s; \ t_4 = t_2 = 5\mu s \). 152

7.9 A Two-Element LMS I.F. Array: \( t_2 = T_0 = 10 \mu s \). 153

7.10 A Two-Element LMS I.F. Array: \( t_1 = 12.5 \mu s; \ t_2 = 8 \mu s \). 154

7.11 A Two-Element LMS I.F. Array: \( t_3 = t_1 = 12.5 \mu s; \ t_2 = 8 \mu s; \ t_4 = t_1+t_2 = 20.5 \mu s \). 155

7.12 A Two-Element LMS I.F. Array: \( t_3 = t_1 = 22 \mu s; \ t_2 = 8 \mu s; \ t_1+t_2 = 3T_0 = 30 \mu s \). 156

7.13 A Two-Element LMS I.F. Array: \( t_3 = t_1 = 12.5 \mu s; \ t_2 = 17.5 \mu s; \ t_1+t_2 = 3T_0 = 30 \mu s \). 157

7.14 Performance of a Two-Element LMS I.F. Array when the Stability Equations Are Applied for Several Values of Group Delays. 159

7.15 A Two-Element LMS I.F. Array Performance Sensitivity around Stable Points. 161

8.1 I.F.-Weighted Adaptive LMS Array Controller with Baseband Integrators. 163

8.2 Prefiltering and Frequency Down-Conversion Requirements. 167

8.3 General Principle of Vector Demodulation and Modulation by Bipolar Mixing. 172

8.4 Double Sampling with Sign Reversal. 175

8.5 Digital Vector Demodulator by Quadrature Sampling and Integration by Digital Accumulation. 176

8.6 Switching Logic as a Vector Modulator. 178

8.7 Timing Diagram for Demodulation and Modulation Control Waveforms. 179

8.8 A Two-Element LMS Array with Digitally Generated I.F. Weights; a) Weight for Loop 1 b) Weight for Loop 2 c) Error Signal. 182
1.1 HISTORICAL BACKGROUND

In recent years, communications systems employing signal processing antennas have been of significant interest. One of the specific types of antennas in this category is the adaptive array.

An adaptive antenna array consists of a set of receiving antenna elements followed by a signal processor which automatically controls the pattern of the antenna in a desired manner [1,2]. The signal from each element is multiplied by a controlled weight which adjusts the amplitude and phase of that signal. The weighted signals are then summed to form the array output.

The signal processor, which is of the feedback type, adjusts the antenna pattern to optimize desired signal reception. The pattern is automatically changed in such a way that maximum gain is provided in the incoming direction of a desired signal, and such that a pattern null is placed toward an undesired signal or interference. The spatial filtering property of adaptive antennas can be combined to great advantage with the temporal processing of spread-spectrum techniques [7], to provide even greater interference rejection.
Adaptive antennas have an increasing range of applications. Their special properties make them attractive for communication and radar applications where protection against accidental interference (radio frequency interference) or intentional interference (jamming) is desired.

Attention has been given recently to their application in satellite communication systems for up-link protection [11,20,21]. Satellite systems with "earth-coverage" beams are highly vulnerable to up-link interference. To reduce system vulnerability, the key is to employ a receiving array at the satellite, which automatically places pattern nulls in the direction of interfering signals and enhances the desired signal. More recently, the growing interest in adaptive arrays is being extended to include their application to TV broadcasting where suppression of interfering stations becomes a necessity.

The adaptive array concept has been the subject of extensive work for the past two decades. Various types of adaptive arrays have been proposed and investigated by several authors. The most common algorithms developed for weight generation led to three types of adaptive arrays: the array based on the LMS (Least Mean Square) algorithm [1], the Applebaum array [2], and the Shor array [3].

In the Applebaum array, the weights are adjusted in such a way that the output signal-to-noise ratio is maximized. Although this array is easier to implement, it requires that the angle of arrival of the desired signal be known at the receiver (which makes it suitable for radar applications, however). In the Shor array, the feedback control
is based on a steepest-ascent optimization of signal-to-noise ratio. However, because of its complicated implementation its investigation has been limited.

On the other hand, the LMS algorithm has been given more attention recently and is the one of concern in this study. The LMS algorithm is of interest because it does not require knowledge of the direction of arrival of the signal. It is based on the minimization of the mean-square error between the array output and a locally-generated signal called the reference signal [6]. The reference signal can be generated using some highly-structured functions, such as spread-spectrum signals [7,22], in such a way that it is correlated with the desired signal and uncorrelated with the interference.

1.2 PURPOSE OF THIS STUDY

The special attention given to the LMS algorithm has led to several publications related to its implementation [5,6,7,8,9,10,11,20,22]. Until recently, all implemented adaptive arrays used real weights operating at baseband. It has been shown [13] that the adaptive array with weights at baseband, which will be referred to as the baseband array (and its feedback structure as the baseband loop), suffers from severe performance deficiencies due to problems of DC offsets, feedthrough and leakage.

Since most of these problems are intrinsic to the baseband loop, it was suggested that the weighting operation be performed at an
intermediate frequency (I.F.) instead of at baseband. That is, the weighting elements will operate at a non-zero frequency. The array that results will be referred to as the I.F. array throughout this study, and its feedback structure as the I.F. loop.

Experiments conducted to implement this idea have shown that the problems affecting the baseband loop have been somewhat alleviated, and that the performance in interference rejection has been greatly improved [9,10]. Nevertheless, the ultimate goal was to be able to generate the I.F. weights by digital means which offers more versatility. It is the desire to achieve this digitization that is the objective of this study.

The major problem in dealing with this task, and which is the motivation for a thorough analysis of this subject, is the instability of the I.F. feedback loop. This stability problem associated with the I.F. loop results from the weights being at a non-zero frequency and the necessary filtering requirements.

Another constraint which contributes to the problem is the desire to accommodate broadband signals, which places stringent requirements on digital processing. In particular, this additional requirement is that a full bandwidth of 40 MHz be used (for a possible application in satellite communications).

Although the LMS array with baseband weights was originally formulated in discrete form [1], neither a digital implementation nor a thorough analysis of the I.F. array have been previously considered.

This investigation is concerned with the problems relevant to the instability of the I.F. array in order to facilitate the digital
implementation of the I.F. weight processor. One of the most difficult tasks is the stabilization of the feedback loops. Hence, this study also includes an in-depth analysis of the I.F. array, which is a necessary key to the stability problem. The differential equations of the weights, which are essential for the stability analysis are derived. Methods of achieving stability are given, and a digital I.F. weight processor is proposed. Extensive computer simulations which pertain to the different steps of this investigation are included as a means of verifying the analysis.

1.3 ORGANIZATION

Chapter II of this study introduces the I.F. loop as a means of improving upon the baseband loop. In Chapter III, the LMS algorithm is applied to the I.F. array, but without any constraint regarding performance in interference rejection. As a consequence, the differential equations derived in this chapter do no constitute the final formulation of the I.F. array.

Chapter IV gives a justification for the array to operate with one frequency sideband at its output, and hence places an additional constraint on the array structure given in Chapter III. In Chapter V, the differential equations of the inphase and quadrature components of the weights are derived for the I.F. array in the presence of the output band-pass filter. These equations, which are shown to be fundamentally different from those of Chapter III, are a necessary basis for the stability investigation. Their steady-state solution is derived and the
effect of the output filter on array performance is considered.

In Chapter VI, the effects of the weight processor filter and the array output filter on array stability and performance are shown both analytically and by computer simulation.

In Chapter VII, means to achieve stability are investigated. The stability conditions are derived and their effectiveness is verified by simulation. In Chapter VIII, a digital implementation for the I.F. weight processor is proposed and described. Finally, a summary and recommendations are given in Chapter IX.
CHAPTER II
SIGNAL PROCESSING REQUIREMENTS FOR THE I.F. FEEDBACK LOOP

2.1 INTRODUCTION

In this chapter, some hardware deficiencies associated with the implementation of the adaptive array with weights at baseband are presented, and the array with weights at an intermediate frequency is introduced.

First, a review of the well-known LMS algorithm [1, 5, 6, 7, 8] and the array structure that results from its implementation is given in Section 2.2. In Section 2.3, its main deficiencies that result in array performance degradation will be discussed. In Sections 2.4 and 2.5 the I.F. loop is introduced and the problem which serves as the focus of the present study is formulated.

2.2 THE LMS ALGORITHM

The LMS algorithm due to Widrow et al. [1], is based on the steepest descent minimization of the mean-square error which is the difference between the array output and a locally generated signal, called the reference signal in [5, 6].

The basic form of the LMS adaptive array with N antenna elements and scalar weights is shown in Figure 2.1. The signal from each element, \( y_i(t) \), is split into inphase and quadrature components. These
Figure 2.1 Block Diagram of an N-Element LMS Adaptive Array.
components are then weighted by means of a feedback circuitry which provides amplitude and phase control. The weighted signals, when summed, produce the array output $s_0(t)$. To make the array adaptive, $s_0(t)$ is subtracted from a reference signal $r(t)$ to form the error signal $e(t)$. By correlating the error signal with the signal from each element, the feedback system adjusts the weights in such a way that the mean-square error is minimized. As a result, the array output is forced to approximate the reference signal. Hence, if an interfering signal is present in the array input signal $y_i(t)$ but not in the reference signal $r(t)$, it appears in the error signal $e(t)$. Thus, the feedback control, by updating the weights, removes the interfering signal from the array output on a mean-square basis.

The derivation of the feedback controller based on the LMS algorithm is described in references [1,5,6]. However, for completeness, it will also be included here.

In Figure 2.1, if we let $x_j(t)$ represent either the inphase or the quadrature component, and $w_j(t)$ the corresponding weight, the array output is given by:

$$s_0(t) = \sum_{i=1}^{2N} w_ix_i(t) \quad (2.1)$$

The error signal is then

$$e(t) = r(t) - \sum_{i=1}^{2N} w_ix_i(t) \quad (2.2)$$

and the mean-square error is

9
\[
\overline{e^2(t)} = \overline{r^2(t)} - 2 \sum_{i=1}^{2N} w_i \overline{r(t)x_i(t)} + \sum_{i=1}^{2N} \sum_{j=1}^{2N} w_i w_j \overline{x_i(t)x_j(t)} \quad (2.3)
\]

where the overbar denotes time average.

Written in matrix form, Equation (2.3) becomes

\[
\overline{e^2(t)} = \overline{r^2(t)} - 2W^T S + W^T \Phi W \quad (2.4)
\]

where \(W\) is the baseband weight vector with \(2N\) components

\[
W = \begin{bmatrix}
    w_1 \\
    w_2 \\
    \vdots \\
    w_{2N}
\end{bmatrix} \quad (2.5)
\]

\(S\) is given by

\[
S = \begin{bmatrix}
    \overline{x_1(t)r(t)} \\
    \overline{x_2(t)r(t)} \\
    \vdots \\
    \overline{x_{2N(t)}r(t)}
\end{bmatrix} \quad (2.6)
\]

and \(\Phi\) is a \(2N \times 2N\) matrix

\[
\Phi = \begin{bmatrix}
    \overline{x_1(t)x_1(t)} & \cdots & \overline{x_1(t)x_{2N(t)}} \\
    \vdots & \ddots & \vdots \\
    \overline{x_{2N(t)}x_1(t)} & \cdots & \overline{x_{2N(t)}x_{2N(t)}}
\end{bmatrix} \quad (2.7)
\]
The mean-square error as given by Equation (2.3), being quadratic with respect to the weights, has a well-defined minimum. This minimum is approached by the steepest descent method [1] according to the rule:

\[
\frac{dw_i}{dt} = -k \nabla_{w_i}[e^2(t)]
\]  

(2.8)

where \( \nabla_{w_i} \) denotes the \( i \)th component of the gradient of \( e^2(t) \) with respect to the vector \( w \), and \( k \) a positive constant which controls the rate of convergence.

Evaluating the gradient in Equation (2.3) and using Equation (2.2) we obtain

\[
\nabla_{w_i}[e^2(t)] = -2x_i(t)e(t)
\]

(2.9)

and from Equation (2.8) the feedback rule becomes

\[
\frac{dw_i}{dt} = 2k x_i(t)e(t)
\]

(2.10)

The feedback structure which implements Equation (2.10) and which results in minimum mean-square error is given in Figure 2.2. This structure will be referred to as the baseband loop.

The steady-state solution under this algorithm is found by setting the gradient to zero. Using Equation (2.4) to evaluate the gradient yields

\[
\nabla_w[e^2(t)] = -2S + 2wW = 0
\]

(2.11)

hence, the optimal weight vector is
Figure 2.2 Correlator Feedback Loop for the LMS Adaptive Array.
\[ w_{\text{opt}} = \Phi^{-1} S \] (2.12)

which is the Wiener solution [1].

2.3 SOME DEFICIENCIES ASSOCIATED WITH THE BASEBAND LOOP

In this section a brief description of the problems that arise from an implementation of the baseband loop will be given. These problems are hardware-oriented and are primarily due to the non-ideal nature of physical devices.

The major problem encountered with the baseband loop is the problem of multiplier offset voltages and feedthrough. From experimental work on adaptive arrays with baseband weights [9], it has been found that the error-by-signal multiplier (lower mixer in Figure 2.2) is the most critical part of the array design, and has a strong impact on array performance.

The problem arises with DC voltages which appear at the output of these multipliers due to leakage effects, nonlinearities and circuit imbalance associated with active circuit devices.

If we denote by \( m(t) \) the output of the lower multiplier of Figure 2.2, then in a practical situation \( m(t) \) will contain terms such as

\[
m(t) = \delta_i + e(t)x_i(t) + c_1 x_i(t) + c_2 e(t) + c_3 x_i^2(t)
+ \ldots
\] (2.13)

Ideally, all that is necessary in \( m(t) \) is the term \( e(t)x_i(t) \) which is required by the LMS algorithm for optimal performance. The term \( \delta_i \),
which is called the multiplier offset voltage, is proper to the type of multiplier used. For an active transconductance multiplier, this offset is due to imbalance in the active circuits.

Another term that results in a DC component in Equation (2.13) is $x_1^2(t)$. This component, referred to as feedthrough, represents a leakage of the input signal to the other port of the mixer before it is mixed with itself. This offset, which is a function of the input signal power, would be hard to adjust in practice. The appropriate adjustment which can be achieved for a given signal power is incorrect for another input signal power. In a mixing operation (mainly with low-level signals), both terms can be responsible for a non-zero mixer output voltage for zero input voltage, as illustrated by the baseband mixer characteristic of Figure 2.3.

The effect of these DC components on weight behavior and array performance has been investigated in reference [13]. It has been found that the offset voltages affect the steady-state values of the weights, but not their time constants. The differential equation of the weights in the presence of offset voltages has been found to be

$$\frac{dW}{dt} + 2k \phi W = 2k[S+\Delta]$$  \hspace{1cm} (2.14)

where $\Delta$ represents the offset voltage vector [13].

Clearly, its steady-state solution is

$$W_{ss} = \phi^{-1}[S+\Delta]$$  \hspace{1cm} (2.15)

and differs from the optimum solution given by Equation (2.12).
Figure 2.3 Transconductance Baseband Weight Multiplier Gain versus Positive Control Voltage.
In addition to the problem of multiplier offset voltages, another problem, related to the presence of analog integrators in the controller loops should be mentioned. The problem here is that of capacitor leakage. The optimal weight value needs to be held constant at steady-state for proper array operation and performance. Any deviation from the optimal weight results in performance degradation. Furthermore, for low-level signals, leakage effects may result in complete array shut-off. Although the effect due to capacitor discharge was not critical in the actual experiments [9,10], it needs to be considered as a serious factor for design improvement.

Finally, pertinent to the array with weights at baseband, is the use of quadrature hybrid devices (Figure 2.1). Their quadrature properties may not be preserved, or may not be precise enough when operating on broadband signals.

2.4 INTRODUCTION TO THE I.F. LOOP

A method to overcome the feedthrough and the offset voltage problem is the use of control loops that operate at I.F. This idea, which has been introduced and partially investigated in [9], has been experimentally pursued in [10].

In such a loop, the weights are at a non-zero frequency (sinusoidal waveform) and carry both amplitude and phase information, hence eliminating the quadrature hybrid circuits used in the baseband loop. Thus, only one loop is needed per antenna element.
The weight signal, being at an I.F. frequency, say $f_0$, requires that the array input signal and error signal be at different frequencies. Therefore, the feedthrough problem is eliminated at the output of the error-by-signal multiplier (or correlator) since there is no baseband signal present in this loop.

It was initially thought that a band-pass filter would be the obvious solution for the I.F. weight generation. In the I.F. loop that results, shown in Figure 2.4, this filter is intended to be the counterpart of the baseband integrators.

However, the loop of Figure 2.4 assumes an ideal band-pass filter in order to replace the baseband integrators. The reason is that the output of such a filter needs to remain constant at steady-state, while its input, monitored by the error signal, assumes negligible values.

In the baseband loop, the output of the correlator is fed to an integrator. Consequently, even if the error signal drops to zero, the output of the integrator will be at its previous value thereby allowing the associated weight to have a steady-state that is not appreciably affected by the decrease of the error signal.

Hence, the baseband loop performs better at this point by virtue of the storage capability of the integrator. It is this storage capability that is lacking in the I.F. loop shown in Figure 2.4, and which constitutes the motivation behind this investigation.

In the actual experimental tests of the I.F. loop [9,10], the band-pass filter shown in Figure 2.4 is replaced by analog circuitry which comprises a vector demodulator that performs a reconversion to
Figure 2.4 Self-Excited I.F. Adaptive Array Feedback Controller.
baseband, two analog integrators for inphase and quadrature component generation, and a vector modulator.

The constructed array, which operates with I.F. weights at 410 MHz, has been shown to operate as desired when the analog circuits are properly aligned. However, some deficiencies still remain, due primarily to the reconversion to baseband which takes place at the vector demodulator; also, the problem of capacitor leakage is still a threat, and its effect on array performance is expected to be similar to that of the band-pass filter with inadequate quality factor.

2.5 PARTIAL DIGITIZATION OF THE I.F. LOOP

In order to cope with the initial lack of storage in the I.F. loop of Figure 2.4 and with the problems due to analog devices in the actual implementation, a digitization of the weight processor, with digital storage, is suggested.

The challenge in attempting this task is enhanced by two factors: 1) As an external constraint, it is desired to accommodate broadband signals. For the actual analog implementation [10], the aim was a 40 MHz bandwidth for the array input spectrum. This requirement which is extreme for easily available digital components, is the reason for the compromise of partial digitization with the correlator mixer and the weight multiplier remaining analog. Thus, with a bandwidth of 40 MHz for the input signal, the correlator output, which represents the error-by-signal multiplication, contains two frequency bands with a beamwidth of 80 MHz. Hence, the digital processor would have to
accommodate a wide bandwidth. 2) The I.F. loop is conditionally stable. The need to reduce the bandwidth at the correlator output is vital for proper digital processing. A band-pass filter at that location introduces undesirable poles which make the feedback loop unstable. As will be seen in the following sections, the presence of this filter, as well as the filter that is required at the array output, is a serious source of instability.
3.1 INTRODUCTION

In this chapter, the LMS algorithm due to Widrow [1] will be extended to the case where the array weights are no longer at baseband but rather at a nonzero frequency. In a method that parallels that of the baseband case, the optimal I.F. weight solution will be derived in Section 3.2. The array structure suitable for practical implementation will then follow the mathematical formulation of the weights. Section 3.3 considers the transient behavior. In Section 3.4, an example is treated of a two-element array and simulated and calculated results are compared.

We have seen how the LMS algorithm was applied to the case where the adaptive array weights are at baseband. The optimal weights were derived by minimizing the mean-square error between the weighted array output and the reference signal. More specifically, the method was based on the steepest descent minimization of the mean-square error with respect to each weight. This rule was expressed in Equation (2.8) and is again given here by Equation (3.1),

\[
\frac{dW_i}{dt} = -k\nabla_{w_i} [e^2(t)]
\]  

(3.1)
where $W_i(t)$ denotes the $i$th baseband weight, $\nabla_{W_i}[e^2(t)]$ the $i$th component of the gradient of $[e^2(t)]$ with respect to $W_i(t)$ and $k$ is a positive constant.

It has been shown that this algorithm leads to a first-order differential equation for the weights. Its steady-state solution represents the optimal weight vector, i.e. the set of array weights that yields minimum mean-square error.

3.2 THE I.F. ALGORITHM

We consider that we can write the I.F. weight, denoted $W_{IF}(t)$, as in Equation (3.2).

$$W_{IF}(t) = P(t)\cos\omega_0 t + Q(t)\sin\omega_0 t$$

(3.2)

$P(t)$ will be called the inphase component and $Q(t)$ the quadrature component.

We assume that $P(t)$ and $Q(t)$ are slowly varying compared to $\cos\omega_0 t$ or $\sin\omega_0 t$ where $\omega_0$ represents the intermediate radian frequency. In this manner $W_{IF}(t)$ can be seen as a narrow-band process with a carrier $\omega_0$ and with a slowly varying envelope given by $\sqrt{P^2(t)+Q^2(t)}$. Hence, the weight can also be written as

$$W_{IF}(t) = \sqrt{P^2(t)+Q^2(t)} \cdot \cos[\omega_0 t - \tan^{-1} \frac{Q(t)}{P(t)}]$$

(3.3)

At steady-state, both $P(t)$ and $Q(t)$ become DC constants, hence each weight becomes a sinusoidal waveform with constant amplitude and phase.
for given array input signal conditions (signal power, angle of arrival, etc).

However, in a practical situation $P(t)$ and $Q(t)$ will exhibit some fluctuations, even at steady-state due to thermal noise present in each antenna element and in the circuitry. Nevertheless, we assume that these fluctuations are small enough that $P(t)$ and $Q(t)$ can be treated as constants once they reach steady-state.

Let the I.F. weight corresponding to the $i^{th}$ element be denoted by $W_i(t)$ and the corresponding inphase and quadrature components by $P_i(t)$ and $Q_i(t)$ respectively. Thus,

$$W_i(t) = P_i(t)\cos\omega t + Q_i(t)\sin\omega t$$

Now we seek a feedback control law that will result in Equation (3.4) and such that the mean-square error between the array output and the reference signal is minimized.

Consider the $N$-element antenna array with the block diagram shown in Figure 3.1. The signals to the elements are denoted $X_i(t)$, $i = 1, 2, ..., N$. Each signal is multiplied by the weight $W_i(t)$, $i = 1, 2, ..., N$. $R(t)$ is the reference signal and $\epsilon(t)$ is the error signal in accordance with convention.

From Figure 3.1, the error signal is expressed as

$$\epsilon(t) = R(t) - \sum_{i=1}^{N} W_i(t)X_i(t)$$

(3.5)

where $W_i(t)$ and $X_i(t)$ are at different frequencies.

The error squared is given by
Figure 3.1 General Configuration for the LMS I.F. Adaptive Array.

\[ W_i = P_i \cos \omega_0 t + Q_i \sin \omega_0 t \]

\[ W_N = P_N \cos \omega_0 t + Q_N \sin \omega_0 t \]
\[ e^2(t) = R^2(t) - 2R(t) \sum_{i=1}^{N} W_i(t)X_i(t) + \sum_{i=1}^{N} \sum_{j=1}^{N} W_i(t)W_j(t)X_i(t)X_j(t) \]  
\hspace{1cm} (3.6)

Substituting the I.F. weight expression (3.4) into (3.6) yields

\[ e^2(t) = R^2(t) - 2R(t) \sum_{i=1}^{N} [P_i(t)\cos\omega_0 t + Q_i(t)\sin\omega_0 t]X_i(t) \]
\[ + \sum_{i=1}^{N} \sum_{j=1}^{N} [P_i(t)\cos\omega_0 t + Q_i(t)\sin\omega_0 t][P_j(t)\cos\omega_0 t + Q_j(t)\sin\omega_0 t] \]
\[ X_i(t)X_j(t) \]  
\hspace{1cm} (3.7)

The time average value of \( e^2(t) \), after expansion of Equation (3.7), is

\[ \overline{e^2(t)} = \overline{R^2(t)} - 2 \sum_{i=1}^{N} P_i(t) \overline{R(t)X_i(t)\cos\omega_0 t} - \]
\[ 2 \sum_{i=1}^{N} Q_i(t) \overline{R(t)X_i(t)\sin\omega_0 t} + \]
\[ \sum_{i=1}^{N} \sum_{j=1}^{N} P_i(t)P_j(t) \overline{X_i(t)X_j(t)\cos^2\omega_0 t} + \]
\[ \sum_{i=1}^{N} \sum_{j=1}^{N} Q_i(t)Q_j(t) \overline{X_i(t)X_j(t)\sin^2\omega_0 t} + \]
\[ \sum_{i=1}^{N} \sum_{j=1}^{N} P_i(t)Q_j(t) \overline{X_i(t)X_j(t)\cos\omega_0 t \sin\omega_0 t} + \]
\[ \sum_{i=1}^{N} \sum_{j=1}^{N} Q_i(t)P_j(t) \overline{X_i(t)X_j(t)\sin\omega_0 t \cos\omega_0 t} \]  
\hspace{1cm} (3.8)
Here, we assume that the variations of $P_i(t)$ and $Q_i(t)$ are relatively slow in comparison to the signals $X_i(t)$ and $R(t)$, so that their contribution to the average in each term of Equation (3.8) can be considered outside the overbar. Also, it should be understood that the above time-averages are taken over an interval short enough so that $P_i(t)$ and $Q_i(t)$ may be considered constant, but long compared to the periods of all the other signals.

Equation (3.8) may be written more compactly in matrix form. To this end, let $P$ and $Q$ denote the inphase and quadrature column vectors respectively, that is:

$$P = \begin{bmatrix} P_1(t) \\ P_2(t) \\ \vdots \\ P_N(t) \end{bmatrix} = \begin{bmatrix} P_1 \\ P_2 \\ \vdots \\ P_N \end{bmatrix} N \times 1 \quad (3.9)$$

$$Q = \begin{bmatrix} Q_1(t) \\ Q_2(t) \\ \vdots \\ Q_N(t) \end{bmatrix} = \begin{bmatrix} Q_1 \\ Q_2 \\ \vdots \\ Q_N \end{bmatrix} N \times 1 \quad (3.10)$$

where $N$ is the number of array elements. Define $S_p$ and $S_Q$ to be the column vectors:
\[ S_p = \begin{bmatrix} R(t)X_1(t)\cos\omega_0t \\ R(t)X_2(t)\cos\omega_0t \\ \vdots \\ R(t)X_N(t)\cos\omega_0t \end{bmatrix} = \begin{bmatrix} S_{p1} \\ \vdots \\ S_{pN} \end{bmatrix} N \times 1 \] (3.11)

\[ S_Q = \begin{bmatrix} R(t)X_1(t)\sin\omega_0t \\ R(t)X_2(t)\sin\omega_0t \\ \vdots \\ R(t)X_N(t)\sin\omega_0t \end{bmatrix} = \begin{bmatrix} S_{Q1} \\ \vdots \\ S_{QN} \end{bmatrix} N \times 1 \] (3.12)

And finally, define the \( N \times N \) matrices:

\[ \phi_p = \begin{bmatrix} x_1^2(t)X_1(t)\cos^2\omega_0t & \cdots & x_1(t)x_N(t)\cos^2\omega_0t \\ x_2^2(t)X_1(t)\cos^2\omega_0t & \cdots & \vdots \\ \vdots & \cdots & \vdots \\ x_N^2(t)X_1(t)\cos^2\omega_0t & \cdots & x_N(t)x_N(t)\cos^2\omega_0t \end{bmatrix} \] (3.13)
The last two terms in Equation (3.8) can be neglected since we have

\[
X_i(t)X_j(t)\cos\omega_d\sin\omega_l = \frac{1}{2} X_i(t)X_j(t)\sin 2\omega_l
\]  

(3.15)

The carrier radian frequency in the input signal \(X_i(t)\), say \(\omega_d\), can be assumed to be far enough from the weight frequency \(\omega_l\) so that the components present in (3.15) of frequencies \(2\omega_l\), \(2(\omega_l-\omega_d)\) and \(2(\omega_l+\omega_d)\) produce negligible contributions on the average. Thus we consider that

\[
X_i(t)X_j(t)\cos\omega_d\sin\omega_l = 0
\]  

(3.16)

Writing Equation (3.8) in matrix form, we have

\[
\bar{e}^2(t) = R^2(t) - 2P^T S p - 2Q^T S Q + P^T \Phi p P + Q^T \Phi Q Q
\]  

(3.17)

where \(T\) denotes transpose.

As we can see in Equation (3.17), the mean-square error \(\bar{e}^2(t)\) is similar in its form to the one derived in the case of the weights at
baseband [6], except that here, we have two extra terms due to the
inphase and quadrature components.

Also, as in the baseband case, $e^2(t)$ is a quadratic function of the
components $P_i$ and $Q_i$. Thus, given that $S_p$ and $S_Q$ are constant vectors
and that $\Phi_p$ and $\Phi_Q$ are constant matrices, the surface representing $e^2(t)$
as a function of either $P_i$ or $Q_i$ is a concave surface. Furthermore,
given the quadratic nature of $e^2(t)$, this surface has only one minimum
when plotted versus the inphase components $P_i$, and one minimum when
plotted versus the quadrature components $Q_i$.

Thus, if stationary statistics are assumed, the optimal estimates
of $P$ and $Q$ that result in minimum $e^2(t)$, which we denote $P_{\text{opt}}$ and $Q_{\text{opt}}$,
are found by setting the gradient of $e^2(t)$ with respect to $P$ and $Q$ equal
to zero; i.e.,

$$\nabla_P e^2(t) = 0 \quad (3.18)$$

and

$$\nabla_Q e^2(t) = 0 \quad (3.19)$$

where $\nabla_P$ and $\nabla_Q$ denote the gradient with respect to $P$ and $Q$
respectively. Applying (3.18) and (3.19) to (3.17) yields

$$\nabla_P e^2(t) = -2S_p + 2\Phi_p P_{\text{opt}} = 0 \quad (3.20)$$

$$\nabla_Q e^2(t) = -2S_Q + 2\Phi_Q Q_{\text{opt}} = 0 \quad (3.21)$$

Hence,

$$P_{\text{opt}} = \Phi_P^{-1} S_p \quad (3.22)$$

$$Q_{\text{opt}} = \Phi_Q^{-1} S_Q \quad (3.23)$$
therefore

\[ W_{\text{opt}} = P_{\text{opt}} \cos \omega t + Q_{\text{opt}} \sin \omega t \]  

(3.24)

where we assume that the matrices \( P \) and \( Q \) are nonsingular so that their inverses \( P^{-1} \) and \( Q^{-1} \) exist.

Equations (3.22) and (3.23) represent the optimal solution which, when substituted in (3.17) yields the minimum mean-square error

\[ e_{\text{min}}^2(t) = R(t) - S_p P^{-1} S_p - S_q Q^{-1} S_q \]  

(3.25)

The solutions given by (3.22) and (3.23) are the well known Wiener-Hopf equations [1] already mentioned in the baseband case.

As noted in the literature, the solution given by Equations (3.22) and (3.23) is not too attractive from a practical point-of-view. It is difficult and time consuming to evaluate the matrices \( P \) and \( Q \) and their inverses. Also, since the array has to adapt on a real time basis, it would be necessary to update such matrices whenever there is a change in the input signal conditions.

Thus, as was done in the baseband array, we use the more appealing method based on a steepest descent minimization of \( e^2(t) \) [1,6]. This rule, as already given by Equation (3.1), can be applied to the vectors \( P \) and \( Q \) to read:

\[ \frac{dP}{dt} = -k_P \left[ e^2(t) \right] \]  

(3.26)
\[
\frac{dQ}{dt} = -k \bar{Q}[e^2(t)] \tag{3.27}
\]

or with respect to the vector components as

\[
\frac{dP_i}{dt} = -k \frac{\partial \bar{Q}[e^2(t)]}{\partial P_i} \tag{3.28}
\]

\[
\frac{dQ_i}{dt} = -k \frac{\partial \bar{Q}[e^2(t)]}{\partial Q_i} \tag{3.29}
\]

where \( k \) is a positive constant that will affect the rate of convergence of \( P \) and \( Q \).

In (3.20) and (3.21) we had

\[
\nabla_p[e^2(t)] = -2S_p + 2\phi_p P \tag{3.30}
\]

\[
\nabla_q[e^2(t)] = -2S_q + 2\phi_q Q \tag{3.31}
\]

Substitution into (3.26) and (3.27) gives

\[
\frac{dP}{dt} = 2k[S_p - \phi_p P] \tag{3.32}
\]

\[
\frac{dQ}{dt} = 2k[S_q - \phi_q Q] \tag{3.33}
\]

We have seen that by definition we have

\[
S_p = \bar{R}(t)X\cos\omega t \tag{3.34}
\]

\[
S_Q = \bar{R}(t)X\sin\omega t \tag{3.35}
\]

\[
\phi_p = \frac{XX^T\cos^2\omega_0 t}{\omega_0 t} \tag{3.36}
\]
and

\( \phi_q = XX^T \sin^2 \omega_0 t \) \hspace{1cm} (3.37)

After substitution into (3.32) and (3.33) we have

\[
\frac{dP}{dt} = 2k [R(t)X \cos \omega_0 t - XX^T \cos^2 \omega_0 t \ P] \hspace{1cm} (3.38)
\]

and

\[
\frac{dQ}{dt} = 2k [R(t)X \sin \omega_0 t - XX^T \sin^2 \omega_0 t \ Q] \hspace{1cm} (3.39)
\]

Now, using the fact that

\[
XX^T P = (P^T X) X \hspace{1cm} (3.40)
\]

and

\[
XX^T Q = (Q^T X) X \hspace{1cm} (3.41)
\]

and again assuming \( P \) and \( Q \) are slowly varying so that they can be included under the overbar, we can modify (3.38) and (3.39) to become

\[
\frac{dP}{dt} = 2k [R(t)X \cos \omega_0 t - (P^T X) \cos^2 \omega_0 t X] \hspace{1cm} (3.42)
\]

\[
\frac{dQ}{dt} = 2k [R(t)X \sin \omega_0 t - (Q^T X) \sin^2 \omega_0 t X] \hspace{1cm} (3.43)
\]

or

\[
\frac{dP}{dt} = 2k \{ [R(t) - (P^T X) \cos \omega_0 t] \cos \omega_0 t \} \hspace{1cm} (3.44)
\]
One the other hand, the error signal is given by

\[ e(t) = R(t) - \sum_{i=1}^{N} (P_i \cos \omega_0 t + Q_i \sin \omega_0 t) X_i(t) \]  \hspace{1cm} (3.46)

or in matrix form by

\[ e(t) = R(t) - P^T X \cos \omega_0 t - Q^T X \sin \omega_0 t \]  \hspace{1cm} (3.47)

where \( P \), \( Q \) and \( X \) are column vectors already defined. From (3.47) it follows that

\[ R(t) - P^T X \cos \omega_0 t = e(t) + Q^T X \sin \omega_0 t \]  \hspace{1cm} (3.48)

and

\[ R(t) - Q^T X \sin \omega_0 t = e(t) + P^T X \cos \omega_0 t \]  \hspace{1cm} (3.49)

Using (3.48) and (3.49) in (3.44) and (3.45) respectively, yields

\[ \frac{dP}{dt} = 2k \left[ e(t) X \cos \omega_0 t + XX^T Q \sin \omega_0 t \cos \omega_0 t \right] \]  \hspace{1cm} (3.50)

\[ \frac{dQ}{dt} = 2k \left[ e(t) X \sin \omega_0 t + XX^T P \sin \omega_0 t \cos \omega_0 t \right] \]  \hspace{1cm} (3.51)

where use has been made of (3.40) and (3.41).
As seen in (3.16), we assume that

\[ XX^T \sin \omega_0 t \cos \omega_0 t = 0 \quad (3.52) \]

Without the above condition, the feedback rule that will result from implementing Equations (3.50) and (3.51) will be fairly complicated. However, the different frequencies can be chosen so that the time average in (3.52) has a negligible component at DC. Under this condition, (3.50) and (3.51) become

\[
\frac{dP}{dt} = 2k \frac{\varepsilon(t) X \cos \omega_0 t}{\omega_0} \quad (3.53)
\]

\[
\frac{dQ}{dt} = 2k \frac{\varepsilon(t) X \sin \omega_0 t}{\omega_0} \quad (3.54)
\]

or in integral forms

\[
P(t) = P(0) + 2k \int_0^t \varepsilon(t) X(t) \cos \omega_0 t \, dt \quad (3.55)
\]

\[
Q(t) = Q(0) + 2k \int_0^t \varepsilon(t) X(t) \sin \omega_0 t \, dt \quad (3.56)
\]

Equations (3.55) and (3.56) constitute the feedback rule by which the optimum inphase and quadrature components are generated in the case of an array with weights at a nonzero frequency.

With the I.F. weights given by

\[
W(t) = P(t) \cos \omega_0 t + Q(t) \sin \omega_0 t \quad (3.57)
\]

the implementation which follows from (3.55) and (3.56) is shown in Figure 3.2.
In a practical implementation of the array, the time-averages involved in Equations (3.55) and (3.56) are not included explicitly. However, the high frequency terms that result from the product under the integral will be averaged out by the integrators shown in Figure 3.2.

3.3 TRANSIENT BEHAVIOR AND SPEED OF RESPONSE

In this section we discuss briefly the general solution for the components $P(t)$ and $Q(t)$, which includes both transient and steady-state contribution. In a method that parallels that of the baseband case, the differential equations satisfied by the vectors $P$ and $Q$ will be given along with their eigenvalue solutions.

The above differential equations have already been established earlier. They are given by Equations (3.32) and (3.33). Denoting the loop gain by $G$, so that $2k = G$, (3.32) and (3.33) are rewritten as

$$\frac{dP}{dt} + G\varphi P = GS_p$$  \hspace{1cm} (3.58)

and

$$\frac{dQ}{dt} + G\varphi Q = GS_q$$  \hspace{1cm} (3.59)

where all the quantities involved have already been defined earlier by Equations (3.34) through (3.37).

The matrices $\varphi_P$ and $\varphi_Q$ were given by

$$\varphi_P = XX^T \cos^2 \omega_0 t$$  \hspace{1cm} (3.60)
Figure 3.2 Basic Feedback Structure for the LMS Array with I.F. Weights.
\[ \phi_Q = \overline{XX^T \sin^2 \omega_0 t} \]  
(3.61)

or

\[ \phi_P = \overline{XX^T (1/2 + 1/2 \cos^2 \omega_0 t)} \]  
(3.62)

and

\[ \phi_Q = \overline{XX^T (1/2 - 1/2 \cos^2 \omega_0 t)} \]  
(3.63)

Neglecting the double frequency term, it follows that

\[ \phi_P = \phi_Q = \frac{1}{2} \overline{XX^T} = \frac{1}{2} \phi \]  
(3.64)

where

\[ \phi = \overline{XX^T} \]  
(3.65)

represents the covariance matrix of the input vector.

The differential equations then become

\[ \frac{dP}{dt} + \frac{GP}{2} = GSP \]  
(3.66)

\[ \frac{dQ}{dt} + \frac{GQ}{2} = GSQ \]  
(3.67)

and their solutions are straightforward after a rotation of coordinates.

To this end define an \( N \times N \) matrix \( M \) such that

\[ P = MV_P \]  
(3.68)
and
\[ Q = MV_Q \] (3.69)

where \( V_P \) and \( V_Q \) are column vectors with \( N \) elements. Equations (3.66) and (3.67) can now be reduced to
\[
\frac{dV_P}{dt} + \frac{G}{2} (M^{-1} \phi M)V_P = GM^{-1}S_P \tag{3.70}
\]
\[
\frac{dV_Q}{dt} + \frac{G}{2} (M^{-1} \phi M)V_Q = GM^{-1}S_Q \tag{3.71}
\]

When \( M \) is chosen nonsingular and such that \( M^{-1} \phi M \) is diagonal, i.e.
\[
M^{-1} \phi M = \Lambda = \begin{bmatrix}
\lambda_1 & 0 \\
 & \lambda_2 \\
 & & \ddots \\
 & & & \lambda_N
\end{bmatrix} \tag{3.72}
\]

then the solutions to (3.70) and (3.71) are given by
\[
V_{Pi}(t) = 2 \frac{a_i}{\lambda_i} + [V_{Pi}(0) - 2 \frac{a_i}{\lambda_i}] e^{2 \lambda_i t}; \tag{3.73}
\]
\[ i = 1, \ldots, N \]
\[
V_{Qi}(t) = 2 \frac{b_i}{\lambda_i} + [V_{Qi}(0) - 2 \frac{b_i}{\lambda_i}] e^{2 \lambda_i t}; \tag{3.74}
\]
\[ i = 1, \ldots, N \]
where $V_{p_i}(0)$ and $V_{q_i}(0)$ denote the initial conditions, the $\lambda_i$'s the eigenvalues of $\phi = \frac{XX^T}{\lambda}$ and the $a_i$'s and $b_i$'s the $i$th components of the column vectors

$$M^{-1}Sp = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ \vdots \\ a_N \end{bmatrix} \quad (3.75)$$

$$M^{-1}S_Q = \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ \vdots \\ b_N \end{bmatrix} \quad (3.76)$$

The inphase and quadrature components are then calculated via (3.68) and (3.69).

The speed of response of the array is characterized by the time constants involved in the exponentials, i.e.

$$t_i = \frac{2}{G\lambda_i} \quad (3.77)$$
3.4 EXAMPLE OF A TWO-ELEMENT ARRAY

In this section, the previous results will be applied to the particular case of a two-element array. The purpose here is to test the validity of the expressions of the components P and Q via computer simulation. Therefore, for the sake of simplicity, we will use a single CW signal as input to the array, and ignore for the moment the interference rejection capability.

The input scenario used in the simulation is shown in Figure 3.3. The CW signal of radian frequency \( \omega_d \) and amplitude \( S \) is incident on the array at an angle \( \theta_d \). The elements are spaced one-half wavelength; therefore, the relative phase between elements is \( \gamma = \pi \sin \theta_d \). Hence, the input vector \( X(t) \) can be written as

\[
X(t) = \begin{bmatrix}
X_1(t) \\
X_2(t)
\end{bmatrix} = \begin{bmatrix}
S \cos(\omega_d t) \\
S \cos(\omega_d t - \gamma)
\end{bmatrix}
\]

(3.78)

with

\[\gamma = \pi \sin \theta_d \] (3.79)

Before we choose the reference signal, we note that since a mixing operation takes place at the weight multipliers shown in Figure 3.3, two frequency sidebands will be present at the array output, namely \( (\omega_0 - \omega_d) \) and \( (\omega_0 + \omega_d) \). Whether to select one or the other or both will be considered later. The same decision needs to be made for the reference signal at this point. Therefore, we consider three cases: first, we
use a reference signal which has both sidebands present in it; second, a reference signal with the lower sideband only; and finally, a reference signal with the upper sideband only.

For each case, the optimum steady-state solution is evaluated using Equations (3.22) and (3.23), namely

\[ P_{\text{opt}} = \frac{-1}{\phi_p} S_p \]

(3.80)

and

\[ Q_{\text{opt}} = \frac{-1}{\phi_q} S_q \]

(3.81)

with \( \phi_p \) and \( \phi_q \) already given by

\[ \phi_p = \phi_q = 1/2 \phi \]

(3.82)

and

\[ \phi = \overline{XX^T} \]

(3.83)

Also, \( S_p \) and \( S_q \) were defined previously by

\[ S_p = R(t)X(t)\cos\omega t \]

(3.84)

\[ S_q = R(t)X(t)\sin\omega t \]

(3.85)

where \( X(t) \) is the input vector and \( R(t) \) the reference signal.

Then, for each case a computer simulation is carried out using the feedback structure shown in Figure 3.2. For each loop, the components \( P \) and \( Q \) are plotted versus time and the steady-state values obtained compared to the expected theoretical values.
Specifications: $f_d = 6 \text{ KHz} (\omega_d = 2\pi f_d)$; $f_0 = 10 \text{ KHz} (\omega_0 = 2\pi f_0)$; $S = 100$; sampling period $T = 10^{-7} \text{ sec}$; loop gain $G = 25$.

Figure 3.3 Scenario for a Two-Element Array Showing Input Signal Arrival.
3.4.1 Reference Signal with Both Sidebands

Denoting the amplitude by $R$, the reference signal with two sidebands can be written as

$$R(t) = R \cos(\omega_0 - \omega_d)t + R \cos(\omega_0 + \omega_d)t$$

(3.86)

Evaluating the $\phi$ matrix, we have:

$$\phi = \begin{bmatrix} x_1^2 & x_1 x_2 \\ x_1 x_2 & x_2^2 \\ x_2 x_1 & x_2^2 \\ x_2 x_1 & x_2^2 \end{bmatrix}$$

(3.87)

whose inverse is

$$\phi^{-1} = \frac{2}{s^2 \sin^2 \gamma} \begin{bmatrix} 1 & -\cos \gamma \\ -\cos \gamma & 1 \end{bmatrix}$$

(3.88)
The column vectors \( S_p \) and \( S_q \) are evaluated according to Equations (3.84) and (3.85)

\[
S_p = \begin{bmatrix}
R(t)\cos \omega t \cos \omega t \\
R(t)\cos (\omega t - \gamma) \cos \omega t
\end{bmatrix} \quad (3.89)
\]

\[
S_q = \begin{bmatrix}
R(t)\cos \omega t \sin \omega t \\
R(t)\cos (\omega t - \gamma) \sin \omega t
\end{bmatrix} \quad (3.90)
\]

Substituting \( R(t) \) by its expression (3.86) and carrying the time-average operation gives

\[
S_p = \begin{bmatrix}
\frac{SR}{2} \\
\frac{SR}{2} \cos \gamma
\end{bmatrix} \quad (3.91)
\]

\[
S_q = \begin{bmatrix}
0 \\
0
\end{bmatrix} \quad (3.92)
\]

From (3.80), (3.81) and (3.82) the optimal steady-state solution is

\[
P_{\text{opt}} = 2\phi^{-1}S_p = \begin{bmatrix}
P_1 \\
P_2
\end{bmatrix} = \begin{bmatrix}
\frac{2R}{S} \\
0
\end{bmatrix} \quad (3.93)
\]
Thus,

\[ W_1(t) = \frac{2R}{S} \cos \omega_0 t \quad (3.95) \]

\[ W_2(t) = 0 \quad (3.96) \]

which is the result expected by inspection of Figure 3.3. Clearly, the array matches its output to the reference signal since

\[ S(t) = W_1 X_1 + W_2 X_2 = \left( \frac{2R}{S} \cos \omega_0 t \right) S \cos \omega_d t \]

\[ = R \cos (\omega_0 - \omega_d) t + R \cos (\omega_0 + \omega_d) t = R(t) \quad (3.97) \]

and hence

\[ \varepsilon(t) = 0 \quad (3.98) \]

The computer simulation is carried out with \( S = R = 100 \). Thus, the steady-state solution is expected to be

\[ P_{opt} = \begin{bmatrix} P_1 \\ P_2 \end{bmatrix} = \begin{bmatrix} 2 \\ 0 \end{bmatrix} \quad (3.99) \]

\[ Q_{opt} = \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (3.100) \]
Figures 3.4 and 3.5 show the computer plots of $P_{\text{opt}}$ and $Q_{\text{opt}}$ for two different values of the angle of arrival $\theta_d$, namely $30^\circ$ and $60^\circ$ respectively. As we can see, the steady-state values agree quite well with the theoretical results given by (3.99) and (3.100).

3.4.2 Reference Signal with the Lower Sideband Only

With the lower sideband, the reference signal is written as

$$R(t) = R \cos(\omega_0 - \omega_d)t$$

(3.101)

Using (3.101) in (3.89) and in (3.90) yields

$$Sp = \begin{bmatrix} SR/4 \\ SR/4 \cos \gamma \\ SR/4 \sin \gamma \end{bmatrix}$$

(3.102)

$$S_Q = \begin{bmatrix} 0 \\ SR/4 \cos \gamma \\ SR/4 \sin \gamma \end{bmatrix}$$

(3.103)

The steady-state solution is thus

$$P_{\text{opt}} = 2\gamma^{-1}Sp = \begin{bmatrix} P_1 \\ P_2 \end{bmatrix} = \begin{bmatrix} R/S \\ 0 \end{bmatrix}$$

(3.104)
Figure 3.4 Inphase and Quadrature Components for a Two-Element Array.

\[ \theta_d = 30^\circ \]

Reference Signal: \( R(t) = R\cos(\omega_0 - \omega_d)t + R\cos(\omega_0 + \omega_d)t \).
Figure 3.5 Inphase and Quadrature Components for a Two-Element Array. 
\[ \theta_d = 60^\circ \]
Reference Signal: \[ R(t) = R \cos(\omega_0 - \omega_d) + R \cos(\omega_0 + \omega_d) t. \]
\[ Q_{\text{opt}} = 2\hat{\psi}^{-1}SQ = \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix} = \begin{bmatrix} -\frac{R \cos \gamma}{S \sin \gamma} \\ \frac{R}{S \sin \gamma} \end{bmatrix} \]  

(3.105)

Using these results to evaluate the array output \( S(t) \) we have:

\[
S(t) = W_1(t)X_1(t) + W_2(t)X_2(t)
\]

\[
= \left[ \frac{R}{S \cos \omega_0 t} - \frac{R \cos \gamma}{S \sin \gamma \sin \omega_0 t} \right] \cos \omega_d t
\]

\[
+ \left[ \frac{R}{S \sin \gamma \sin \omega_0 t} \right] \cos (\omega_d t - \gamma)
\]

(3.106)

Developing further, yields

\[
S(t) = \frac{R}{2} \cos(\omega_0 - \omega_d)t - \frac{R \cos \gamma}{2 S \sin \gamma} \sin(\omega_0 - \omega_d)t
\]

\[
+ \frac{R}{2 S \sin \gamma} [\sin(\omega_0 - \omega_d)t \cdot \cos \gamma + \cos(\omega_0 - \omega_d)t \cdot \sin \gamma]
\]

\[
+ \frac{R}{2} \cos(\omega_0 + \omega_d)t - \frac{R \cos \gamma}{2 S \sin \gamma} \sin(\omega_0 + \omega_d)t
\]

\[
+ \frac{R}{2 S \sin \gamma} [\sin(\omega_0 + \omega_d)t \cdot \cos \gamma - \cos(\omega_0 + \omega_d)t \cdot \sin \gamma]
\]

\[
= R \cos(\omega_0 - \omega_d)t
\]

Hence,

\[
S(t) = R \cos(\omega_0 - \omega_d)t = R(t)
\]

(3.108)

Again, the array feedback processor matches the array output to the reference signal even though only the lower sideband is used in the latter.

49
To illustrate the above steady-state solution, a computer simulation was done using Equation (3.101) as the reference signal. The simulation was done for two different values of \( \theta_d \); \( \theta_d = 30^\circ \) and \( \theta_d = 60^\circ \). For each case, the vectors \( P \) and \( Q \) are plotted versus time.

From Equations (3.104) and (3.105) with \( R = S = 100 \), the steady-state solution is:

1. For \( \theta_d = 30^\circ \), for which \( y = \pi \sin \theta_d = \pi/2 \) and hence \( \cos y = 0 \) and \( \sin y = 1 \), we have:

\[
\begin{align*}
P_{opt} &= \begin{bmatrix} -P_1 \\ P_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\
Q_{opt} &= \begin{bmatrix} -Q_1 \\ Q_2 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}
\end{align*}
\]

(3.109) (3.110)

2. For \( \theta_d = 60^\circ \), \( y = \frac{\pi \sqrt{3}}{2} \) and

\[
\begin{align*}
P_{opt} &= \begin{bmatrix} -P_1 \\ P_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \\
Q_{opt} &= \begin{bmatrix} -Q_1 \\ Q_2 \end{bmatrix} = \begin{bmatrix} -1 \\ -\sqrt{3} \tan \frac{\pi}{2} \\ \frac{1}{\sqrt{3}} \sin \frac{\pi}{2} \end{bmatrix} = \begin{bmatrix} 2.2339 \\ 2.4475 \end{bmatrix}
\end{align*}
\]

(3.111) (3.112)
On the other hand, the vectors $P_{opt}$ and $Q_{opt}$ that result from the computer simulation are shown plotted in Figures 3.6 and 3.7, corresponding to $\theta_d = 30^\circ$ and $\theta_d = 60^\circ$ respectively. For each case, the simulated steady-states come close to the calculated ones.

### 3.4.3 Reference Signal with the Upper Sideband Only

With the upper sideband only, the reference signal is written as

$$R(t) = R\cos(\omega_0 + \omega_d)t$$  \hspace{1cm} (3.113)

Carrying the previous calculations, the results are:

$$S_p = \begin{bmatrix} \frac{SR}{4} \\ 0 \\ \frac{SR}{4\cos\gamma} \end{bmatrix}$$  \hspace{1cm} (3.114)

$$S_Q = \begin{bmatrix} 0 \\ \frac{SR}{-4\sin\gamma} \end{bmatrix}$$  \hspace{1cm} (3.115)

$$P_{opt} = 2\phi^{-1}S_p = \begin{bmatrix} P_1 \\ P_2 \end{bmatrix} = \begin{bmatrix} R/S \\ 0 \end{bmatrix}$$  \hspace{1cm} (3.116)

$$Q_{opt} = 2\phi^{-1}S_Q = \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix} = \begin{bmatrix} -\frac{R \cos\gamma}{S \sin\gamma} \\ -\frac{R}{S \sin\gamma} \end{bmatrix}$$  \hspace{1cm} (3.117)
Figure 3.6 Inphase and Quadrature Components for a Two-Element Array. 

\[ \theta_d = 30^\circ \]

Reference Signal: \[ R(t) = R \cos(\omega_0 - \omega_d) t . \]
Figure 3.7 Inphase and Quadrature Components for a Two-Element Array.
\[ \theta_d = 60^\circ \]
Reference Signal: \[ R(t) = R \cos(\omega_d - \omega_d)t. \]
Evaluating the array output from these results we have

\[
S(t) = R \cos \omega_0 t + \frac{R \cos \gamma}{\text{S}} \cos \omega_0 t \cdot \text{S} \cos \omega_d t + \frac{R}{-\text{S} \sin \gamma \sin \omega_0 t} \cos(\omega_d t - \gamma)
\]

(3.118)

After expansion we get

\[
S(t) = R \cos(\omega_0 + \omega_d) t = R(t)
\]

(3.119)

and the array output is again matched to the reference signal.

The numerical solutions are in this case:

with \( \theta_d = 30^\circ \):

\[
P_{\text{opt}} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad \quad \text{(3.120)}
\]

\[
Q_{\text{opt}} = \begin{bmatrix} 0 \\ -1 \end{bmatrix} \quad \quad \text{(3.121)}
\]

and with \( \theta_d = 60^\circ \):

\[
P_{\text{opt}} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad \quad \text{(3.122)}
\]

\[
Q_{\text{opt}} = \begin{bmatrix} -2.2339 \\ -2.4475 \end{bmatrix} \quad \quad \text{(3.123)}
\]

The simulation plots for the above cases are shown in Figures 3.8 and 3.9 respectively; again, the steady-state values shown in the plots agree quite well with the theoretical ones.
Figure 3.8 Inphase and Quadrature Components for a Two-Element Array.

\[ \theta_d = 30^\circ \]

Reference Signal: \[ R(t) = R \cos(\omega_0 + \omega_d)t. \]
Figure 3.9 Inphase and Quadrature Components for a Two-Element Array.

\[ \theta_d = 60^\circ \]
Reference Signal: \( R(t) = R cos(\omega t + \phi) t. \)
Thus, in this chapter the LMS algorithm was applied to the case where the weights are at a non-zero frequency. The differential equations satisfied by the inphase vector P and the quadrature vector Q were derived. An example of a two-element array operating under interference-free conditions was presented. Computer simulation results for that particular case were shown to agree with the expected theoretical values.
4.1 INTRODUCTION

In this chapter, the interference rejection capabilities of the basic array structure shown in Figure 3.2 will be investigated. A condition for optimal performance will be given. Computer simulations will be added to verify the validity of this investigation.

The motivation for this portion of the study comes from the results obtained in the example of the two-element array treated in Section 3.4 of Chapter III. In that section, where interference-free conditions were considered, the structure of Figure 3.2 allowed the array output to be matched to the reference signal resulting in a zero error signal (assuming no thermal noise). In addition, in deriving the theoretical results, that is, the equations which resulted in the above structure, no conditions were imposed on the array input signal vector $X(t)$. In a real situation, $X(t)$ contains a desired signal, one or more interfering signals (or jammers) and thermal noise. Therefore, we desire to investigate the case where interference is present at the array input using the structure of Figure 3.2.

Furthermore, by inspection of Figure 3.2, we see that at the output of the weight multiplier (top mixer) in each loop, both the lower and
the upper sidebands are present, that is, the difference-frequency term and the sum-frequency term. It is the presence of this upper frequency sideband that is of concern here.

Under the interference-free condition of Chapter III, we saw that the array output and the reference signal were the same at steady-state, even though both frequency sidebands were present at the output of each weight multiplier. In that particular case, the weights were such that when the weighted product from loop one was added to the weighted product from loop two, the upper sideband present in loop one cancels the one present in loop two by virtue of the amplitude and phase correction provided by the weights. Therefore, we want to determine if this property still holds, at least approximately, in the case where both the desired signal and the interfering signal are present in the array input vector X(t).

In the following, Section 4.2 considers the example of a two-element array with interference added, using the array structure of Figure 3.2. Two general cases will be considered: in case one, both frequency sidebands are retained at the array output; in case two, a single sideband is retained. For each case, two reference signal conditions will be considered. For the first case, the reference signal will have both sidebands first, then the lower sideband only; for the second case, first the array output and the reference signal will be taken with the lower sideband only, then with the upper one only. Without loss of generality, this example will be found sufficient to derive the condition for optimal array performance.
Section 4.3 gives a justification for the above cases by consideration of the mean-square error which corresponds to each case. To illustrate the findings in both sections, computer simulations were conducted to measure the interference rejection capability and to show the error signal under each condition.

4.2 EXAMPLE OF A TWO-ELEMENT ARRAY

For the sake of illustration, we consider again the example of a two-element antenna array, the configuration of which is shown in Figure 4.1.

In this example, we consider that a desired signal $X_d(t)$ is incident on the array from broadside, and an interference signal $S_I(t)$ is incident at an angle $\theta_I$ from broadside. Assume that the desired signal and the interference signal are both CW signals given respectively by $S \cos \omega_d t$ and $I \cos \omega_I t$, with $\omega_I$ very close to $\omega_d$.

First, we write the array output by inspection of Figure 4.1. Then we try to solve for the weight components $P_i$ and $Q_i$ ($i=1,2$) by forcing the array output $Y(t)$ to be equal to the reference signal $R(t)$. This, of course, depicts a steady-state situation.

Thus, referring to Figure 4.1, the input signal to element one is

$$X_1(t) = S \cos \omega_d t + I \cos \omega_I t$$

(4.1)

and the input to element two is

$$X_2(t) = S \cos \omega_d t + I \cos(\omega_I t - \gamma_I)$$

(4.2)
Figure 4.1 A Two-Element LMS I.F. Array: Desired and Interference Signal Arrivals.
where
\[ Y_1 = \pi \sin \theta_1 \]  \hspace{1cm} (4.3)
represents the inter-element phase shift associated with the interference signal.

The weights are written in terms of the inphase and quadrature components and hence are given by
\[ W_1(t) = P_1 \cos \omega_0 t + Q_1 \sin \omega_0 t \]  \hspace{1cm} (4.4)
\[ W_2(t) = P_2 \cos \omega_0 t + Q_2 \sin \omega_0 t \]  \hspace{1cm} (4.5)

The array output is thus
\[ y(t) = W_1(t)X_1(t) + W_2(t)X_2(t) \]  \hspace{1cm} (4.6)

Substituting the different signals by their respective expression and carrying the expansions in order to show the sum and difference frequency terms, yields:
\[ y(t) = \frac{S}{2}(P_1 + P_2)\cos(\omega_0 - \omega_d) t + \frac{S}{2}(Q_1 + Q_2)\sin(\omega_0 - \omega_d) t \]
\[ + \frac{1}{2}(P_1 + P_2)\cos(\omega_0 + \omega_d) t + \frac{1}{2}(Q_1 + Q_2)\sin(\omega_0 + \omega_d) t \]
\[ + \frac{1}{2}(P_1 + P_2)\cos \gamma_1 + Q_2 \sin \gamma_1)\cos(\omega_0 - \omega_1) t \]
\[ + \frac{1}{2}(Q_1 - P_2 \sin \gamma_1 + Q_2 \cos \gamma_1)\sin(\omega_0 - \omega_1) t \]
\[ + \frac{1}{2}(P_1 + P_2 \cos \gamma_1 - Q_2 \sin \gamma_1)\cos(\omega_0 + \omega_1) t \]
\[ + \frac{1}{2}(Q_1 + P_2 \sin \gamma_1 + Q_2 \cos \gamma_1)\sin(\omega_0 + \omega_1) t \]  \hspace{1cm} (4.7)
We now consider the different cases mentioned earlier.

Case I: Retaining Both the Lower and the Upper Sideband at the Output

1.a. Reference signal with both sidebands

For this case, assume that the reference signal is given by

\[ R(t) = R \cos(\omega_0 - \omega_d) t + R \cos(\omega_0 + \omega_d) t \]  \hspace{1cm} (4.8)

In an ideal situation, the array output will contain the desired signal only. Thus, forcing the output \( y(t) \) given by (4.7) to be equal to the reference signal \( R(t) \) given by (4.8), results in the following:

a) Retaining the desired signal yields

\[ S \left( P_1 + P_2 \right) = R \]  \hspace{1cm} (4.9)
\[ S \left( Q_1 + Q_2 \right) = 0 \]  \hspace{1cm} (4.10)

b) Nulling the interference by setting to zero the corresponding coefficients in Equation (4.7) yields

\[ I \left( P_1 + P_2 \cos \gamma_1 + Q_2 \sin \gamma_1 \right) = 0 \]  \hspace{1cm} (4.11)
\[ I \left( Q_1 - P_2 \sin \gamma_1 + Q_2 \cos \gamma_1 \right) = 0 \]  \hspace{1cm} (4.12)
\[ I \left( P_1 + P_2 \cos \gamma_1 - Q_2 \sin \gamma_1 \right) = 0 \]  \hspace{1cm} (4.13)
\[ I \left( Q_1 + P_2 \sin \gamma_1 + Q_2 \cos \gamma_1 \right) = 0 \]  \hspace{1cm} (4.14)
As we can see, the above system with four unknowns, \(P_1\), \(P_2\), \(Q_1\) and \(Q_2\), is overconstrained. The solution to the system of equations given by (4.11) through (4.14) is the trivial solution

\[
P_1 = P_2 = Q_1 = Q_2 = 0
\]  

(4.15)
a solution which is not acceptable and also which is in contradiction with condition (4.9).

Hence, we see that in this case, it is not possible to retain the desired signal and null the interference signal completely at the same time.

1.b. Reference signal with the lower sideband only

Assuming that the reference signal is given by the difference frequency term

\[
R(t) = R \cos(\omega_0 - \omega_d)t
\]

(4.16)
then, by inspection of the array output expression given by (4.7), it would not be possible to null out the sum frequency term \((\omega_0 + \omega_d)\) which came from the desired signal. As a result, the array output cannot be matched to the reference signal which is given by (4.16). This is due to the fact that the lower and the upper sideband terms in the desired signal, namely, \(\cos(\omega_0 - \omega_d)t\) and \(\cos(\omega_0 + \omega_d)t\), have equal amplitudes in Equation (4.7) as a result of the desired signal arriving from broadside. Therefore, the necessary condition to retain the desired signal happens to retain both the lower and the upper sideband.
The previous results would suggest to consider the other case where either frequency sideband is retained at the array output and the other removed.

Case 2: Retaining a Single Sideband at the Output

2.a. Retaining the lower frequency sideband only

In this case, we consider only the difference frequency terms in Equation (4.7), i.e., \((\omega_0 - \omega_d)\) for the desired signal and \((\omega_0 - \omega_I)\) for the interference. The reference signal has only the term with the difference frequency \((\omega_0 - \omega_d)\), as given by (4.16).

Retaining the desired signal in \(y(t)\) gives

\[
\frac{S}{2}(P_1 + P_2) = R \tag{4.17}
\]

\[
\frac{S}{2}(Q_1 + Q_2) = 0 \tag{4.18}
\]

Nulling the interference, yields

\[
P_1 + P_2\cos\gamma_1 + Q_2\sin\gamma_1 = 0 \tag{4.19}
\]

\[
Q_1 - P_2\sin\gamma_1 + Q_2\cos\gamma_1 = 0 \tag{4.20}
\]

combining (4.17) through (4.20) and solving for \(P_1, P_2, Q_1\) and \(Q_2\) yields

\[
P_1 = P_2 = \frac{R}{S} \tag{4.21}
\]

\[
Q_1 = -Q_2 = \frac{1 + \cos\gamma_1}{\sin\gamma_1} \tag{4.22}
\]
Hence, a steady-state solution exists for this case which simultaneously nulls the interference and makes the array output match the reference signal.

2.b. Retaining the upper frequency sideband only

This subcase is identical to the previous one. Thus, considering the sum frequency terms in Equation (4.7), i.e., \((\omega_0 + \omega_d)\) and \((\omega_0 + \omega_I)\) and assuming that the reference signal is given by

\[
R(t) = R \cos(\omega_0 t + \omega_d t) \tag{4.23}
\]

We find by carrying out similar calculations that

\[
P_1 = P_2 = R/S \tag{4.24}
\]

and

\[
Q_1 = -Q_2 = -R/S \frac{1 + \cos{\gamma_I}}{\sin{\gamma_I}} \tag{4.25}
\]

Thus, we conclude from the previous results that in order for the array to reject the interference and retain the desired signal simultaneously, one of the frequency sidebands resulting from the weighing operation should be removed by filtering.

One should note that in the above calculations the goal was to force the interference at the array output to be nulled completely, a condition that should result in a zero mean-square error. Such a condition is too strong for the LMS algorithm from a practical viewpoint. Rather, the LMS algorithm operates with a certain tolerance in a real situation. As a result, the mean-square error is never zero.
in a practical sense. What the LMS algorithm guarantees is that for a
given scenario at the array input and for a given reference signal, the
mean-square error in question is minimized.

Hence, the fact that the mean-square error is minimized should also
hold whether we retain one or both frequency sidebands at the array
output. However, the array performance in interference rejection should
be greatly affected by this choice. What makes this difference is the
value of the least-mean-square error itself. For instance, choosing the
lower sideband only will result in a certain LMS error; choosing both
sidebands will result in a different LMS error. The smaller of the two
will lead to the better array performance.

The next section illustrates partially this idea, with attention
given to the LMSE (Least-Mean-Square Error) and to the error signal.

4.3 ARRAY PERFORMANCE VERSUS LEAST-MEAN-SQUARE ERROR

Thus, an appropriate way to illustrate the previous results would
be to compare the LMSE in both cases. For case 1, the LMSE expression
was given in Chapter III by Equation (3.25). However, for case 2, which
depicts an array with the output filter, the weights obey a differential
equation which is different from that of case 1, as it will be seen in
the next chapter. The resulting LMSE expression is mathematically
involved and therefore will not be given.

Instead, we will present the results of a computer simulation
showing the error signal, the value of the associated LMSE and the
corresponding signal-to-interference ratio for both cases.
Before presenting the simulation results, we will, however, write the LMSE expressions for the first case only (that is, no output filter) by considering two different subcases according to whether we take the reference signal with one or both frequency sidebands.

1. Reference signal with both sidebands

The general expression for the LMSE was given in Chapter III for the case of no output filter and is repeated here in Equation (4.26).

\[ \frac{e_{\text{min}}^2}{R(t)} = R^2(t) - S_p^T \phi_p^{-1} S_p - S_q^T \phi_q^{-1} S_q \]

(4.26)

where \( R(t) \) represents the reference signal, the vectors \( S_p \) and \( S_q \) were defined by (3.11) and (3.12) respectively, and the matrices \( \phi_p \) and \( \phi_q \) were defined by (3.13) and (3.14).

In the CW situation, the reference signal was written as

\[ R(t) = R \cos(\omega_0 - \omega_d) t + R \cos(\omega_0 + \omega_d) t \]

(4.27)

In a more general way, let the reference signal be given by

\[ R(t) = R_L(t) + R_U(t) \]

(4.28)

where \( R_L(t) \) and \( R_U(t) \) represent the lower and the upper frequency sideband respectively.

On the other hand, the elements of the vectors \( S_p \) and \( S_q \) were given by

\[ R(t) X_i(t) \cos \omega_0 t \quad ; \quad i = 1, \ldots, N \]

(4.29)

for \( S_p \) and
\[ R(t)X_i(t) \sin \omega t \quad i=1, \ldots, N \quad (4.30) \]

for \( S_q \), \( \omega_0 \) being the radian frequency of the weight signal.

Substituting (4.28) into (4.29) and (4.30) will result in the vectors \( S_p \) and \( S_q \) being written as

\[ S_p = [S_p]_L + [S_p]_U \quad (4.31) \]

and

\[ S_q = [S_q]_L + [S_q]_U \quad (4.32) \]

where the terms with subscript \( L \) are due to \( R_L(t) \) and the terms with subscript \( U \) are due to \( R_U(t) \).

Theoretically, \( R_L(t) \) and \( R_U(t) \) contribute equally to the time-average in Equations (4.29) and (4.30). As a result it follows that

\[ S_p = 2[S_p]_L = 2[S_p]_U \quad (4.33) \]

and

\[ S_q = 2[S_q]_L = 2[S_q]_U \quad (4.34) \]

Using (4.28) to evaluate \( \overline{R^2(t)} \), we have

\[ \overline{R^2(t)} = \overline{[R_L(t) + R_U(t)]^2} = \overline{R^2_L(t)} + \overline{R^2_U(t)} \quad (4.35) \]

and on the average we have

\[ \overline{R^2(t)} = 2 \overline{R^2_L(t)} = 2 \overline{R^2_U(t)} \quad (4.36) \]
Using the previous results in Equation (4.26), we get

\[
\bar{\epsilon}_{\min}^2 \leq 2R^2_L(t) - 4[S_p]^T_L \cdot [\phi_p]^{-1} \cdot [S_p]_L \nonumber
\]

\[
- 4[S_q]^T_L \cdot [\phi_q]^{-1} \cdot [S_q]_L
\]

which is the LMS when both sidebands are present at the array output and in the reference signal. This expression is identical when written in terms of the upper sideband only.

We now consider the other subcase where the reference signal contains the lower (or the upper) sideband only.

2. **Reference signal with the lower (or upper) sideband only**

Let \( R(t) \) contain, for instance, just the lower sideband, i.e.,

\[
R(t) = R_L(t)
\]  

(4.38)

so that

\[
\bar{R}^2(t) = \bar{R}_L^2(t)
\]  

(4.39)

then, each of the vectors \( S_p \) and \( S_q \) given by (4.31) and (4.32) will now contain the lower term only. Hence,

\[
S_p = [S_p]_L
\]  

(4.40)

\[
S_q = [S_q]_L
\]  

(4.41)

Using the above results in the general expression of the mean-square error, we have:
which is the L M S E sought for subcase 2. An identical expression, with subscript U instead of L, would result if \( R(t) = R_U(t) \).

From (4.37) and (4.42) one establishes that

\[
[e_{\min}^2]_2 = \frac{1}{4} [e_{\min}^2]_1 + \frac{1}{2} R_L^2
\]  

(4.43)

and as long as

\[
[e_{\min}^2]_1 < \frac{2}{3} R_L^2
\]  

(4.44)

we have

\[
[e_{\min}^2]_2 > [e_{\min}^2]_1
\]  

(4.45)

as it will be confirmed by the simulation.

The computer simulation was performed using the two-element array example of Section 4.2 with the input scenario shown in Figure 4.1. The parameters used for the simulation are as follows:

- Desired signal arrival: broadside
- Interference signal arrival: \( \theta_I = 60^\circ \) from broadside
- Input signal-to-interference ratio: 0 dB
- Desired signal amplitude: \( S = 10 \)
- Interference signal amplitude: \( I = 10 \)
- Reference signal amplitude: \( R = 10 \)
- Frequency of incoming desired signal: \( f_d = 6 \) KHz
- Frequency of incoming interference signal: $f_I = 6.4$ KHz
- Frequency of weight signal: $f_0 = 10$ KHz
- Loop gain: $G = 10$
- Sampling period: $T = 1 \mu$s
- Running time: 27 ms

For each case considered previously, the error signal is plotted versus time and the value of the LMSE and the output signal-to-interference ratio (SIR) computed.

The plots of the error signal are shown in Figure 4.2 for case one and in Figure 4.3 for case two, with the corresponding values of LMSE and SIR.

As we can see in Figure 4.3, which corresponds to the array operating with a single sideband, the error signal is shown to decrease toward zero at steady-state. The array is performing quite successfully with an output signal-to-interference ratio of about 50 dB, and equally well regardless of which frequency sideband is selected.

In Figure 4.2, which shows the case of the array operating with both the lower and the upper sideband, the error signal does not decrease further even at steady-state and, as expected, the performance is degraded: the output signal-to-interference ratio is only about 13 dB when the reference signal has both sidebands (Figure 4.2a), and only about 8 dB when it has only the lower sideband (Figure 4.2b).

The least-mean-square errors (LMSE) computed are also shown in the above figures. For case 1, we see that these values are in agreement with Equations (4.43) and (4.45) since with
a) $R(t) = R\cos(\omega_0 - \omega_d)t + R\cos(\omega_0 + \omega_d)t$
LMSE = 7.27
SIR = 13.601 dB

b) $R(t) = R\cos(\omega_0 - \omega_d)t$
LMSE = 26.5
SIR = 8.460 dB

Figure 4.2 Error Signal for a Two-Element LMS I.F. Array with no Output Filter.
a) \( R(t) = R \cos(\omega_0 - \omega_d)t \)
Upper sideband attenuation = 51 dB
LMSE = 4.26
SIR = 50.336 dB

b) \( R(t) = R \cos(\omega_0 + \omega_d)t \)
Lower sideband attenuation = 51 dB
LMSE = 4.29
SIR = 50.089 dB

Figure 4.3 Error Signal for a Two-Element LMS I.F. Array with Output Filter.
\[ R_e^2 = \frac{R_e^2}{2} = 50 \]  
\[ e_{\text{min}}^2 = 7.27 \]  
\[ \frac{e_{\text{min}}^2}{2} = \frac{1}{4} (7.27) + \frac{1}{2} (50) = 26.82 \]

which agrees with the result of Figure 4.2b (LMSE = 26.5).

For case 2 (Figure 4.3), the LMSE value is much smaller and will theoretically go to zero at steady-state for this example with deterministic signals.

Thus, in this chapter we have seen that in order for the array to null an interfering signal and retain a desired signal in an optimal manner, it is necessary to eliminate one of the frequency sidebands that resulted from the weighting operation. The corresponding array structure is given in Figure 4.4 which shows the necessary output band-pass filter.

The main concern of this portion of the study was to determine if the array can operate acceptably without this filter. From experimental results and as we will show in the following chapters, the presence of this filter in the feedback paths introduces an undesirable time delay that is a source of array instability.
Figure 4.4 LMS I.F. Array Configuration with Output Band-Pass Filter.
CHAPTER V
EFFECTS OF UPPER SIDEBAND ATTENUATION ON I.F. ARRAY WEIGHTS

5.1 INTRODUCTION

The purpose of this chapter is the derivation of the differential equations for the I.F. array weights (inphase and quadrature components) in the presence of the output band-pass filter required to remove one of the sidebands resulting from the weight mixer. We will see that the presence of this filter leads to an optimal weight solution that is fundamentally different from the one derived in Chapter III where such output filtering was not considered. The analysis of this chapter will provide a useful basis for subsequent work which deals with the stability problem.

Starting in Section 5.2, the LMS algorithm will be applied to the I.F. array, taking into account the effect of the output filter which follows the weight mixer. The differential equations for the inphase and quadrature components will be derived and the dependence on the upper sideband attenuation coefficient will be shown. In Section 5.3, the differential equations will be derived directly from the array structure shown in Figure 4.4 of the previous chapter. Section 5.4 gives the general optimal steady-state solution that results. Finally, in Section 5.5, the results of the previous section will be used to show
the array interference rejection as a function of the filter attenuation coefficient by considering an example of a two-element array.

5.2 DIFFERENTIAL EQUATIONS FROM THE LMS ALGORITHM

In this section, the LMS algorithm described in Chapter III will be applied again, but this time the outputs of the weighted array elements are fed to a band-pass filter following the summing operation. We will consider that the sum-frequency terms are the ones to be removed from the array output. Hence, the filter performance will be characterized by an upper sideband attenuation coefficient \( n \). The results will be presented as a function of this parameter.

First we will derive the expression of the mean-square error to which the gradient technique can be applied.

Once the array elements are weighted and summed, the summing junction output is given in matrix form by

\[
y(t) = W^T X \tag{5.1}
\]

where the weight vector is written as

\[
W = P \cos \omega_0 t + Q \sin \omega_0 t \tag{5.2}
\]

and \( X \) is the array input signal vector. Also, it is understood that \( P \) and \( Q \), representing the inphase and quadrature vectors respectively, are functions of time during adaptation.

Equation (5.1) can be written as a sum of a lower sideband which represents the difference-frequency terms and an upper sideband which
represents the sum-frequency terms, i.e.

\[ y(t) = (W^T X)_L + (W^T X)_U \]  

(5.3)

The presence of the band-pass filter can now be introduced in Equation (5.3) with the introduction of the upper sideband attenuation \( n \). The useful array output, which we denote \( y_F(t) \), is written accordingly as

\[ y_F(t) = (W^T X)_L + \frac{1}{n} (W^T X)_U \]  

(5.4)

where the subscripts \( L \) and \( U \) denote the lower and upper sidebands respectively.

Application of the above output expression yields the error signal in the form

\[ e(t) = R(t) - y_F(t) \]

\[ = R(t) - (W^T X)_L - \frac{1}{n} (W^T X)_U \]  

(5.5)

where \( R(t) \) is the reference signal.

We will assume all along that the central radian frequency of the reference signal is given by \( (\omega_0 - \omega_d) \), where \( \omega_0 \) is the radian frequency of the weights and \( \omega_d \) the central radian frequency of the desired input signal.

Squaring Equation (5.5) and taking the time-average, results in the mean-square error expression
\[ e^2(t) = R^2(t) + (W^T X)^2_L + \frac{1}{n^2} (W^T X)^2_U - 2R(t)(W^T X)_L \]

\[ - \frac{2}{n} R(t)(W^T X)_U + \frac{2}{n} (W^T X)_L (W^T X)_U \]  

(Equation 5.6)

The last two terms in Equation (5.6) can be neglected because of the zero time-average of the product of the difference-frequency components by the sum-frequency components, that is, \( R(t) \) and \( (W^T X)_L \) are at \( (\omega_0 - \omega_1) \) whereas \( (W^T X)_U \) is at \( (\omega_0 + \omega_1) \).

Substituting the expression for \( W \) into Equation (5.6) and rearranging terms yields

\[ e^2(t) = R^2(t) + [P^T(X \cos \omega_0)_{L}]^2 + [Q^T(X \sin \omega_0)_{L}]^2 \]

\[ + 2[P^T(X \cos \omega_0)_{L}][Q^T(X \sin \omega_0)_{L}] \]

\[ + \frac{1}{n^2} [P^T(X \cos \omega_0)_{U}]^2 + \frac{1}{n^2} [Q^T(X \sin \omega_0)_{U}]^2 \]

\[ + \frac{2}{n^2} [P^T(X \cos \omega_0)_{U}][Q^T(X \sin \omega_0)_{U}] \]

\[- 2R(t)[P^T(X \cos \omega_0)_{L}] - 2R(t)[Q^T(X \sin \omega_0)_{L}] \]

(Equation 5.7)

We now introduce some matrix notations in order to write Equation (5.7) in a more compact and useful form. We will make use of the following matrix identity:
\[(p^TY)(Q^TZ) = (p^TY)(Z^TQ) = p^T(YZ^T)Q = p^TAQ\]  \hspace{1cm} (5.8)

where \(P\), \(Q\), \(Y\) and \(Z\) are column matrices of dimension \(N\) and \(A\) an \(N \times N\) matrix defined by

\[A = YZ^T\]  \hspace{1cm} (5.9)

Application of Equation (5.8) to the different terms of Equation (5.7) and use of the usual assumption that \(P\) and \(Q\) are slowly varying with time compared to the other signals, we get:

\[\left[p^T(X\cos\omega_0t)_L\right]^2 = p^T[(X\cos\omega_0t)_L(X^T\cos\omega_0t)_L]P = p^TA_1P \]  \hspace{1cm} (5.10)

\[\left[q^T(X\sin\omega_0t)_L\right]^2 = q^T[(X\sin\omega_0t)_L(X^T\sin\omega_0t)_L]Q = q^TA_2Q \]  \hspace{1cm} (5.11)

\[\left[p^T(X\cos\omega_0t)_U\right]^2 = p^T[(X\cos\omega_0t)_U(X^T\cos\omega_0t)_U]P = p^TA_3P \]  \hspace{1cm} (5.12)

\[\left[q^T(X\sin\omega_0t)_U\right]^2 = q^T[(X\sin\omega_0t)_U(X^T\sin\omega_0t)_U]Q = q^TA_4Q \]  \hspace{1cm} (5.13)

\[\left[p^T(X\cos\omega_0t)_L][q^T(X\sin\omega_0t)_L\right]
\[\left[p^T(X\cos\omega_0t)_U][q^T(X\sin\omega_0t)_U\right] \]

\[= p^T[(X\cos\omega_0t)_L(X^T\sin\omega_0t)_L]Q = p^TA_5Q \]  \hspace{1cm} (5.14)

and finally

\[\left[p^T(X\cos\omega_0t)_U][q^T(X\sin\omega_0t)_U\right] = \]

\[p^T[(X\cos\omega_0t)_U(X^T\sin\omega_0t)_U]Q = p^TA_6Q \]  \hspace{1cm} (5.15)

81
We can simplify the above expressions further as shown below. For instance, consider the matrix $XX^T \cos^2 \omega_0 t$ which can be written as

$$XX^T \cos^2 \omega_0 t = (X \cos \omega_0 t)(X^T \cos \omega_0 t) \quad (5.16)$$

With the introduction of the lower and upper sideband notation, we have

$$X \cos \omega_0 t = (X \cos \omega_0 t)_L + (X \cos \omega_0 t)_U \quad (5.17)$$

After substitution of this expression into Equation (5.16) and cancellation of cross-product terms we obtain

$$XX^T \cos^2 \omega_0 t = (X \cos \omega_0 t)_L (X^T \cos \omega_0 t)_L + (X \cos \omega_0 t)_U (X^T \cos \omega_0 t)_U \quad (5.18)$$

Since the input vector $X$ is generally a bandlimited process, it can easily be shown that the lower and upper sidebands contribute equally on the average, and as a result, the two terms in Equations (5.18) are equal. Hence, we have

$$XX^T \cos^2 \omega_0 t = 2(X \cos \omega_0 t)_L (X^T \cos \omega_0 t)_L = 2(X \cos \omega_0 t)_U (X^T \cos \omega_0 t)_U \quad (5.19)$$

Similarly, the matrix $XX^T \sin^2 \omega_0 t$ can be written as

$$XX^T \sin^2 \omega_0 t = (X \sin \omega_0 t)(X^T \sin \omega_0 t) \quad (5.20)$$

and further as
\[ XX^T \sin^2 \omega_0 t = 2(X \sin \omega_0 t)^L (X \sin \omega_0 t)^L = 2(X \sin \omega_0 t)_U (X \sin \omega_0 t)_U \]  

(5.21)

Now since

\[ XX^T \cos^2 \omega_0 t = XX^T \sin^2 \omega_0 t = \frac{1}{2} \ XX^T = \Phi_1 \]  

(5.22)

we conclude that

\[ A_1 = A_2 = A_3 = A_4 = \frac{1}{2} \ \Phi_1 \]  

(5.23)

where by definition

\[ \Phi_1 = \frac{1}{2} \ XX^T \]  

(5.24)

and we recall that \( XX^T \) represents the covariance matrix of the array input signals.

For the matrices \( A_5 \) and \( A_6 \) introduced in Equations (5.14) and (5.15), we elaborate as follows. We have

\[ XX^T \sin \omega_0 t \cos \omega_0 t = 0 \]  

(5.25)

on the average.

The above can be written as

\[ XX^T \sin \omega_0 t \cos \omega_0 t = \frac{(X \cos \omega_0 t) (X \sin \omega_0 t)}{0} \]  

(5.26)

or in terms of the lower and upper sidebands as

\[ (X \cos \omega_0 t)_L (X \sin \omega_0 t)_L + (X \cos \omega_0 t)_U (X \sin \omega_0 t)_U = 0 \]  

(5.27)
Therefore we have

\[(X\cos\omega_0 t)_L(X^T\sin\omega_0 t)_L = -(X\cos\omega_0 t)_U(X^T\sin\omega_0 t)_U\]  \hspace{1cm} (5.28)

and hence

\[A_5 = -A_6 = \phi_{pq}\]  \hspace{1cm} (5.29)

where by definition we set

\[\phi_{pq} = (X\cos\omega_0 t)_L(X^T\sin\omega_0 t)_L\]  \hspace{1cm} (5.30)

In Equation (5.7) we also have by definition

\[R(t)(X\cos\omega_0 t)_L = R(t)X\cos\omega_0 t = S_p\]  \hspace{1cm} (5.31)

and

\[R(t)(X\sin\omega_0 t)_L = R(t)X\sin\omega_0 t = S_q\]  \hspace{1cm} (5.32)

Application of the preceding notation and substitution into Equation (5.7) permits the expression for the mean-square error to be written in a more compact form as follows:

\[\varepsilon^2(t) = R(t) - 2p^TS_p - 2q^TS_q + 1/2 (1+1/\pi^2)p^Tq_{\Phi}p\]
\[+ 1/2 (1+1/\pi^2)q^T\Phi_s\Phi_{pq}q + 2(1-1/\pi^2)p^Tq_{\Phi}q\]  \hspace{1cm} (5.33)

where we recall that the matrices \(\Phi_s\), \(\Phi_{pq}\), \(S_p\) and \(S_q\) are defined by Equations (5.22), (5.30), (5.31) and (5.32) respectively.
Equation (5.33) is the desired expression for the mean-square error. It is still quadratic with respect to \( P \) or \( Q \), a condition that is necessary for the steepest descent minimization.

We now apply the LMS algorithm to the above expression. The procedure was already outlined in Chapter III. That is, the rate of change of the components \( P \) and \( Q \) with time is set equal to the negative of the gradient of the mean-square error with respect to each one of these components. Thus, this law requires that

\[
\frac{dP}{dt} = -k \nabla_P [e^2] \tag{5.34}
\]

and

\[
\frac{dQ}{dt} = -k \nabla_Q [e^2] \tag{5.35}
\]

where \( k \) is the usual positive constant which controls the rate of convergence.

The matrix operations necessary to evaluate the gradients are the following:

Let \( U \) and \( V \) be vectors of dimension \( N \) and \( A \) a square matrix of order \( N \), then

\[
\frac{\partial (U^T V)}{\partial U} = V \tag{5.36}
\]

\[
\frac{\partial (U^T A U)}{\partial U} = AU + A^T U \tag{5.37}
\]

\[
\frac{\partial (U^T A V)}{\partial U} = AV \tag{5.38}
\]
\[
\frac{\partial (U^T A V)}{\partial V} = A^T U 
\] (5.39)

Applying the above formulas to Equation (5.33) and using the fact that
\[
\phi_T^T \phi = \phi_I 
\] (5.40)
(which is due to the input covariance matrix being symmetrical), the
gradient with respect to \( P \) is given by
\[
\nabla_P[e^2] = \frac{\partial [e^2]}{\partial P} = -2P + \left( 1 + \frac{1}{\eta^2} \right) \phi^I P + 2\left( 1 - \frac{1}{\eta^2} \right) \phi_{pq} Q 
\] (5.41)
and the gradient with respect to \( Q \) by
\[
\nabla_Q[e^2] = \frac{\partial [e^2]}{\partial Q} = -2Q + \left( 1 + \frac{1}{\eta^2} \right) \phi^I Q + 2\left( 1 - \frac{1}{\eta^2} \right) \phi_{pq}^T P 
\] (5.42)

Finally, combining the above with Equations (5.34) and (5.35) and
setting \( 2k = G \), results in the differential equations sought:
\[
\frac{dP}{dt} + \frac{G}{2} \left( 1 + \frac{1}{\eta^2} \right) \phi^I P + G \left( 1 - \frac{1}{\eta^2} \right) \phi_{pq} Q = G S_P 
\] (5.43)
and
\[
\frac{dQ}{dt} + \frac{G}{2} \left( 1 + \frac{1}{\eta^2} \right) \phi^I Q + G \left( 1 - \frac{1}{\eta^2} \right) \phi_{pq}^T P = G S_Q 
\] (5.44)

As we can see, Equations (5.43) and (5.44) are coupled differential
equations in \( P \) and \( Q \). Their steady-state solutions will be the optimal
vectors \( P \) and \( Q \) which will result in the least-mean-square error for a
given upper sideband attenuation coefficient \( \eta \).
One will be tempted to seek a feedback processor that will be described by the above differential equations. Instead of doing so directly, we recall first the feedback array structure that was given in Chapter IV. Such a structure, shown in Figure 4.4, was supposed to lead to optimal array performance. We will thus write the differential equations starting with that given configuration and compare them to the ones resulting from the direct application of the LMS algorithm of this section. This analysis is the subject of the next section.

5.3 Differential Equations from the Already Known Feedback Structure

The procedure here is to derive the expression for the derivatives of P and Q by inspection of the feedback loop depicted in Figure 4.4 of the previous chapter. We saw that this structure included the output filter which we characterize by the same upper sideband attenuation factor $\eta$.

Thus, from Figure 4.4, we have:

$$\frac{dP}{dt} = G \varepsilon(t)X\cos\omega t$$  \hspace{1cm} (5.45)

$$\frac{dQ}{dt} = G \varepsilon(t)X\sin\omega t$$  \hspace{1cm} (5.46)

As seen before, the error signal is given by

$$\varepsilon(t) = R(t) - (WTX)_L - \frac{1}{\eta} (WTX)_U = R(t) - PT(X\cos\omega t)_L$$

$$- QT(X\sin\omega t)_L - \frac{1}{\eta} PT(X\cos\omega t)_U - \frac{1}{\eta} QT(X\sin\omega t)_U$$  \hspace{1cm} (5.47)
Substitution into Equations (5.45) and (5.46) yields

\[ \begin{align*}
\frac{dp}{dt} &= G[R(t)X\cos\omega_0 t - p^T(X\cos\omega_0 t)_L(X\cos\omega_0 t) \\
&\quad - \frac{1}{n} \frac{Q^T(X\sin\omega_0 t)_L(X\cos\omega_0 t) - \frac{1}{n} p^T(X\cos\omega_0 t)_U(X\cos\omega_0 t)}{\frac{1}{n} Q^T(X\sin\omega_0 t)_U(X\cos\omega_0 t)} \\
&= G[R(t)X\cos\omega_0 t - \frac{Q}{n} (X\sin\omega_0 t)_L(X\cos\omega_0 t)] (5.48)
\end{align*} \]

\[ \begin{align*}
\frac{dq}{dt} &= G[R(t)X\sin\omega_0 t - \frac{p}{n} (X\cos\omega_0 t)_L(X\sin\omega_0 t) \\
&\quad - \frac{1}{n} \frac{Q^T(X\sin\omega_0 t)_L(X\sin\omega_0 t) - \frac{1}{n} p^T(X\cos\omega_0 t)_U(X\sin\omega_0 t)}{\frac{1}{n} Q^T(X\sin\omega_0 t)_U(X\sin\omega_0 t)} \\
&= G[R(t)X\sin\omega_0 t - \frac{Q}{n} (X\sin\omega_0 t)_L(X\sin\omega_0 t)] (5.49)
\end{align*} \]

By making use of the matrix identity

\[ (P^T X) Y = (X^T P) Y = (Y X^T) P = AP \] (5.50)

where \( P, X \) and \( Y \) are column matrices of equal dimension and \( A \) the associated square matrix, Equations (5.48) and (5.49) become

\[ \begin{align*}
\frac{dp}{dt} &= G[R(t)X\cos\omega_0 t - [(X\cos\omega_0 t)(X^T\cos\omega_0 t)_L]P \\
&\quad - [(X\cos\omega_0 t)(X^T\sin\omega_0 t)_L]Q - \frac{1}{n} [(X\cos\omega_0 t)(X^T\cos\omega_0 t)_U]P \\
&\quad - \frac{1}{n} [(X\cos\omega_0 t)(X^T\sin\omega_0 t)_U]Q] (5.51)
\end{align*} \]
\[
\frac{d\mathbf{Q}}{dt} = G(R(t)x\sin\omega_0 t) - \left[(x\sin\omega_0 t)(x^T\cos\omega_0 t)_L\right]^P
\]
\[
- \left[(x\sin\omega_0 t)(x^T\sin\omega_0 t)_L\right]^Q - \frac{1}{n} \left[(x\sin\omega_0 t)(x^T\cos\omega_0 t)_U\right]^P
\]
\[
- \frac{1}{n} \left[(x\sin\omega_0 t)(x^T\sin\omega_0 t)_U\right]^Q
\]  

(5.52)

In the above expressions, we recognize matrices \(A_1\) through \(A_6\) which were defined in the previous section except for \(A_5\) and \(A_6\). Each of these appear in Equation (5.52) as its own transpose; that is,

\[
(x\sin\omega_0 t)(x^T\cos\omega_0 t)_L = (x\sin\omega_0 t)_L(x^T\cos\omega_0 t)_L = A_5^T
\]  

(5.53)

and

\[
(x\sin\omega_0 t)(x^T\cos\omega_0 t)_U = (x\sin\omega_0 t)_U(x^T\cos\omega_0 t)_U = A_6^T
\]  

(5.54)

where \(T\) denotes transpose.

Now we recall that

\[
A_1 = A_2 = A_3 = A_4 = \frac{1}{2}\Phi_1
\]  

(5.55)

with

\[
\Phi_1 = XX^T\cos^2\omega_0 t = XX^T\sin^2\omega_0 t = \frac{1}{2}XX^T
\]  

(5.56)

and

\[
A_5 = -A_6 = \Phi_{pq} = (x\cos\omega_0 t)_L(x^T\sin\omega_0 t)_L
\]  

(5.57)
Substitution into Equations (5.51) and (5.52) yields the final forms of the desired differential equations:

\[
\frac{dP}{dt} + \frac{G}{2} (1 + \frac{1}{n})G_P + G(1 - \frac{1}{n})G_{pq} = GS_p
\]  

(5.58)

\[
\frac{dQ}{dt} + \frac{G}{2} (1 + \frac{1}{n})G_Q + G(1 - \frac{1}{n})G_{pq}P = GS_q
\]  

(5.59)

Comparing these equations to the ones derived in Section 5.2, which were given by Equations (5.43) and (5.44), we notice that their forms are identical except for the attenuation coefficient \( n \) which appears here as a first order term, rather than squared as in the previous section.

The conclusion is that the LMS algorithm, when applied with the presence of the band-pass filter, does not lead exactly to the structure given in Figure 4.4 for a given upper sideband attenuation coefficient \( n \). However, that structure will have the same performance if the filter used has an attenuation coefficient twice (in dB) that of the filter used when the LMS algorithm was applied.

Note that when there is no output filter, i.e., when \( n = 1 \), the differential equations from both sections are identical and reduce to

\[
\frac{dP}{dt} + G_1P = GS_p
\]  

(5.60)

\[
\frac{dQ}{dt} + G_1Q = GS_q
\]  

(5.61)

a result that was derived in Chapter III.
Also, when there is perfect attenuation of the upper sideband, i.e., when \( n = \omega \), the results of both sections again match, but are this time given by

\[
\frac{dP}{dt} + G \frac{d}{dt} \phi_1 P + G_{pq} Q = G_S p \tag{5.62}
\]

\[
\frac{dQ}{dt} + G \frac{d}{dt} \phi_1 Q + G_{pq}^T P = G_S q \tag{5.63}
\]

In subsequent work, we will be using the array structure given in Figure 4.4 and along with it the differential equations which describe it, namely Equations (5.58) and (5.59).

5.4 STEADY-STATE SOLUTION

In this section, the steady-state solution that satisfies the differential Equations (5.58) and (5.59) will be given as a function of the parameter \( n \).

At steady-state we have

\[
\frac{dP}{dt} = 0 \tag{5.64}
\]

and

\[
\frac{dQ}{dt} = 0 \tag{5.65}
\]

To simplify notations let

\( a = 1 + 1/n \) \tag{5.66}
and

\[ b = 1 - 1/n \]  \hspace{1cm} (5.67)

Applying Equations (5.64) and (5.65) to Equations (5.58) and (5.59), we obtain

\[ a \phi_i P + 2b \phi_{pq} Q = 2S_p \]  \hspace{1cm} (5.68)

\[ a \phi_i Q + 2b \phi_{pq}^T P = 2S_q \]  \hspace{1cm} (5.69)

When combined, these two equations can be written in block-matrix form as

\[
\begin{pmatrix}
-a \phi_i & 2b \phi_{pq} \\
-2b \phi_{pq}^T & a \phi_i
\end{pmatrix}
\begin{pmatrix}
P \\
Q
\end{pmatrix} = 2
\begin{pmatrix}
S_p \\
S_q
\end{pmatrix}
\]  \hspace{1cm} (5.70)

Therefore we have

\[
\begin{pmatrix}
P \\
Q
\end{pmatrix} = 2
\begin{pmatrix}
-a \phi_i & 2b \phi_{pq} \\
-2b \phi_{pq}^T & a \phi_i
\end{pmatrix}^{-1}
\begin{pmatrix}
S_p \\
S_q
\end{pmatrix}
\]  \hspace{1cm} (5.71)

where we assume that the 2N x 2N matrix in Equation (5.71) can be inverted.

In order to write Equation (5.71) more explicitly, we will make use of a result known in linear algebra. Given four square matrices A, B, C and D, and that A\(^{-1}\) and B\(^{-1}\) exist, then the following formula holds:
\[
\begin{bmatrix}
A & D \\
C & B
\end{bmatrix}^{-1} = 
\begin{bmatrix}
(A - DB^{-1}C)^{-1} & -E \Delta^{-1} \\
-\Delta^{-1}F & \Delta^{-1}
\end{bmatrix}
\]  
(5.72) 

with 
\[\Delta = B - CA^{-1}D\]  
(5.73) 
\[E = A^{-1}D\]  
(5.74) 
\[F = CA^{-1}\]  
(5.75) 

\(\Delta\) is known as the Schur complement of \(A\) (see [15]).

Application of the above relation to Equation (5.71), we obtain the correspondence:

\[A = B = a_{ij}ij\]  
(5.76) 
\[C = 2b_{pq}^{\top}\]  
(5.77) 
\[D = 2b_{pq}\]  
(5.78) 

We recall that 
\[\phi_{\top} = \frac{1}{2} \overline{XX^\top}\]  
(5.79) 

where \(\overline{XX^\top}\) is the covariance matrix of the input vector and is never singular when thermal noise is taken into consideration. Hence, we consider that its inverse exists.

Using Equations (5.76) through (5.78) to evaluate the quantities given by Equations (5.72) through (5.75) and after defining the matrices
\[ \phi_p = a \phi_I - \frac{4b^2}{a} \phi_{pq} \phi_{pq}^{-1} \phi \] (5.80)

and

\[ \phi_q = a \phi_I - \frac{4b^2}{a} \phi_{pq} \phi_{pq}^{-1} \phi \] (5.81)

we obtain, from Equation (5.71), the final steady-state expressions of the inphase vector P and the quadrature vector Q as functions of the output upper sideband attenuation coefficient \( \eta \) introduced through the parameters \( a \) and \( b \). These expressions are:

\[ P = 2[\phi_{pq}^{-1} S_p - \frac{2b}{a} \phi_{pq} \phi_{pq}^{-1} S_q] \] (5.82)

\[ Q = 2[\phi_{pq}^{-1} S_q - \frac{2b}{a} \phi_{pq} \phi_{pq}^{-1} S_p] \] (5.83)

where we recall that the matrices \( S_p \) and \( S_q \) were defined by Equations (5.31) and (5.32), \( \phi_I \) and \( \phi_{pq} \) by Equations (5.56) and (5.57), \( \phi_p \) and \( \phi_q \) by Equations (5.80) and (5.81) and the parameters \( a \) and \( b \) by Equations (5.66) and (5.67) respectively.

We note that if there is no output filter \( (\eta=1) \) we have \( a=2, b=0 \) and the solutions given by Equations (5.82) and (5.83) reduce to

\[ P = \phi_{pq}^{-1} S_p \] (5.84)

and

\[ Q = \phi_{pq}^{-1} S_q \] (5.85)

which is the result obtained in Chapter III.
Equations (5.82) and (5.83) represent the steady-state solution that will result in the least-mean-square error for a given $\eta$ and for a given scenario at the array input. In the next section we will show the array performance degradation as a function of this parameter and we will see that optimal performance is obtained for $\eta = \infty$.

5.5 ARRAY PERFORMANCE VERSUS UPPER SIDEBAND ATTENUATION

In this section we will first check the validity of the steady-state solution derived in the previous section by considering a particular case of a two-element array. Then, we will treat the same example to show the adaptive antenna interference rejection performance versus the attenuation coefficient $\eta$ using the same steady-state solution.

Thus, consider the two-element adaptive array example with the input scenario depicted in Figure 4.1 of Chapter IV. In this example, a CW desired signal given by $S \cos \omega_d t$ is incident from broadside and a CW interfering signal given by $I \cos \omega_i t$ is arriving at an angle $\theta_I$ from broadside. The interference radian frequency $\omega_I$ is close but not equal to $\omega_d$, a necessary condition when both the interference and the desired signal are CW signals.

In this first step, we will assume a perfect attenuation of the upper sidebands at the array output, i.e. $\eta = \infty$.

We have seen that by nulling the interference and forcing the desired signal to be equal to the reference signal, the ideal steady-state solution of $P$ and $Q$ for that particular example was given
by Equations (4.21) and (4.22) in the previous chapter, namely

\[ P_1 = P_2 = \frac{R}{S} \quad (5.86) \]

and

\[ Q_1 = -Q_2 = \frac{R \cos \gamma}{S \sin \gamma} \quad (5.87) \]

where \( R \) and \( S \) represent the amplitude of the reference signal and the desired signal respectively and \( \gamma = \pi \sin \theta_i \) for a spacing of \( \lambda/2 \) between the array elements.

Evaluated numerically with the parameters used during the two-loop simulation, that is, with \( \theta_i = 60^\circ \) and \( R = S = 10 \), Equations (5.86) and (5.87) give

\[ P_1 = P_2 = 1 \quad (5.88) \]

\[ Q_1 = -Q_2 = 0.213 \quad (5.89) \]

The above results will allow us to compare the ideal solutions given by Equations (5.86) and (5.87) with the results obtained with the closed-loop simulation of a two-element array and with the theoretical solutions given by Equations (5.82) and (5.83).

Now we evaluate Equations (5.82) and (5.83) for the above example.

The elements of the input signal vector \( X \), referring to the scenario of Figure 4.1 of Chapter IV, are given by

\[ X_1(t) = S\cos \omega_d t + I\cos \omega_1 t \quad (5.90) \]

for element one, and by
\[ x_2(t) = S \cos \omega_d t + I \cos (\omega_I t - \gamma) \] (5.91)

for element two.

The reference signal is taken to be

\[ R(t) = R \cos (\omega_0 - \omega_d) t \] (5.92)

where \( \omega_0 \) is the radian frequency of the I.F. weights.

The basic matrices which are involved in Equations (5.82) and (5.83) can now be expressed in terms of the above signals. Evaluating each entry and taking care to perform the necessary time-averages so as to have time-invariant matrices, we obtain

\[
\begin{align*}
\phi_I &= \frac{1}{4} \begin{bmatrix} (S^2 + I^2) & (S^2 + I^2 \cos \gamma) \\ (S^2 + I^2 \cos \gamma) & (S^2 + I^2) \end{bmatrix} \\
\phi_{pq} &= \frac{1}{8} \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} \\
S_p &= \frac{RS}{4} \begin{bmatrix} 1 \\ 1 \end{bmatrix} \\
S_q &= \begin{bmatrix} 0 \\ 0 \end{bmatrix}
\end{align*}
\] (5.93-5.96)
Using these results to evaluate Equations (5.80), (5.81), (5.82) and (5.83) and after replacement of the parameters $a$ and $b$ by their expressions (5.66) and (5.67), we obtain the final steady-state solution as a function of the attenuation coefficient $\eta$:

$$P_1 = P_2 = \frac{R}{S} \cdot \frac{1 + \frac{1}{\eta}}{(1 + \frac{1}{\eta})^2 + 2 \frac{I^2 \sin^2 \gamma}{\eta S^2 (1 - \cos \gamma)}}$$ (5.97)

for the inphase components of loop 1 and 2 and

$$Q_1 = -Q_2 = \frac{R}{S} \cdot \frac{1 + \cos \gamma}{\sin \gamma} \cdot \frac{1 - \frac{1}{\eta}}{(1 + \frac{1}{\eta})^2 + 2 \frac{I^2 \sin^2 \gamma}{\eta S^2 (1 - \cos \gamma)}}$$ (5.98)

for the quadrature components.

We verify that for a perfect attenuation of the upper sideband ($\eta = \infty$), the above steady-state solution reduces to the ideal one given previously by Equations (5.86) and (5.87).

In Equations (5.97) and (5.98), we notice that if the attenuation coefficient $\eta$ is not large enough, the array weights will be sensitive to the interference signal amplitude $I$ which appears in the denominators. Clearly, since the ideal steady-state solution is obtained only when $\eta = \infty$, the array output signal-to-interference ratio is an increasing function of the attenuation coefficient, as it will be seen shortly.

In order to illustrate the array performance for the example treated in this section, the signal-to-interference ratio (SIR) is
With weight 1 given by
\[ W_1 = P_1 \cos \omega_0 t + Q_1 \sin \omega_0 t \] (5.99)
and weight 2 by
\[ W_2 = P_2 \cos \omega_0 t + Q_2 \sin \omega_0 t \] (5.100)
the array output due to the desired signal only is given by
\[ Y_D(t) = \frac{S}{2} (P_1 + P_2) \cos(\omega_0 - \omega_d)t + \frac{S}{2} (Q_1 + Q_2) \sin(\omega_0 - \omega_d)t \] (5.101)
and the output due to the interfering signal only is given by
\[ Y_I(t) = \frac{I}{2} (P_1 + P_2 \cos \gamma + Q_2 \sin \gamma) \cos(\omega_0 - \omega_I)t 
+ \frac{I}{2} (Q_1 - P_2 \sin \gamma + Q_2 \cos \gamma) \sin(\omega_0 - \omega_I)t \] (5.102)
where only the difference-frequency terms are retained. The sum-frequency terms are also present at the output but they are attenuated by the coefficient \( \eta \).

The SIR is not computed by considering the total output power present in both the desired signal and the interfering signal, but rather by taking the ratio of the maximum amplitudes of the difference-frequency signals given by Equations (5.101) and (5.102).

The input signal-to-interference ratio is set to 0 dB, that is, \( S = I \). Also with \( R = S \) and with \( P_1, P_2, Q_1 \) and \( Q_2 \) given by Equations (5.97) and (5.98), the output signal-to-interference ratio is
\[
SIR(n) = \frac{P_1 + P_2}{[(P_1 + P_2 \cos \gamma + Q_2 \sin \gamma)^2 + (Q_1 - P_2 \sin \gamma + Q_2 \cos \gamma)^2]^{1/2}}
\]

Equation (5.103) is plotted for values of \( n \) from 0 to 80 dB and for an interference incident angle of 60°. The performance curve which results is shown in Figure 5.1. The SIR varies from 13.6 dB for \( n = 0 \) dB to 87.6 dB for \( n = 80 \) dB.

In order to verify the theoretical results of this section, a computer simulation was performed using this two-element array example. The simulated array was allowed to adapt and reach steady-state with the same parameters and input scenario as before. The results of the simulation are as follows:

- Upper sideband attenuation : \( n = 33.84 \) dB
- Output signal-to-interference ratio : \( SIR = 40.789 \) dB
- Inphase component for loop 1 : \( P_1 = 1.000 \)
- Inphase component for loop 2 : \( P_2 = 0.991 \)
- Quadrature component for loop 1 : \( Q_1 = 0.207 \)
- Quadrature component for loop 2 : \( Q_2 = -0.235 \)

Figure 5.2 shows the plots for the inphase and quadrature components whose steady-state values are indicated above. The array was left to adapt for 22 ms with a sampling rate of 0.1 microsec.

On the other hand, evaluating \( P_1 \) and \( P_2 \) through Equation (5.97), \( Q_1 \) and \( Q_2 \) through Equation (5.98) and SIR through Equation (5.103) with \( n = 33.94 \) dB, we obtain:
Figure 5.1 Theoretical Output Signal-to-Interference Ratio versus Upper Sideband Attenuation $\eta$ for a Two-Element Array.
Figure 5.2  Inphase and Quadrature Components for a Simulated Two-Element LMS I.F. Array with Output Filter.
- SIR = 41.596 dB
- $P_1 = P_2 = 0.9767$
- $Q_1 = -Q_2 = 0.2003$

As we can see, the theoretical results and the simulated ones are fairly close to each other. The small discrepancies that occur are due primarily to limitations in signal representation and filtering effects during the computer simulation.

Finally, one could substitute the optimal solution of $P$ and $Q$ given by Equations (5.82) and (5.83) into Equation (5.33) of the mean-square error in order to find an expression for the least-mean-square error. The mathematics involved is rather cumbersome in this last step, so that the expression will not be given. However, from the performance curve of Figure 5.1, one can reasonably conclude that the least-mean-square error will be some decreasing function of $n$.

Thus, in this chapter, the differential equations for the inphase vector $P$ and the quadrature vector $Q$ have been derived in terms of the upper sideband attenuation coefficient in two different ways. The first was by application of the LMS algorithm, the second by inspection of the feedback array structure already given in Chapter IV in Figure 4.4. Except for the filter attenuation coefficient required in each case, the two approaches led to the same differential equations. The steady-state solution of $P$ and $Q$ was then derived as a function of this coefficient. Finally, an example of a two-element array was treated to illustrate the validity of this analysis and to show the array performance versus the upper sideband attenuation.
CHAPTER VI

EFFECTS OF FILTERS ON I.F. ARRAY STABILITY AND PERFORMANCE

6.1 INTRODUCTION

This chapter addresses the I.F. array stability problem due to filters in the feedback loops.

In chapters IV and V, a justification for the output band-pass filter was given. It is desirable to consider adding another band-pass filter at the correlator output (lower mixer) of the array feedback loops. Because of the wide signal bandwidth that might be present at that location (80 MHz for the actual array implementation), this filter becomes quite necessary if one chooses to implement the weight processor in digital form. Because it is only necessary to process the central component of either sideband after the correlator mixer, even with an analog implementation one would wish to attenuate the undesirable frequency components surrounding the chosen central component.

Hence, in either case, a band-pass filter preceding the weight processor is desirable. A general block diagram showing the location of the filters is shown in Figure 6.1.
Figure 6.1 The LMS I.F. Array Loop Showing Group Delays $t_1$ and $t_2$. 

---

105
The effect of these two filters, that is, the array output filter and the correlator output filter, on array stability and performance is the subject of the present discussion. It will be shown that the group delays associated with these filters have a significant effect on the stability of the I.F. array.

In Section 6.2 the effects of these time delays on the inphase and quadrature components and also on the output signal-to-interference ratio will be shown through computer simulation. Then, in Section 6.3, the differential equations derived in the previous chapter will be revised in order to include these delays. Finally, Section 6.4 analyzes the transient behavior of the inphase and quadrature components as a function of the above group delays.

6.2 COMPUTER SIMULATION OF THE EFFECTS OF GROUP DELAYS

The I.F. array structure to be analyzed was shown in Figure 4.4 of Chapter IV; but the correlator filter has been added for reasons previously stated. For convenience, this final structure is shown in Figure 6.1.

From experimental work [9], it was noted that the introduction of the output filter with group delay $t_1$ in Figure 6.1 leads to instability of the I.F. array. This is reasonable from a closed-loop system viewpoint since this filter is in the feedback path of the processor loop. The general effect due to this filter can be interpreted as a decorrelation effect which occurs in the lower mixer of Figure 6.1. More precisely, one port of this mixer contains the array input signal.
passing through the upper filter of Figure 6.1 and is delayed by $t_1$ while the other port contains the same input signal but without delay.

The correlator filter in Figure 6.1 was not incorporated in the experimental array because it was believed that it would lead to the same difficulties. As will be shown in subsequent sections, both filters are indeed sources of instability.

For a filter with transfer function

$$H(\omega) = |H(\omega)| / \theta(\omega), \quad (6.1)$$

the group delay is defined as the derivative of $-\theta(\omega)$ with respect to frequency. Hence, the group delay $tg$ is by definition:

$$tg(\omega) = \frac{d[-\theta(\omega)]}{d\omega} \quad (6.2)$$

It is a measure of how much the envelope of a signal is delayed when it goes through the filter.

Since in general the phase function $\theta(\omega)$ is not necessarily a linear function of frequency, even within the bandwidth of interest, the group delay will not always be constant. However, in our computer simulation, as well as in our subsequent analysis, a constant time-delay will be assumed for simplicity.

In the practical domain, one can come close to the above assumption by choosing filters with nearly constant group delays such as the maximally-flat-time-delay filters (MFTD) often referred to as either Bessel filters or Gaussian filters. However, their attenuation in the stopband is somewhat limited.
The computer simulation was performed with the array example used in the previous chapters, that is, a two-element array with a desired signal coming from broadside and an interfering signal coming from an angle of 60° from broadside. The input signal-to-interference ratio was set to 0 dB (S = I = 10), the loop gain to G = 10 and the sampling time to 1 microsec.

The two band-pass filters in Figure 6.1 were simulated as sixth-order Butterworth digital filters. In the previous simulations (and in subsequent simulations) where attention is focused on array performance, the bandwidth of these filters is chosen in such a way that the corresponding group delay has no serious effect on array performance. This is done by setting the group delay to be a multiple of the period of the CW signal that goes through the filter in question. For this simulation, the bandwidth for both filters was set to 2 KHz which yields a group delay of approximately 500 us which is five times the period of the weight waveform (sum frequency term at the correlator output), and approximately twice the period of the array output signal.

At some time during the simulation, it would be necessary to have the option of varying the group delays. This is desirable to determine the array performance as a function of these delays. To accommodate this possibility, extra time delays t₁ and t₂ now included in the signal representation will be introduced. The signals passing through the filters are thus delayed accordingly during the simulation with t₁ and t₂ as input parameters which can be set to desirable values.
The inphase and quadrature components P and Q for the two loops are chosen to show the effect of \( t_1 \) and \( t_2 \) on array stability. First, with \( t_1 \) and \( t_2 \) set to zero, P and Q reach a constant value at steady-state (see Figure 6.2). Hence, the array is stable as expected. However, if we set, for example, \( t_1 = 80 \text{\,\mu s} \) and \( t_2 = 0 \), P and Q no longer reach stable steady-state values (see Figure 6.3). Similarly, with \( t_1 = 0 \) and \( t_2 = 50 \text{\,\mu s} \), the plots shown in Figure 6.4 depict P and Q growing without bound.

Clearly, for the above group delay values the array is unstable. However, as we will demonstrate both by computer simulation and analysis, there are values of group delays for which the array is stable and its performance is preserved.

To demonstrate this property partially, the output signal-to-interference ratio is plotted as a function of \( t_1 \) and \( t_2 \). Figure 6.5 shows the signal-to-interference ratio (SIR) plotted versus \( t_1 \) with \( t_2 = 0 \). Figure 6.6 shows the plot versus \( t_2 \) with \( t_1 = 0 \). For each point in these figures the simulated array is permitted to adapt for 11 ms and the SIR is computed as the ratio of maximum amplitudes. In Figure 6.5 the group delay \( t_1 \) is increased to include approximately the period of the output signal (250 \text{\,\mu s}). In Figure 6.6 \( t_2 \) is increased to include the period of the weighting signal (100 \text{\,\mu s}).

As can be seen in both figures, the array experiences performance degradation for certain ranges of group delays. In these figures, only the regions near the maxima correspond to stable steady-state solutions. The off-peak regions correspond to marginally stable or unstable conditions.
Figure 6.2 Inphase and Quadrature Components of a Two-Element I.F. Array with no Group Delays; $t_1 = 0$, $t_2 = 0$. 
Figure 6.3 Effect of Group Delay $t_1$ on the Inphase and Quadrature Components of a Two-Element I.F. Array; $t_1 = 80$ μs, $t_2 = 0$. 
Figure 6.4 Effect of Group Delay $t_2$ on the Inphase and Quadrature Components of a Two-Element I.F. Array; $t_1 = 0$, $t_2 = 50 \mu$s.
Figure 6.5  Effect of Group Delay $t_1$ on Output Signal-to-Interference Ratio for a Two-Element I.F. Array with $t_2 = 0$. 

113
Figure 6.6 Effect of Group Delay $t_2$ on Output Signal-to-Interference Ratio for a Two-Element I.F. Array with $t_1 = 0$. 
In summary, the computer simulation of this section serves as a preliminary method of analysis to show that the group delays $t_1$ and $t_2$ corresponding to the output filter and the correlator filter respectively can cause the array feedback loops to be unstable.

The following section will provide a key to a more analytical study of these delays.

6.3 DIFFERENTIAL EQUATIONS FOR THE WEIGHTING FUNCTIONS INCLUDING GROUP DELAYS

The purpose of this section is to derive the differential equations for $P$ and $Q$ which include the time delays $t_1$ and $t_2$ shown in Figure 6.1.

By inspection of Figure 6.1, which depicts the I.F. array loop "i", and after generalization to $N$ loops, we write:

$$\frac{dP(t)}{dt} = G e(t-t_2)X(t-t_2)\cos\omega_0 t$$

(6.3)

for the inphase vector, and

$$\frac{dQ(t)}{dt} = G e(t-t_2)X(t-t_2)\sin\omega_0 t$$

(6.4)

for the quadrature vector.

The vector $X$ in the above equation is the input signal vector to the correlator mixer (lower mixer in Figure 6.1) delayed by $t_2$.

The error signal is written as:

$$e(t) = R(t) - [W^T(t-t_1)X(t-t_1)]_L - \frac{1}{n} [W^T(t-t_1)X(t-t_1)]_U$$

(6.5)
where the vector $X$ is the input signal vector to the weight multiplier (upper mixer in Figure 6.1) passing through the output filter and hence being delayed by $t_1$.

Here we retain the same notations as before regarding lower and upper sidebands and filter attenuation coefficient.

After going through the lower filter in Figure 6.1, the error signal undergoes delay $t_2$. Hence, insertion of $t_2$ in Equation (6.5) yields:

$$E(t-t_2)=R(t-t_2)-[W^T(t-t_1-t_2)X(t-t_1-t_2)]L^{-1}[W^T(t-t_1-t_2)X(t-t_1-t_2)]U$$

(6.6)

The delayed I.F. weights in Equation (6.6) are explicitly written in terms of the inphase and quadrature components $P$ and $Q$ as:

$$W(t-t_1-t_2) = P(t-t_1-t_2)\cos\omega_0(t-t_1-t_2) + Q(t-t_1-t_2)\sin\omega_0(t-t_1-t_2)$$

(6.7)

After substitution of (6.7) into (6.6) and afterwards of (6.6) into both (6.3) and (6.4) we obtain equations similar to equations (5.48) and (5.49) of Chapter V, with the exception that all the quantities involved here will contain delays $t_1$ and $t_2$.

The matrix manipulation technique and the method of rearranging terms are identical to those of chapter V; thus, they will not be shown here. The end product of these manipulations will yield expressions for the derivatives of $P$ and $Q$ similar to Equations (5.51) and (5.52) of the previous chapter. In condensed notation these expressions, which are
the differential equations of $P$ and $Q$, are:

$$\frac{dP(t)}{dt} + G(\phi_1+\phi_2/n)P(t-t_1-t_2) + G(\phi_3+\phi_4/n)Q(t-t_1-t_2) = G S_{dp} \quad (6.8)$$

and

$$\frac{dQ(t)}{dt} + G(\phi_5+\phi_6/n)P(t-t_1-t_2) + G(\phi_7+\phi_8/n)Q(t-t_1-t_2) = G S_{dq} \quad (6.9)$$

The matrices introduced in the above equations are defined as follows:

$$\phi_1 = \frac{[X(t-t_2)\cos\omega t]_L [X^T(t-t_1-t_2)\cos\omega (t-t_1-t_2)]_L}{(6.10)}$$

$$\phi_2 = \frac{[X(t-t_2)\cos\omega t]_U [X^T(t-t_1-t_2)\cos\omega (t-t_1-t_2)]_U}{(6.11)}$$

$$\phi_3 = \frac{[X(t-t_2)\cos\omega t]_L [X^T(t-t_1-t_2)\sin\omega (t-t_1-t_2)]_L}{(6.12)}$$

$$\phi_4 = \frac{[X(t-t_2)\cos\omega t]_U [X^T(t-t_1-t_2)\sin\omega (t-t_1-t_2)]_U}{(6.13)}$$

$$\phi_5 = \frac{[X(t-t_2)\sin\omega t]_L [X^T(t-t_1-t_2)\cos\omega (t-t_1-t_2)]_L}{(6.14)}$$

$$\phi_6 = \frac{[X(t-t_2)\sin\omega t]_U [X^T(t-t_1-t_2)\cos\omega (t-t_1-t_2)]_U}{(6.15)}$$

$$\phi_7 = \frac{[X(t-t_2)\sin\omega t]_L [X^T(t-t_1-t_2)\sin\omega (t-t_1-t_2)]_L}{(6.16)}$$

$$\phi_8 = \frac{[X(t-t_2)\sin\omega t]_U [X^T(t-t_1-t_2)\sin\omega (t-t_1-t_2)]_U}{(6.17)}$$

$$S_{dp} = R(t-t_2)X(t-t_2)\cos\omega t \quad (6.18)$$
and finally

\[ S_{dq} = R(t-t_2)X(t-t_2)\sin\omega_0 t \]  \hspace{1cm} (6.19)

where as usual the overbar denotes time averaging.

Because of the quantity \((t_1+t_2)\), Equations (6.8) and (6.9) represent difference-differential equations in \(P\) and \(Q\). However, the time-delays \(t_1\) and \(t_2\) will be generally small compared to the speed of variation of \(P(t)\) and \(Q(t)\) and furthermore at steady-state these components will be relatively constant. Hence, as a first approximation we consider that

\[ P(t-t_1-t_2) = P(t) \]  \hspace{1cm} (6.20)

and

\[ Q(t-t_1-t_2) = Q(t) \]  \hspace{1cm} (6.21)

so that Equations (6.8) and (6.9) become regular differential equations and constitute the result sought in this section.

6.4 ANALYTICAL STUDY OF GROUP DELAYS EFFECTS

In this section we will treat the usual example of the two-element array with deterministic inputs, and provide an analytical explanation of the effects of \(t_1\) and \(t_2\) on stability.

Throughout this chapter we have attempted to show the effects of \(t_1\) and \(t_2\) on array stability. In doing so here, one could solve the differential equations given by (6.8) and (6.9) for \(P(t)\) and \(Q(t)\) and display explicitly their dependence on \(t_1\) and \(t_2\). However, this task is
algebraically involved even for a two-element array, and therefore such a solution will not be considered here. Instead, since we are concerned primarily with the stability problem, it is only necessary to analyze the transient behavior in terms of \( t_1 \) and \( t_2 \). The way to achieve this is through the characteristic equation associated with the differential equations of \( P \) and \( Q \).

The differential equations (6.8) and (6.9) for the particular case mentioned above will be evaluated. The transient solution will be analyzed by treating the array feedback loop as a linear system whose input is the reference signal port.

The input signal vector \( X \) considered for this example has its entries given by

\[
X_1(t) = S \cos \omega t + I \cos \omega t \quad (6.22)
\]

for element one, and by

\[
X_2(t) = S \cos \omega t + I \cos(\omega t - \gamma) \quad (6.23)
\]

for element two, in accordance with the input scenario described in Figure 4.1 of Chapter IV.

Evaluating the matrices given by Equations (6.10) - (6.17) with the above signals, it is found that

\[
\phi_5 = - \phi_3 \quad (6.24)
\]

\[
\phi_6 = - \phi_4 \quad (6.25)
\]

\[
\phi_7 = \phi_1 \quad (6.26)
\]

\[
\phi_8 = \phi_2 \quad (6.27)
\]
To further condense the notations in Equations (6.8) and (6.9),

\[
\phi_A = \phi_1 + \frac{\phi_2}{\pi} \quad (6.28)
\]

and

\[
\phi_B = \phi_3 + \frac{\phi_4}{\pi} \quad (6.29)
\]

Hence, the differential equations for this particular case become:

\[
\frac{dP(t)}{dt} + G \phi_A P(t) + G \phi_B Q(t) = G S_{dp} \quad (6.30)
\]

\[
\frac{dQ(t)}{dt} + G \phi_A Q(t) - G \phi_B P(t) = G S_{dq} \quad (6.31)
\]

A useful tool in dealing with the transient behavior of Equations (6.30) and (6.31) is by means of the D operator. This operator, which acts on P and Q is by definition given by

\[
D = \frac{d}{dt} \quad (6.32)
\]

therefore we have

\[
\frac{dP(t)}{dt} = DP = D \begin{bmatrix} P_1 \\ P_2 \end{bmatrix} \quad (6.33)
\]

and

\[
\frac{dQ(t)}{dt} = DQ = D \begin{bmatrix} Q_1 \\ Q_2 \end{bmatrix} \quad (6.34)
\]

Let

\[
\phi_A = \begin{bmatrix} a_1 & a_2 \\ a_3 & a_4 \end{bmatrix} \quad (6.35)
\]
Substitution of Equations (6.33) - (6.38) into (6.30) and (6.31), yields:

\[
\begin{pmatrix}
P_1 \\
P_2 \\
Q_1 \\
Q_2
\end{pmatrix} = 
\begin{pmatrix}
P_1 \\
P_2 \\
Q_1 \\
Q_2
\end{pmatrix}
\]

with the matrix \( \Delta \) given by

\[
\Delta = 
\begin{pmatrix}
(D+G_a) & G_a & G_b & G_b \\
G_a & (D+G_a) & G_b & G_b \\
-G_b & -G_b & (D+G_a) & G_a \\
-G_b & -G_b & G_a & (D+G_a)
\end{pmatrix}
\]

Now one can solve Equation (6.39) by inverting this determinant and obtain the four uncoupled differential equations for the \( P \)'s and \( Q \)'s after setting the operator \( D \) equal to the expression given by Equation (6.32).
However, since we are interested only in the natural response of the system we can set the right-hand side of Equation (6.39) to zero in order to simplify the analysis. Although this is equivalent to having a zero reference signal, this will not affect the transient behavior since the array can be considered as a linear system with respect to the reference signal.

Hence, it is only necessary to find the characteristic roots of the system by setting the determinant to zero.

The characteristic equation of this system is

$$\text{det}(A) = 0 \quad (6.41)$$

which will result in four characteristics roots which we denote $D_1$, $D_2$, $D_3$ and $D_4$.

The general form of the transient solution of each component will thus be written as a combination of exponentials containing these roots.

For instance, the transient solution of $P_1(t)$ will have the form:

$$P_1(t) = c_1e^{D_1t} + c_2e^{D_2t} + c_3e^{D_3t} + c_4e^{D_4t} \quad (6.42)$$

Had the right-hand side of Equation (6.39) not been set to zero, a steady-state term would have been added to the transient solution in Equation (6.42).
By inspection of Equation (6.42) it is clear that in order for the transient solution to decay to zero and hence to assure stability, the real part of all the roots must be strictly negative. Therefore we must have

\[ \text{Re}[D_i] < 0 \quad (6.43) \]

to guarantee stability.

Evaluating Equation (6.41), the characteristic equation was found to be:

\[ D^4 + H_3D^3 + H_2D^2 + H_1D + H_0 = 0 \quad (6.44) \]

with its coefficients given by:

\[ H_0 = 4G \alpha_1^4 + G \alpha_1^2E_2 + Ga_1E_1 + E_0 \quad (6.45) \]
\[ H_1 = 4G \alpha_1^3 + 2Ga_1E_2 + E_1 \quad (6.46) \]
\[ H_2 = 6G \alpha_1^2 + E_2 \quad (6.47) \]
\[ H_3 = 4Ga_1 \quad (6.48) \]

and with

\[ E_0 = 4G^4 (a_2^2a_3+b_3^2a_2^2+2b_1^2a_2a_3+b_1^4-2b_1^2b_2b_3+b_2^2a_3+b_2^2b_3^2) \quad (6.49) \]
\[ E_1 = -4G^3 b_1 (b_3a_2+b_2a_3) \quad (6.50) \]
\[ E_2 = 2G^2 (b_1^2b_2+b_3^2a_2a_3) \quad (6.51) \]

where we recall that the a's and b's in these expressions contain explicitly the parameters \( t_1 \) and \( t_2 \).
For a given set of values for $t_1$ and $t_2$ the characteristic roots can now be found by solving Equation (6.44) numerically.

In order to show the behavior of these roots as a function of either $t_1$ or $t_2$, computer plots are generated by solving (6.44) for several values of these time-delays.

The following are the two-element array parameters used to evaluate the quantities given by Equations (6.10) - (6.17), (6.28) and (6.29) and the coefficients of the characteristic equation:

- Amplitude of desired signal : $S = 10$
- Amplitude of interfering signal : $I = 10$
- Angle of interfering signal arrival : $\theta = 60^\circ$
- Frequency of desired signal : $f_d = 6$ KHz
- Frequency of interfering signal : $f_1 = 6.4$ KHz
- Frequency of I.F. weight signal : $f_0 = 10$ KHz
- Loop gain : $G = 10$
- Upper sideband attenuation : $n = 51.6$ dB

The plots showing the four characteristic roots $D_1$, $D_1^*$, $D_2$ and $D_2^*$, where * denotes complex conjugate, are given in Figures 6.7 - 6.10 respectively. Both their real and imaginary parts are plotted versus $t_1$ and $t_2$.

From these figures we can easily see that there are intervals for which the real part is positive thus causing the exponentials in (6.42) to grow without bound and hence making the array unstable.
This drastic effect was already seen in the plots given in Figures 6.3 and 6.4 shown in Section 6.2, where the values of $t_1$ (80 µs) and $t_2$ (50 µs) correspond to a positive real part of the roots.

Null real parts are not acceptable either since they lead to oscillatory effects in $P(t)$ and $Q(t)$.

Also, there are values of $t_1$ and $t_2$ for which the roots have negative real parts but small amplitudes. A relatively small amplitude will not allow the components to reach steady-state rapidly enough.

Thus, besides causing instability, the group delays can cause the array to have a long adaptation time.

It is of interest to note that as a function of $t_1$, the real part of each root has a period equal to that of the array output signal (250 µs), and that as a function of $t_2$ it has a period equal to that of the weight signal (100 µs). This property will be found useful in connection with the time-delay compensation which will be treated in the next chapter.

As a final note, we can verify that the characteristic roots are linear functions of the loop gain $G$. Figure 6.11 shows the computer plots with $t_1$ and $t_2$ set to zero. Since the roots occur in conjugate pairs in general, only two plots are shown, the other being identical for this particular case.

The characteristic roots can then be written as:

$$D_1 = -G \lambda_1$$  \hspace{1cm} (6.52)

and the transient solution for $P_1$, given by Equation (6.42), will
Figure 6.7 Characteristic Root $D_1$ for a Two-Element LMS I.F. Array versus Group Delays $t_1$ and $t_2$. 

a) Versus $t_1$

b) Versus $t_2$
Figure 6.8 Characteristic Root $D_1^*$ for a Two-Element LMS I.F. Array versus Group Delays $t_1$ and $t_2$. 

a) Versus $t_1$

b) Versus $t_2$
Figure 6.9 Characteristic Root $D_2$ for a Two-Element LMS I.F. Array versus Group Delays $t_1$ and $t_2$. 

a) Versus $t_1$  
b) Versus $t_2$
Figure 6.10 Characteristic Root $\hat{\nu}_2$ for a Two-Element LMS I.F. Array versus Group Delays $t_1$ and $t_2$. 

a) Versus $t_1$

b) Versus $t_2$
Figure 6.11 Characteristic Roots for a Two-Element LMS I.F. Array versus Loop Gain $G$ with $t_1 = t_2 = 0$. 

a) Root $D_1$.  

b) Root $D_2$. 
contain just two exponential terms and thus will be of the form

\[ P_1(t) = c_1 e^{-G\lambda_1 t} + c_2 e^{-G\lambda_2 t} \]  

(6.53)

The \( \lambda \)'s can be referred to as a set of pseudo-eigenvalues which are functions of the input signal amplitudes and will contain, in general, the parameters \( t_1 \) and \( t_2 \).

Thus, in this chapter the effects of the group delays introduced by the output filter and the correlator filter on the inphase and quadrature components of the I.F. array were shown. The analysis of the I.F loop was performed and it was shown that both group delays constitute major sources of instability.
CHAPTER VII

I.F. ARRAY LOOP STABILIZATION

7.1 INTRODUCTION

In Chapters IV and V it was shown that removal of one of the frequency sidebands at the array output was a necessary condition for optimum performance in interference rejection. The introduction of the output filter was, hence, greatly justified in both chapters.

The correlator filter, also desirable for an analog implementation of the I.F. array weight processor, is mandatory if this processor is to be implemented in digital form because of prefiltering requirements. In addition, only the central component which comes out of the correlator mixer is of interest for both implementations.

On the other hand, it was shown in Chapter VI that the I.F. array feedback loops can become unstable because of the group delays introduced by these filters, and since steady-state operation cannot be achieved under these conditions, the array cannot provide interference rejection.

This chapter presents a method to achieve loop stability in the presence of these filters. In Section 7.2 the stability requirements will be derived, and in Section 7.3 their effectiveness will be demonstrated through computer simulation.
7.2 THE STABILITY EQUATIONS

The time delays $t_1$ and $t_2$ are introduced in the differential equations of the inphase and quadrature components via Equations (6.10) - (6.19) in Chapter VI. We have seen that the associated characteristic equations are dependent on these delays and as a result, the time domain transient solutions exhibit exponential terms which are unbounded for certain values of $t_1$ and $t_2$.

The matrices which contain $t_1$ and $t_2$ explicitly are given by Equations (6.10) - (6.19) in Chapter VI. From the example treated in that chapter, it was found that these matrices contain terms such as $\cos \omega t_1$, $\cos \omega (t_1+t_2)$ and $\cos \omega t_1$ which can make the real part of the characteristic roots either positive or null.

In this section we need to elaborate further on the above matrices in order to establish the stability conditions. Note that the matrices of interest given by Equations (6.10) - (6.17) exhibit identical dependence on $t_1$ and $t_2$. Thus, for this investigation we will consider only one of these equations. The results will, however, apply to all of them.

For example, consider the matrix given by Equation (6.10) in the preceding chapter, that is

$$\phi_1 = [X(t-t_2)\cos \omega t]_L[X^T(t-t_1-t_2)\cos \omega (t-t_1-t_2)]_L$$  \hspace{1cm} (7.1)
Referring to Figure 6.1 of Chapter VI, it was already mentioned that the vector \(X(t-t_2)\) in Equation (7.1) is the array element input vector to the correlator mixer delayed by the correlator filter (group delay \(t_2\)). Likewise, the vector \(X(t-t_1-t_2)\) is the same input vector, but to the weight multiplier which is delayed by the output filter first (group delay \(t_1\)), then delayed further by the correlator filter (\(t_2\)).

The cosine term inside the left-hand side bracket in Equation (7.1) which is part of the vector demodulator undergoes no delay. The cosine term in the right-hand side bracket, which is part of the vector modulator, goes to the weight multiplier and hence undergoes delays \(t_1\) and \(t_2\).

In order to display more precisely the disturbances of \(t_1\) and \(t_2\) within Equation (7.1), assume that a desired frequency component present in the input vector \(X(t)\) has a radian frequency \(\omega_d\). Also assume that an interfering signal component of radian frequency \(\omega_I\) is present in the same vector. Then, the quantity \([X(t-t_2)\cos(\omega_d t)]_L\) will contain a phase term due to the desired signal, i.e.

\[
\psi_1 = (\omega_0 - \omega_d)t + \omega_d t_2
\] (7.2)

and a phase term due to the interfering signal, i.e.

\[
\psi_2 = (\omega_0 - \omega_I)t + \omega_I t_2
\] (7.3)

where only the difference frequency components are considered as implied by the subscript \(L\).
In the same manner, the quantity \[ X(t-t_1-t_2)\cos\omega_b(t-t_1-t_2) \] will contain the phase

\[ \psi_3 = (\omega_0-\omega_d)t + \omega_d(t_1+t_2) - \omega_b(t_1+t_2) \]  

(7.4)

associated with the desired signal, and the phase

\[ \psi_4 = (\omega_0-\omega_i)t + \omega_i(t_1+t_2) - \omega_b(t_1+t_2) \]  

(7.5)

associated with the interfering signal.

Now since Equation (7.1) represents a relatively constant time-average and since its quantities in brackets have the same central frequency for a given signal, the differences of the phases mentioned earlier should be taken in order to obtain a DC component in that equation.

Thus, when Equation (7.1) is evaluated with the phase quantities given by Equations (7.2) - (7.5), the following phase differences are obtained:

\[ \psi_1-\psi_3 = -\omega_d t_1 + \omega_0(t_1+t_2) \]  

(7.6)

\[ \psi_1-\psi_4 = \Delta\omega(t-t_2) - \omega_it_1 + \omega_0(t_1+t_2) \]  

(7.7)

\[ \psi_2-\psi_3 = -\Delta\omega(t-t_2) - \omega_d t_1 + \omega_0(t_1+t_2) \]  

(7.8)

\[ \psi_2-\psi_4 = -\omega_it_1 + \omega_0(t_1+t_2) \]  

(7.9)

where by definition we set

\[ \Delta\omega = \omega_l-\omega_d \]  

(7.10)

that is, the beat-frequency between the desired and interfering signals.
In the ideal situation of zero time delay \((t_1 = t_2 = 0)\), the above quantities are given by:

\[
\begin{align*}
\psi_1 - \psi_3 & = 0 \quad (7.11) \\
\psi_1 - \psi_4 & = \Delta \omega t \quad (7.12) \\
\psi_2 - \psi_3 & = -\Delta \omega t \quad (7.13) \\
\psi_2 - \psi_4 & = 0 \quad (7.14)
\end{align*}
\]

Equations \((7.6) - (7.9)\) represent the disturbances which affect the vector demodulator and modulator of Figure 6.1 given in Chapter VI, and which appear in the derivative of \(P(t)\) and \(Q(t)\), that is, just before the integrators.

The phase differences given by Equations \((7.7)\) and \((7.8)\) are the results of cross-correlations between the interference and the desired signal. The phase difference given by Equation \((7.9)\) is due to the correlation of the interference with itself. The amplitudes of the corresponding cosine and sine waveforms involve the interference signal amplitude, and as a result, the effects of such phase differences can be neglected at steady-state.

On the other hand, the phase difference given by Equation \((7.6)\) is the one that contributes primarily to the DC component of the vector demodulator, since it is the result of the correlation of the desired signal by itself.

In view of the ideal zero-delay situation suggested by Equations \((7.11) - (7.14)\), one would be inclined to add the phase quantity

\[
\Delta \psi_1 = \omega_0 t_1 - \omega_0 (t_1 + t_2) \quad (7.15)
\]
to the phase $\psi_1$ given by Equation (7.2) in order to make it equal to $\psi_3$, which is given by Equation (7.4) and thus satisfy Equation (7.11).

Similarly, one would add the phase quantity

$$\Delta \psi_2 = \omega_1 t_1 - \omega_0 (t_1 + t_2) \quad (7.16)$$

to the phase $\psi_2$ given by Equation (7.3) in order to satisfy Equation (7.14).

It has been implied above that the vector $X(t-t_2)$ and the cosine term in the quantity

$$Y(t) = [X(t-t_2) \cos \omega_0 t]_t \quad (7.17)$$

should be delayed accordingly in order to phase-match the other quantity given by

$$Z(t) = [X(t-t_2-t_1) \cos \omega_0 (t-t_1-t_2)]_t \quad (7.18)$$

From Equations (7.15) and (7.16) which are the phase differences needed to resolve the mismatch, it is clear that the vector $X(t-t_2)$ in Equation (7.17) should be delayed further by $t_1$ and that the cosine term in the same equation should be delayed by $(t_1 + t_2)$. Carrying these phase corrections, Equations (7.6) - (7.9) become:

$$\psi_1 - \psi_3 = 0 \quad (7.19)$$
$$\psi_1 - \psi_4 = \Delta \omega [t-(t_1+t_2)] \quad (7.20)$$
$$\psi_2 - \psi_3 = -\Delta \omega [t-(t_1+t_2)] \quad (7.21)$$
$$\psi_2 - \psi_4 = 0 \quad (7.22)$$

137
Introducing delays $t_3$ and $t_4$, we rewrite Equation (7.17) as

$$Y(t) = \left[ X(t-t_2-t_3)\cos \omega_0(t-t_4) \right]_L \quad (7.23)$$

and in order for Equation (7.23) to match Equation (7.18) we must have

$$t_3 = t_1 \quad (7.24)$$

and

$$t_4 = t_1 + t_2 \quad (7.25)$$

The above time delay compensations will then alleviate the phase discrepancies which affect the matrices given by Equations (6.10) - (6.17) in Chapter VI.

The locations of the time delay compensators $t_3$ and $t_4$ are shown in Figure 7.1. In order to compensate the effect of group delay $t_1$ of the output filter, the array input signal should be delayed by $t_3$ before it can be applied to the correlator mixer. Also, in order to compensate for both $t_1$ and $t_2$, the local oscillator waveform should be delayed by $(t_1 + t_2)$ in both the inphase and the quadrature branches of the vector demodulator.

A physical interpretation of the effect of $t_3$ and $t_4$ can be given by inspection of Figure 7.1. Time delay compensator $t_3$ allows maximum correlation by the bottom mixer; whereas $t_4$ allows maximum quadrature detection by the vector demodulator.

An alternate and probably more practical way of satisfying Equation (7.25), is that since the phase associated with $t_4$ is given by

$$\psi_0 = \omega_0 t_4 = \omega_0 (t_1 + t_2) \quad (7.26)$$
Figure 7.1 The LMS I.F. Array Loop Showing Time-Delay Compensators $t_3$ and $t_4$. 

139
where $\omega_0$ is the local oscillator radian frequency, it is possible to choose $(t_1 + t_2)$ to make this phase a multiple to $2\pi$, thus avoiding the need for a physical delay of value $t_4$. Hence, it can be required that

$$\omega_0(t_1 + t_2) = n \cdot 2\pi \quad (7.27)$$

or

$$t_1 + t_2 = nT_0 \quad (7.28)$$

where $T_0$ is the period of the local oscillator signal and $n$ an integer.

From a practical viewpoint, Equation (7.28) implies that some pure time delay should be intentionally added either in series with the output filter or in series with the correlator filter in order to make the sum of the delays satisfy that equation.

Summarizing the above findings, the time delay compensations can be achieved in two ways:

1) $t_3 = t_1 \quad (7.29)$
   and $t_4 = t_1 + t_2 \quad (7.30)$

or

2) $t_3 = t_1 \quad (7.31)$
   and $t_1 + t_2 = nT_0 \quad (7.32)$

Equations (7.29) - (7.30) or Equations (7.31) - (7.32) are the fundamental conditions for stability of the I.F. array feedback loop.
Thus far, the group delays $t_1$ and $t_2$ have been treated as pure time delays. In a real situation, however, they are subject to variations within the pass-band of the filters since they are by definition the slopes of the phase characteristics versus frequency. Thus, the time delay compensation technique introduced here can only be an approximation. For example, if delay lines are chosen as compensator devices for both $t_1$ and $t_2$, and the phase characteristics of the filters are not linear within their operating bandwidth, then the delay compensation will only be achieved for a single frequency. However, filters can be realized with a fairly linear phase response within the bandwidth of interest, so that a simple delay-line can be used to approximate their group delay.

An alternate method for compensating the effect of the output filter consists of using another filter in lieu of a delay line. This approach has been used successfully in the actual implementation of the I.F. array [10] in order to implement time delay $t_3$. A band-pass filter of group delay $t_3$ was introduced in the corresponding branch (see Figure 7.1) in order to match group delay $t_1$ of the output filter over its bandwidth. Although the slopes of their phase characteristics match only approximately, array stability is achieved.

On the other hand, in order to satisfy the other condition for stability, Equation (7.32) implies that compensation will be best achieved if the phase characteristics of both the output filter and the correlator filter are linear in the pass-bands. Therefore, in any case, filters with linear phase responses are desirable in order to meet the stability requirements.
For the correlator filter, whose bandwidth need not be as wide as that of the output filter, the problem is more manageable. Since in general the phase response is most linear around the center frequency, linearity can be reasonably assumed within a narrow bandwidth, thus justifying the assumption of group delay $t_2$ being a constant.

In the next section, the results of a computer simulation will be presented to show the merits of the time delay compensation schemes presented here.

### 7.3 COMPUTER SIMULATION OF TIME DELAY COMPENSATIONS

This section presents the results of a computer simulation which is intended to verify the array stability requirements. The results of this simulation are obtained for a two-element LMS I.F. array based on the single feedback loop given in Figure 7.1 which shows the pertinent time delays introduced in Section 7.2.

The desired input signal chosen for one part of the computer simulation is a CW signal bi-phase modulated by a 7-bit pseudo-noise code generated by means of a 3-bit shift register clocked at a rate of 2 KHz. The interference is a CW signal whose frequency coincides with the carrier of the desired signal.

The other specifications used for this simulation are the following:

- Amplitudes of desired and interfering signals : $S = I = 100$
- Angle of interfering signal arrival : $60^\circ$
- Desired signal carrier frequency : $f_d = 60$ KHz
- Interfering signal: CW signal with \( f_I = 60 \text{ KHz} \)
- Reference Signal: Equal to the shifted version of the desired signal with carrier at 40 KHz
- I.F. weight frequency: \( f_0 = 100 \text{ KHz} \)
- Loop gain: \( G = .2 \)
- Sampling period: \( T = .2 \mu s \)
- Output and correlator filters: 6 x 2nd-order Butterworth band-pass in cascade.
- Filter bandwidth: 16 KHz
- Output filter upper sideband attenuation: 62 dB
- Pseudo-noise code rate: 2 KHz
- Number of run periods: 5 (17500 \( \mu s \))

During this simulation, time delays are deliberately introduced according to Figure 7.1. For each loop situation, the inphase and quadrature components as well as the array output spectrum are plotted. The array input spectrum is also shown as reference for each case.

We recall that for given group delays \( t_1 \) and \( t_2 \), stability can be achieved either by having

\[
t_3 = t_1 \quad (7.33)
\]

and

\[
t_4 = t_1 + t_2 \quad (7.34)
\]

or by having

\[
t_3 = t_1 \quad (7.35)
\]

and

\[
t_1 + t_2 = nT_0 \quad (7.36)
\]

The results of this simulation are shown in Figures 7.2 through 7.13. Shown in Figure 7.2 is the array situation corresponding to zero.
a) Inphase and Quadrature Components

b) Input and Output Spectra

Figure 7.2 A Two-Element LMS I.F. Array: $t_1 = t_2 = 0 \mu$s.
time delays \((t_1 = t_2 = 0)\), thus providing a reference for stability, output signal-to-interference ratio (SIR) and output spectrum. For this stable case, the computed SIR is 36.383 dB.

Figures 7.3 - 7.6 consider situations with \(t_1\) alone, that is, with the output filter only. With \(t_1 = 12.5\) μs and \(t_2 = 0\), the inphase and quadrature components are unstable and the resulting output spectrum is meaningless as seen in Figure 7.3. Figure 7.4 shows that stability cannot be established by compensating with \(t_3\) only. Figure 7.5 depicts a stable situation which results from the application of the stability equations (7.33) and (7.34). Figure 7.5 shows the P's and Q's reaching steady-state. The output spectrum shown in Figure 7.5 is comparable to that of the zero-delay situation (Figure 7.2). The interfering signal is being nulled in that spectrum and the frequency component at 40 KHz is due primarily to the DC term present in the pseudo-noise signal \((100/4 = 7.143)\), which has a computed value of approximately 7.158. The array performance is also restored since the computed SIR is about 37.296 dB.

Figure 7.6 shows an alternate method to maintain stability, i.e. by application of Equations (7.35) and (7.36) when given the same situation, that is, when \(t_1 = 12.5\) μs. In this case, an extra time delay \(7.5\) μs is added in series with the output filter in order to satisfy Equation (7.36), given that \(T_0=10\) μs (period of weight signal). After Equation (7.36) is satisfied, the resulting value of \(t_1\) is used to satisfy Equation (7.35), i.e \(t_3=t_1=20\) μs. As seen in Figure 7.6, this situation corresponds to stable conditions; for this case, SIR= 37.251 dB, which compares with the zero-delay performance (SIR=36.383 dB).
Figure 7.3 A Two-Element LMS I.F. Array: $t_1 = 12.5 \mu s$. 

a) Inphase and Quadrature Components

b) Input and Output Spectra
a) Inphase and Quadrature Components

b) Input and Output Spectra

Figure 7.4 A Two-Element LMS I.F. Array: \( t_2 = 0; t_3 = t_1 = 12.5 \mu s \).
a) Inphase and Quadrature Components

b) Input and Output Spectra

Figure 7.5 A Two-Element LMS I.F. Array: \( t_3 = t_1 = 12.5 \mu s; t_4 = t_1 = 12.5 \mu s. \)
a) Inphase and Quadrature Components

b) Input and Output Spectra

Figure 7.6 A Two-Element LMS I.F. Array: $t_3 = t_1 = 12.5 + 7.5 = 20$ $\mu$s; $t_1 = 2T_0 = 20$ $\mu$s.
Figures 7.7-7.9 are related to situations involving time delay $t_2$ only. With $t_2 = 5 \mu s$ and $t_1 = 0$, the array is unstable, as seen in Figure 7.7. Applications of Equation (7.34) (Equation 7.33 is already satisfied since $t_3 = t_1 = 0$) yields the stable conditions of Figure 7.8. The resulting SIR is 36.976 dB. Application of the alternate method, i.e., Equation (7.36), leads to the situation shown in Figure 7.9. In this situation, stability is again achieved, and the resulting SIR is 36.521 dB.

Note that the steady-state values of the P's and Q's shown in Figures 7.8a and 7.9a are identical to those obtained with zero time delays (Figure 7.2).

Finally, Figures 7.10-7.13 illustrate situations where both $t_1$ and $t_2$ are present. Figure 7.10 shows an unstable situation which results when $t_1 = 12.5 \mu s$ and $t_2 = 8 \mu s$. Stabilization by means of Equations (7.33) and (7.34) leads to the results shown in Figure 7.11. The array yields a SIR of 36.808 dB. Figures 7.12 and 7.13 show two ways of satisfying Equation (7.36). In Figure 7.12, the extra time delay needed to make $(t_1 + t_2)$ a multiple of the period $T_0$ is added in series with the output filter, hence making $t_1 = 22 \mu s$; for this case, the SIR is 36.874 dB. Similarly, in Figure 7.13 the same value of additional time delay is added in series with the correlator filter instead, thus making $t_2 = 17.5 \mu s$; the corresponding output signal-to-interference ratio is $SIR = 36.976 \text{ dB}$.
Figure 7.7 A Two-Element LMS I.F. Array: $t_2 = 5\mu s$. 

a) Inphase and Quadrature Components  
b) Input and Output Spectra
Figure 7.8 A Two-Element LMS I.F. Array: $t_2 = 5 \mu s$; $t_4 = t_2 = 5 \mu s$. 

a) Inphase and Quadrature Components  
b) Input and Output Spectra
a) Inphase and Quadrature Components

b) Input and Output Spectra

Figure 7.9 A Two-Element LMS I.F. Array: \( t_2 = T_0 = 5+5 = 10 \mu s \).
a) Inphase and Quadrature Components

b) Input and Output Spectra

Figure 7.10 A Two-Element LMS I.F. Array: $t_1 = 12.5 \mu s; t_2 = 8 \mu s$. 
a) Inphase and Quadrature Components

b) Input and Output Spectra

Figure 7.11 A Two-Element LMS I.F. Array: \( t_1 = 12.5 \mu s; t_2 = 8 \mu s; t_3 = t_1 = 12.5 \mu s; t_4 = t_1 + t_2 = 20.5 \mu s. \)
Figure 7.12 A Two-Element LMS I.F. Array: \( t_1 = 12.5 + 9.5 = 22 \mu s; \) \( t_2 = 8 \mu s; \) \( t_3 = t_1 = 22 \mu s; \) \( t_1 + t_2 = 3T_0 = 30 \mu s. \)
a) Inphase and Quadrature components

Figure 7.13 A Two-Element LMS I.F. Array: $t_1 = 12.5$ μs; $t_2 = 8 + 9.5 = 17.5$ μs
$t_3 = t_1 = 12.5$ μs; $t_1 + t_2 = 3T_0 = 30$ μs.

b) Input and Output Spectra
The second part of the computer simulation is performed with CW signals only, because of the excessive computer CPU time otherwise required with pseudo-noise modulated signals. The goal is first to show the validity of Equations (7.33) - (7.36) for several values of time delays, and second to test the performance sensitivity as a function of these delays around stable points. The array input scenario is as shown in Figure 4.1 in Chapter IV, and the array parameters are identical to those used in Chapter VI with the exception that the adaptation time is set to 11 ms.

Figure 7.14a shows the results which apply to Equations (7.33) and (7.34) with \( t_1 \) and \( t_2 \) as varying parameters. Equations (7.33) and (7.34) are satisfied for each set of \( t_1 \) and \( t_2 \), and the corresponding signal-to-interference ratio is plotted. It can be seen that the array performance is preserved for all values of \( t_1 \) and \( t_2 \).

Likewise, Figure 7.14b shows results which apply to Equations (7.35) and (7.36). Here the quantity \( (t_1 + t_2) \) is such that \( (t_1 + t_2) = 3T_0 = 300 \) \( \mu \)s; \( t_1 \) is then used as the varying parameter. The signal-to-interference ratio is computed after Equations (7.35) and (7.36) are both satisfied.

The other point of interest is the array sensitivity as a function of \( t_1 \) and \( t_2 \) around stable states. The time delay compensation techniques considered here will deviate from their ideal forms in a practical implementation. Their application will involve alignment and compensation of physical components which is always a tedious task. Hence, deviations from stable conditions are likely to occur in a practical situation.
a) \[ t_3 = t_1 \]
\[ t_4 = t_1 + t_2 \]

b) \[ t_3 = t_1 \]
\[ t_1 + t_2 = 3T_0 = 300 \text{ µs} \]

Figure 7.14 Performance of a Two-Element LMS I.F. Array when the Stability Equations are Applied for Several Values of Group Delays.
The results of this part of the simulation are shown in Figure 7.15. Figures 7.15a and 7.15b consider the presence of \( t_1 \) only, i.e. \( t_1 = 80 \) \( \mu s \). Stability is restored by application of Equations (7.33) and (7.34), that is, \( t_3 = t_1 = t_4 = 80 \) \( \mu s \). Figure 7.15a shows the output SIR versus the quantity \( (t_3-t_1) \) with \( t_3 \) varying around 80 \( \mu s \). Figure 7.15b shows the case of \( t_4 \) varying around 80 \( \mu s \). In Figure 7.15c, only \( t_2 \) is considered. With \( t_2 = 80 \) \( \mu s \) = \( t_4 \), by virtue of Equation (7.34), the SIR is plotted versus \( (t_4-t_2) \) with \( t_4 \) varying around this stable point. Finally, in Figure 7.15d both \( t_1 \) and \( t_2 \) are included. Equation (7.35) is satisfied with \( t_3=t_1=100 \) \( \mu s \). The stable state dictated by Equation (7.36) is such that \( (t_1+t_2)=2T_0=200 \) \( \mu s \). Delay \( t_2 \) then varies around 100 \( \mu s \), thus showing the effect on SIR when the quantity \( (t_1+t_2) \) is not an integer multiple of the weight period \( T_0 \).

The curves of Figure 7.15 can also be considered as performance curves versus adaptation time. Since the array feedback loops can be stable for certain values of \( t_1 \) and \( t_2 \), the degradations experienced around stable points are due primarily to a longer adaptation time. However, for higher deviations from these states, the array performance degradation is due to effects of instability.

In summary, the I.F. array stability requirements were derived in this chapter when the differential equations given by Equations (6.8) and (6.9) include the group delays due to the output filter and the correlator filter. These requirements were then applied to a two-element array, and it was shown by computer simulation that they are sufficient to preserve stability, which was the main concern of this study.
Figure 7.15 A Two-Element LMS I.F. Array Performance Sensitivity Around Stable Points.
CHAPTER VIII
GENERAL GUIDELINES FOR A DIGITAL I.F. WEIGHT PROCESSOR

8.1 INTRODUCTION

This chapter presents a method which might be used to implement the I.F. array weight processor in digital form.

The motivation behind this attempt is due to some deficiencies associated with an all-analog implementation of the adaptive I.F. array. A four-element I.F. array based on the latter type of implementation has been recently constructed [10]. A block diagram that describes a single LMS I.F. loop which is used in this implementation is shown in Figure 8.1. The weight processor includes the I-Q (Inphase and Quadrature) detector, the two analog integrators and the vector modulator.

Even though this array functions properly under certain signal conditions, some deficiencies which result in performance degradation have been reported. Primarily, the problem of DC offsets still exists due to the introduction of the I-Q detector and the vector modulator. Both use mixers and therefore suffer the same problems generally associated with these devices. The DC offsets can be intolerable at the I-Q detector level where a reconversion to baseband takes place by mixing with the local oscillator (410 MHz for the actual implementation). This offset drives the integrators into saturation and
Figure 8.1 I.F.-Weighted Adaptive LMS Array Controller with Baseband Integrators.
is thus a major concern for the designer.

Also associated with the analog integrators is a possible drift due to capacitor discharge. This effect can lead to a complete array shut-off, particularly when processing low-level signals. For this reason, a digital implementation of the integrators would be more appropriate.

Thus, in order to overcome some of the difficulties due to the characteristics of physical components in an all-analog I.F. array, a partial digitization of the I.F. loop is proposed. As an external constraint, it was desired to have the array accommodate an input signal bandwidth of 40MHz (for an eventual use in satellite communications); consequently, only a partial digitization is attempted. The digitization applies to the I.F. weight processor only, whereas the correlator multiplier and the weight multiplier remain analog at this point.

8.2 GENERAL CONSIDERATIONS

As mentioned earlier, the problems of DC offsets and drifts due to analog mixers and analog integrators respectively, are a prime concern. Therefore, attention will be focused on these two problems in order to determine a possible method of reducing the adverse effects by digital means.

The goal of the digital processor is to generate a sinusoidal waveform having an amplitude and phase controlled by the correlator signal. Since the input signals to both sides of the correlator
have a bandwidth of 40 MHz, the sidebands at the correlator output have
twice that bandwidth or 80 MHz. However, all that is needed in a
sideband is the central component at the frequency of the weight signal.
This component contains the necessary information (amplitude and phase)
on the spatial distribution of interference, which results from the
correlation between the array input signal and the error signal.

If the lower sideband is selected at the array output (which is
likely in a communication channel), then the weight frequency is the
center of the upper sideband present at the correlator output. If
instead the upper sideband is retained at the array output, then the
weight frequency is the center of the lower sideband.

8.3 PREFILTERING AND FREQUENCY CONVERSION

Since the weight frequency (410 MHz was the value used in the
actual implementation) is generally higher than the practical range for
digital devices, a down-conversion in frequency is necessary before
further processing occurs. Before performing this step, a prefILTERING
operation is required on the correlator signal, since the local
oscillator frequency for the down-conversion is within the passband of
this signal (see Figure 8.2). It is this prefILTERING requirement that
led to the introduction of the correlator filter leading to the
stability conditions investigated in Chapters VI and VII. Methods to
achieve stability in the presence of this filter have been given in
Chapter VII. Hence, this filter can be incorporated in the I.F. array
implementation in order to reduce the correlator output signal
bandwidth.

There is a stringent requirement on the attenuation of this filter, however. As seen in Figure 8.2, the desired frequency for the weight is the difference between the correlator signal center frequency $f_C$ and the local oscillator frequency $f_L$, that is $f_0 = f_C - f_L$. However, if the component at $2f_L - f_C$ is not removed from the correlator signal before mixing, it would also give rise to a component of frequency $f_0$, hence disturbing the component of interest that is solely due to $f_C$. For this reason, the attenuation of the correlator filter at $2f_L - f_C$ should be a parameter to consider during the array implementation. An exact figure for this attenuation is subject to experimental considerations and hence will not be given here. However, it is expected that an attenuation of at least 60 dB would be required.

Also shown in Figure 8.2 is a pre-processing filter which follows the down-conversion mixer. Its function is to remove the sum-frequency term following the mixer and to further reduce the processing bandwidth if necessary. Since some low-pass filtering will follow the quadrature detector, the introduction of this filter is not critical.

However, this filter introduces a delay that should be taken into consideration during loop stabilization. Since this filter and the correlator filter are in the same path, their effects are additive. Therefore, the group delay $t_2$ introduced in Chapters VI and VII can be seen as a total delay due to both filters or to any other source of time delay before the vector demodulator.

The local oscillator frequency $f_L$ and the pre-processing filter
Figure 8.2 Prefiltering and Frequency Down-Conversion Requirements.
bandwidth $B_0$ are parameters that can be dictated by readily available
digital components and the desired speed of response of the array.
Again, since this is a problem to consider for every particular
implementation, it will not be discussed further here.

After the weight processor, an up-conversion in frequency should be
performed on the weight waveform in order to bring its frequency back to
the center frequency of the correlator output signal. This operation
will make the weight multiplication and the correlation compatible in
terms of the difference and sum frequencies needed to maintain the
correct phase around the loop. This up-conversion also requires
filtering. Thus, the same precautions should be taken as before
concerning appropriate attenuation before mixing. Therefore, a filter
is needed after the weight processor. Such a filter was not mentioned
during the stability analysis of Chapters VI and VII, although its
effect has been considered.

Thus, the stability conditions need to be updated accordingly. To
this end, let the delay due to this filter be denoted $t_5$. Then,
carrying a similar analysis to that of Chapter VI, one finds that $t_5$
appears in the cosine term in the right-hand side bracket of each of
Equations (6.10)-(6.17). Equation (6.10), for example, becomes

$$\phi_1 = [X(t-t_2)\cos\omega_o t]_L [X^T(t-t_1-t_2)\cos \omega_o (t-t_1-t_2-t_5)]_L. \quad (8.1)$$

By application of the stability analysis of Chapter VII, where the
compensating delays $t_3$ and $t_4$ have been defined, Equation (8.1) becomes

$$\phi_1 = [X(t-t_2-t_3)\cos\omega_o (t-t_4)]_L [X^T(t-t_1-t_2)\cos \omega_o (t-t_1-t_2-t_5)]_L \quad (8.2)$$
Therefore, the stability requirements which account for group delay $t_5$ are:

\[ t_3 = t_1 \]  \hspace{1cm} (8.3)

and

\[ t_4 = t_1 + t_2 + t_5 \]  \hspace{1cm} (8.4)

or

\[ t_3 = t_1 \]  \hspace{1cm} (8.5)

and

\[ t_1 + t_2 + t_5 = nT_0 \]  \hspace{1cm} (8.6)

where $T_0$ is the period of the local oscillator used for the vector demodulator, i.e. after the pre-processing filter.

By inspection of Equation (8.2) one can see that once the requirement for $t_1$ is satisfied, the perturbation due to $t_5$ is identical to that due to $t_2$. Hence, one can expect $t_2$ and $t_5$ to have similar effects on stability. This point has been verified by computer simulation as it has been done for $t_2$. The conclusions were identical to the case of $t_2$, and as a result, the curves obtained will be omitted.

The time delay $t_5$ introduced here should be understood as the total delay due to the post-processing filters or to any other source of time delay that can be present between the weight processor and the weight multiplier.

Finally, the frequency conversion discussed here is performed as a one-stage conversion (one stage down and one stage up). From a practical viewpoint, this might lead to stringent requirements on filter quality factor, filter bandwidth, speed of available digital components, etc. A remedy to such a situation would be to perform a two-stage
frequency conversion instead of a one-stage conversion, provided that care is taken regarding filtering and mixing with appropriate frequencies. The total time delay involved should also be evaluated at both ends of the weight processor in order to apply the time-compensation techniques.

8.4 DIGITAL WEIGHT PROCESSOR

8.4.1 General Description

The characteristic of the digital processor is the generation of the inphase and quadrature components in digital form. Once generated, these components are combined digitally. The result is D/A-converted and the resulting waveform is band-pass filtered to yield the I.F. weight at the desired frequency $f_0$.

The main distinctions that can be made between the analog and the digital processor are the following:

1. Both the vector demodulation and modulation are conceptually performed by means of switching with bipolar squarewaves at the fundamental frequency $f_0$.

2. The integration in each channel is performed as a digital accumulation.

The general principle of the overall processor is illustrated by the block diagram of Figure 8.3.

The input to this processor is the band-limited signal $s(t)$ resulting from the down-conversion of Figure 8.2. Assume that this signal is written as

170
\[ s(t) = A(t) \cos[\omega_0 t + \alpha(t)] \]  \hfill (8.7)

The locally generated quadrature waveforms \( C_I \) and \( C_Q \) can be expanded into cosine and sine terms, i.e.

\[ C_I = \sum_{n=0}^\infty a_n \cos n\omega_0 t \]  \hfill (8.8)

and

\[ C_Q = \sum_{n=0}^\infty b_n \sin n\omega_0 t \]  \hfill (8.9)

Demodulation of \( s(t) \) by means of \( C_I \) and \( C_Q \) yields

\[ s(t) * C_I = A(t) \cos[\omega_0 t + \alpha(t)] \cdot [a_1 \cos \omega_0 t + a_3 \cos 3\omega_0 t + \ldots] \]  \hfill (8.10)

\[ s(t) * C_Q = A(t) \cos[\omega_0 t + \alpha(t)] \cdot [b_1 \sin \omega_0 t + b_3 \sin 3\omega_0 t + \ldots]. \]  \hfill (8.11)

The low-pass filter shown in Figure 8.3 removes all but the frequency band which falls around DC for each channel. From Equations (8.10) and (8.11) these low frequency terms are

\[ [s(t) * C_I]_{LP} = \frac{a_1 A(t)}{2} \cos \alpha(t) \]  \hfill (8.12)

for the inphase component, and

\[ [s(t) * C_Q]_{LP} = \frac{-b_1 A(t)}{2} \sin \alpha(t) \]  \hfill (8.13)

for the quadrature component.

Each one of these bands is then sampled to allow a digital accumulation.

The resulting inphase \( (I_n) \) and quadrature \( (Q_n) \) components are combined using the same bipolar waveforms \( C_I \) and \( C_Q \) to form
Figure 8.3 General Principle of Vector Demodulation and Modulation by Bipolar Mixing.
\[ Y(t) = I_n \cdot C_I + Q_n \cdot C_Q \] (8.14)

which, after band-pass filtering becomes the I.F. weight

\[ W(t) = a_1 \cdot I_n \cdot \cos \omega_0 t + b_1 \cdot Q_n \cdot \sin \omega_0 t. \] (8.15)

In fact \( a_1 = b_1 \), hence

\[ W(t) = a_1 \sqrt{I_n^2 + Q_n^2} \cdot \cos \left[ \omega_0 t - \tan^{-1} \frac{Q_n}{I_n} \right] \] (8.16)

with \( \omega_0 = 2\pi f_0 \).

The following two sections consider the practical design aspects of both the vector demodulator and modulator.

8.4.2. Vector Demodulator

The intrinsic feature of the vector demodulator (and modulator) shown in Figure 8.3 is the by-passing of the analog multiplication which usually relies on the nonlinear characteristic of analog mixers. Instead, by allowing the locally generated quadrature waveforms to be squarewaves, a switching operation can be performed on the pre-filtered signal \( s(t) \) (Figure 8.3).

However, the multiplication by a bipolar waveform which results in \( +s(t) \) during half of the period \( T_0 \) and in \(-s(t)\) during the other half, would require a sign inversion during that last half-period.

In order to avoid an analog sign inversion, for which case a constant gain will be required over the pre-filtering bandwidth \( B_0 \), this sign reversal can be done digitally instead.

The digital principle equivalent to the bipolar mixing method is illustrated in Figure 8.4 (this principle is equivalent to what is known
in the literature as the chopper demodulation technique). For example, in order to generate the inphase component $I_n$, the signal $s(t)$ with center frequency $f_0=1/T_0$ is sampled every $T_0$ seconds according to the periodic waveform $D_I$ (Figure 8.4). It is also sampled $T_0/2$ later according to waveform $\overline{D_I}$. The samples due to $D_I$ are then inverted by simple sign-bit inversion after the corresponding A/D converter. These samples are added digitally to the ones due to $D_I$ and the result is stored in a digital accumulator. This operation is thus equivalent to multiplying $s(t)$ by the bipolar waveform $C_I$ (Figure 8.3), low-pass filtering and sampling at twice the carrier frequency $f_0$.

The block diagram which shows the implementation of this principle as a vector demodulator is shown in Figure 8.5.

In order to generate the quadrature component $Q_n$, the signal $s(t)$ is first sampled according to the periodic pulse train $D_Q$ (Figure 8.5) which is in quadrature with $D_I$ (see timing diagram of Figure 8.7). It is also sampled according to $\overline{D_Q}$ which is waveform $D_Q$ delayed by $T_0/2$. Again, after sign-bit inversion the samples are accumulated to yield $Q_n$.

Figure 8.5 also shows the integrators whose function is now realized as a digital accumulation.

8.4.3 Vector Modulator

According to the principle depicted in Figure 8.3, the vector modulator multiplies the inphase and quadrature components $I_n$ and $Q_n$ by the local bipolar squarewaves $C_I$ and $C_Q$, and combines the result into a periodic waveform (at steady-state) which, after proper filtering,
Figure 8.4 Double Sampling with Sign Reversal.
Figure 8.5 Digital Vector Demodulator by Quadrature Sampling and Integration by Digital Accumulation.
renders the I.F. weight at the fundamental frequency $f_o$.

In order to simplify the modulator design, the control waveforms $C_I$ and $C_Q$ can be made to switch between 0 and 1 instead of between -1 and +1 (a multiplication by -1 would require additional gating and another sign inversion for proper mixing). The DC component which results from the unipolar waveforms will be removed by the band-pass filter which follows the switching logic. However, if the presence of this component becomes critical because of available devices, then the bipolar switching should be fully implemented.

The simplicity of the modulator design relies on the fact that since $I_n$ and $Q_n$ are available in digital form, the periodic multiplication by +1 can be implemented as a gating operation. Similarly, if the bipolar multiplication is used instead, the additional multiplication by -1 can be implemented identically, after sign-bit inversion has been completed.

For the unipolar multiplication, the gating scheme shown in Figure 8.6 is suggested.

The components $I_n$ and $Q_n$, resulting from the digital accumulation of Figure 8.5, are gated ON and OFF toward the digital adder according to the switching functions denoted $M_I$ and $M_Q$ (Figure 8.6), which are in quadrature. The timing of these waveforms should be commensurate with the demodulator waveforms $D_I$ and $D_Q$ (Figure 8.5). The timing diagram illustrating this requirement, and the analog waveform which results before filtering are shown in Figure 8.7. The time relationship between $D_I$ and $M_I$ is such that the sampling pulses, which are represented as the leading edges of waveform $D_I$, occur in the middle of each half-period of
Figure 8.6 Switching Logic as a Vector Modulator.
Figure 8.7 Timing Diagram for Demodulation and Modulation Control Waveforms.
waveform $M_1$. The same relationship holds between $D_Q$ and $M_Q$.

Under the timing condition of Figure 8.7, the digital adder performs the following sequence of additions: $I_n+0$, $I_n+Q_n$, $Q_n+0$ and $0+0$. (Had the bipolar modulation been performed instead, the adder sequence would have been: $I_n-Q_n$, $I_n+Q_n$, $Q_n-I_n$, $-I_n-Q_n$.) This sequence repeats periodically, and at steady-state the resulting output $Y(t)$ from the D/A-converter becomes periodic. Band-pass filtering at the fundamental frequency will result in the analog I.F. weight with frequency $f_0$.

An up-conversion in frequency will then take this weight to the proper frequency which matches the correlator output center frequency.

8.4.4 Computer Simulation

To investigate the merit of the digital weight processor proposed in this chapter, a computer simulation was done on a two-element LMS array with CW input signals.

Both the vector demodulator and modulator were simulated in accordance with their respective block diagrams of Figures 8.5 and 8.6, and with the timing diagram of Figure 8.7. The band-pass filter was simulated as a 12th-order digital filter (6xsecond-order filters in cascade) with a 2KHz bandwidth.
The other specifications used in the simulation are the following:

- Desired signal: \( S=100; f_d=6\text{kHz}; \theta_d=0^\circ \)
- Interference: \( SI=100; f_I=6.4\text{kHz}; \theta_I=60^\circ \)
- Reference signal: \( R=100; f_p=4\text{kHz} \)
- Weight signal: \( f_o=10\text{KHz} \)
- Loop gain: \( G=10^{-5} \)
- Sampling period: \( T=1\mu s \)
- Output filter upper sideband attenuation: \( n=51.6\text{dB} \)
- Adaptation time: 30ms.

The result of the computer simulation is shown in Figure 8.8. Figures 8.8a and 8.8b represent the last periods of the analog weights after adaptation, for the first and second feedback loop respectively. Figure 8.8c represents the resulting error signal during and after adaptation.

For an input signal-to-interference (SIR) ratio of 0dB, the output SIR which results with this processor is SIR=47.183 dB.

### 8.4.5 Dynamic Range of the Weight Processor

From a design viewpoint, the dynamic range is an important parameter since it is a measure of the signal swing that can be handled by the processor. For a digital processor, it is a necessary design parameter for the choice of the proper A/D (Analog-to-Digital) converter.
Figure 8.8 A Two-Element LMS Array with Digitally Generated I.F. Weights.
The intrinsic ability of the processor to detect the smallest signal-level is primarily dictated by the desired dynamic range, and by thermal noise present in the circuitry. On the other hand, the highest signal-level is limited by the maximum power that can be delivered by the analog components (correlator mixer).

The dynamic range requirement determines the accuracy of the A/D converter to be used, and therefore the number of bits needed to represent a digital sample. However, thermal noise usually puts a lower bound on the smallest detectable signal. For this reason, 95% of the noise should be substantially less than the magnitude associated with the LSB (Least Significant Bit) in order to use the A/D to its full potential.

For a signal with full-scale voltage FS and an A/D convertor with n-bit accuracy, the dynamic range DR is expressed as

\[ DR = 20 \log_{10} \frac{FS}{2^n} = 20 \log_{10} 2^n. \]  \hspace{1cm} (8.17)

Numerically we have:

for n=8 bits, \( DR = 48.2 \)dB
for n=10 bits, \( DR = 60.2 \)dB
for n=12 bits, \( DR = 72.2 \)dB.

In the recent all-analog implementation of a four-element adaptive array [10], the average dynamic range was around 65dB, and a maximum of 73dB was able to be achieved under optimal adjustments.
In this analog version of the array, the dynamic range limitation is primarily due to the analog mixers used for the weight and correlator multipliers. Hence, if these multipliers remain analog (because of the constraint to receive broadband signals) they determine the dynamic range for the digital processor.

Thus, in order to maintain a dynamic range of at least 70dB around the loops of a hybrid version of the adaptive array (that is, an array with analog multipliers but digital weight processor), a 12-bit A/D converter would be required for the weight processor. A higher accuracy would introduce additional constraints related to permitted speed of conversion, thermal noise level and cost.

8.4.6 Advantages of Digitizing

During the design of the digital processor, the emphasis has been on eliminating signal feedthrough which is associated with analog multipliers.

In addition to complying with this requirement, the digital processor has other desirable features as follows:

1. No integrator drift since the inphase and quadrature components are now available in digital accumulators. This might be found useful if these stored weights are needed for further development in the overall adaptive system.

2. Possibility of DC offset subtraction, if any, before integration.
3. Precise local quadrature squarewave generation. This operation is limited only by the precision of the master clock generator and uniformity of gate risetimes.

4. Processing at lower frequencies reduces the potential feedthrough that might be partly due to radiation effects otherwise possible with high frequencies.

Although not emphasized here, digital techniques are preferred because they are suitable for a high level of future integration (VLSI). For this reason, the trend would be to digitize, when possible, in order to facilitate a future VLSI implementation of the adaptive processor. In addition, digital VLSI can be combined to great advantage with thick-film hybrid techniques for future realizations of adaptive antennas.

One possible limitation of the proposed digital weight processor resides in the choice of the sampling frequency $f_0$. The minimum value of $f_0$ is limited by the smallest array adaptation time, and its maximum by the speed of the available A/D converter. The larger the array input signal, the shorter the adaptation time (the array time constants are inversely proportional to signal power). Therefore, the choice of $f_0$ should be based on the strongest signal received. If the minimum adaptation time is denoted by $t_A$, then the minimum $f_0$ is such that

$$t_A \gg \frac{T_0}{2} \quad (8.18)$$

where $f_0=1/T_0$. 185
With the condition of Equation (8.18), enough time is allowed to perform the integration by several accumulations in accordance with the double sampling method presented in Section 8.4.

On the other hand, the maximum sampling frequency is set be the speed of the A/D converter which is chosen to fulfill a certain dynamic range requirement; it is also set, at a lower level, by the speed of the D/A (Digital-to-Analog) converter which is used in the vector modulator and which should be four times faster than the A/D according to the timing diagram of Figure 8.7.

Another limitation, which is intrinsic to any digital processor, is due to the usual quantization effects. These are the input quantization errors due to the discrete representation of signals during sampling (A/D errors), the round-off or truncation errors committed during add operations (accumulation), and finally the D/A conversion errors. However, it has been shown [19] that the LMS algorithm is extremely tolerant of amplitude and phase errors in the weights. This property and the advantages mentioned earlier, make the digitizing of the adaptive array weight processor worth considering.
The purpose of this study was to develop a method of generating weights for the LMS adaptive array at a non-zero frequency by digital means, in order to improve upon the array with weights at baseband.

This task was not trivial because of stability problems that arise from hardware implementation of the adaptive array when its weights are at an intermediate frequency. Despite this problem, this task was shown to be feasible. A theoretical analysis of the I.F. array was given, and a possible digital weight processor was proposed.

Because of the constraint put on signal bandwidth, only a partial digitization of the array was considered. Thus, the correlation of the input signal with the error signal, as well as the weighting operation, were assumed in analog form.

It was shown, by reviewing the baseband array, that a critical factor in the implementation of the LMS algorithm is the signal-by-error multiplication. It is this critical point that was behind the motivation of introducing I.F. weights and of seeking digital methods.

For the I.F. array to operate optimally, it was shown that it is necessary to close the feedback loop with a single frequency sideband. Hence, it became a necessity to remove one sideband from the array.
output. The output band-pass filter became a characteristic of the I.F. array.

In order to apply the LMS algorithm in an optimal manner, it was then required to include the effect of this filter in the derivation of the differential equations of the weights. By characterizing the output filter with its upper sideband attenuation coefficient, it was shown, by solving for the steady-state solution, that the array performance was a critical function of this parameter.

On the other hand, the need to process the feedback control digitally introduces a prefiltering requirement at the output of the signal-by-error multiplier. The filter introduced in this branch, as well as the output filter, were shown to be primary sources of instability in the I.F. array.

The effects of these two filters were investigated by introducing their group delays into the differential equations of the inphase and quadrature components of the weights. Time-compensation methods were deduced from these equations, and their effectiveness in stabilizing the I.F. feedback loops was demonstrated.

A digital weight processor that can allow the generation of the inphase and quadrature components as digital quantities was proposed. In designing this processor, the focus has been on two main objectives:

1. Elimination of DC offsets and feedthrough from the structure.
2. Storage of the weighting components in digital form.

Offsets and feedthrough due to reconversion to baseband are
conceptually nonexistent in the proposed system which relies on sampling but not on the usual baseband mixing operation.

The additional motivation for digital techniques is that they are suitable for possible VLSI implementation, which would be highly desirable for satellite applications where weight is always a limiting factor.

The computer simulations conducted during this study appear to verify the feasibility of the intermediate frequency weighting by means of the proposed implementation.

Future studies should include a complete experimental implementation of the system in order to confirm the validity of the computer model. The stability criteria which have also been confirmed should provide a sufficient basis for the design of a properly functioning experimental system.
BIBLIOGRAPHY


[44] "Multiplier-Accumulators Parallel 8,12, or 16 bits", TRW LSI Products, P.O. Box 2472, La Jolla, CA 92038, 1979-1981.


193

APPENDIX

This appendix contains the main computer programs developed for the purpose of the simulations related to this study.

195
PROGRAM 1
============================================
G PROGRAM 1
C  =============================================
G = A TWO-ELEMENT LMS ARRAY WITH IF WEIGHTING =
C  =============================================
G
G ------------------------------------------------------
G  - ANALOG IMPLEMENTATION
G - EFFECT OF GROUP DELAYS ON I-Q COMPONENTS
G ------------------------------------------------------
G
G
G
G
G
DOUBLE PRECISION X1D,X2D,RS,YD(4),
1Y1(4),Y(4),YOUTD(22003),YOUT(22003), YOUT(22003),
2ERR(22003),G01(4),G02(4),
3Y1(5,4),Y2(5,4),
4C11,C12,C21,C22,
5G11(22003),G12(22003),G21(22003),G22(22003),
6W1,W2,S,S1,R,THETAD,THETA1,FD,FI,F0,T,G,PI,DT,
7GAMMA1,GAMMAI,BW1,BW2,BW3,YIM,YIM,OSIPR,
8YDMAX,YIMAX,BW4,CF1(4),
9CF2(4),T1,T2,T3,T4,T5
DIMENSION TM(22003),YP(22003)
C S=AMPLITUDE OF DESIRED SIGNAL
S=100.D00
C SI=AMPLITUDE OF INTERFERING SIGNAL.
SI=100.D00
C R=AMPLITUDE OF REFERENCE SIGNAL.
R=100.D00
C THETAD=ANGLE OF INCOMING DESIRED SIGNAL (DEGREES).
THETAD=0.D00
C THETA1=ANGLE OF INCOMING INTERFERING SIGNAL (DEGREES).
C = THETA1=60.D00
C FD=FREQUENCY OF INPUT DESIRED SIGNAL(HZ).
FD=6000.D00
C FI=FREQUENCY OF INTERFERING SIGNAL(HZ).
FI=6400.D00
C FR=FREQUENCY OF REFERENCE SIGNAL(HZ).
FR=4090.D00
C T0=PERIOD OF WEIGHT SIGNAL (T0=100 MICROSECONDS; F0=10KHZ).
C T=SAMPLING INTERVAL(SEC.)
T=1.D-06
C TM=TIME VARIABLE.
C G=LOOP GAIN.
G=.1D-06
G PI=3.1415926545D00
C INITIAL CONDITIONS .
C C1(3)=0.D00
C G01(3)=0.D00
C C2(3)=0.D00
C G02(3)=0.D00
C
C
C
DO 600 I=1,3
YD(I)=0.D00
600
Y1(1)=0.D00
Y(I)=0.D00
CO1(I)=0.D00
CO2(I)=0.D00
CF1(I)=0.D00
CF2(I)=0.D00

CONTINUE
DO 601 I=1,3
  Y1(I,1)=0.D00
  Y2(I,1)=0.D00
CONTINUE

CONTINUE

C WEIGHTS GENERATION.
C ===============

C FILTERS GROUP DELAY AND COMPENSATING TIME DELAYS (SEC.).
T1=0.D-06
T2=50.D-06
T3=0.D-06
T4=0.D-06

C GAMMAA=PI*DSIN(THETAD*(P1/180.D00))
C GAMMAI=PI*DSIN(THETAI*(P1/180.D00))
DO 2 K=1,22000
  DT=DFLOAT(K-1)*T
  CALL SIGNS(PI,FD,DT,FI,GAMMA,GI,S1,R,FR,X1D,X2D)
  CALL SIGNS(PI,FD,DT,FI,GAMMA,GI,S1,R,FR,X1D,X2D)
C ARRAY OUTPUT DESIRED SIGNAL ONLY (BPF AT 4 KHZ)—YOUTD.
FR=4000.D00
BW1=4000.D00
CALL BPF(BW1,K,1,YD(4),YD(2),YOUTD(K+3),YOUTD(K+2),YOUTD(K+1),
  SFR,T,P1,Y1,Y2)
C ARRAY OUTPUT INTERFERENCE SIGNAL ONLY (BPF AT 4 KHZ)—YOUTI.
FR=4000.D00
BW2=4000.D00
CALL BPF(BW2,K,2,YI(4),YI(2),YOUTI(K+3),YOUTI(K+2),YOUTI(K+1),
  SFR,T,P1,Y1,Y2)
C ARRAY OUTPUT (SIGNAL+INTERFERENCE)—BPF AT 4 KHZ—YOUT.
FR=4000.D00
BW3=4000.D00
CALL BPF(BW3,K,3,Y(4),Y(2),YOUT(K+3),YOUT(K+2),YOUT(K+1),FR,T,
  P1,Y1,Y2)
C ERROR SIGNAL.
ERR(K+3)=RS-YOUT(K+3)

C CORRELATORS OUTPUT.
F0=10000.D00
CO1(4)=CXID*ERR(K+3)
CO2(4)=CXY2D*ERR(K+3)
BW4=4000.D00
CALL BPF(BW4,K,4,CO1(4),CO1(2),CF1(4),CF1(3),CF1(2),
  IF0,T,P1,Y1,Y2)
CALL BPF(BW4,K,5,CO2(4),CO2(2),CF2(4),CF2(3),CF2(2),
  IF0,T,P1,Y1,Y2)
C11=CF1(4)*DCOS(2.D00*PI*F0*(DT-T4))
C12=CF1(4)*DSIN(2.D00*PI*F0*(DT-T4))

197
C \[ C_{21} = \cos(2 \cdot 0.00 \cdot \pi \cdot f_0 \cdot (D_T - T_4)) \]
\[ C_{22} = \sin(2 \cdot 0.00 \cdot \pi \cdot f_0 \cdot (D_T - T_4)) \]

C INTEGRATION: CI=INPHASE-CQ=QUADRATURE.

C
\[ C_{11}(K+3) = C_{11}(K+2) + C_{11} \]
\[ C_{01}(K+3) = C_{01}(K+2) + C_{01} \]
\[ C_{12}(K+3) = C_{12}(K+2) + C_{12} \]
\[ C_{02}(K+3) = C_{02}(K+2) + C_{02} \]

C W1=WEIGHT FOR LOOP 1 - W2=WEIGHT FOR LOOP 2.

C
\[ W_1 = C_{11}(K+3) \cos(2 \cdot 0.00 \cdot \pi \cdot f_0 \cdot (D_T - T_1 - T_2 - T_5)) + C_{01}(K+3) \sin(2 \cdot 0.00 \cdot \pi \cdot f_0 \cdot (D_T - T_1 - T_2 - T_5)) \]
\[ W_2 = C_{12}(K+3) \cos(2 \cdot 0.00 \cdot \pi \cdot f_0 \cdot (D_T - T_1 - T_2 - T_5)) + C_{02}(K+3) \sin(2 \cdot 0.00 \cdot \pi \cdot f_0 \cdot (D_T - T_1 - T_2 - T_5)) \]

C
D0 603 I=1,3
YD(I)=YD(I+1)
Y1(I)=Y1(I+1)
Y2(I)=Y2(I+1)
C01(I)=C01(I+1)
C02(I)=C02(I+1)
CF1(I)=CF1(I+1)
CF2(I)=CF2(I+1)
603 CONTINUE
D0 604 L=1,5
D0 605 I=1,3
Y1(L,I)=Y1(L,I+1)
Y2(L,I)=Y2(L,I+1)
605 CONTINUE
604 CONTINUE
2 CONTINUE
C
C OUTPUT DESIRED SIGNAL MAX AMPLITUDE YDM.
YDM=DABS(YOUTD(20500))
D0 6 J=20560,22003
IF(YDM.GE.DABS(YOUTD(J))) GO TO 6
YDM=DABS(YOUTD(J))
6 CONTINUE

C YDMAX=DABS(YOUTD(1))
D0 400 J=1,22003
IF(YDMAX.GE.DABS(YOUTD(J))) GO TO 400
YDMAX=DABS(YOUTD(J))
400 CONTINUE
C
C OUTPUT INTERFERENCE MAX AMPLITUDE YIM.
YIN=DABS(YOUTI(20500))
D0 7 J=20530,22009
IF(YIN.GE.DABS(YOUTI(J))) GO TO 7
YIN=DABS(YOUTI(J))
7 CONTINUE
CONTINUE

OUTPUT SIGNAL-TO-INTERFERENCE RATIO.

PRINT 10, G, S, SI, R, THETAD, THETA1
FORMAT(3X,2HC=,E6.1,3X,2HS=,F6.1,3X,2HR=,F6.1,
3X,7HTHETAD=,F6.1,3X,7HTHETA1=,F6.1)
PRINT 20, YDM
FORMAT(3X,4HYDM=,D9.3)
PRINT 402, YDMAX
FORMAT(3X,6HYDMAX=,D9.3)
PRINT 8, OSIPR
FORMAT(3X, 'OSIPR=', F9.3, ' DB')
PRINT 80
FORMAT(10X, 'ERROR SIGNAL')
PRINT 90, ('ERR(K+3),K=21500,22000')
PRINT 120, (2X, D9.3))
PRINT 360
FORMAT(10X, 'C(K-3)')
PRINT 370, (C(K-3), K=21500,22000)
FORMAT(12(2X, D9.3))
PRINT 380
FORMAT(10X, 'CQ1(K-3)')
PRINT 390, (CQ1(K-3), K=21500,22000)
FORMAT(12(2X, D9.3))
PRINT 391
FORMAT(10X, 'CQ2(K-3)')
PRINT 392, (CQ2(K-3), K=21500,22000)
FORMAT(12(2X, D9.3))
DO 500 I=1,22003
TH(I)=FLOAT(1-I-1)
PRINT 500, 1=1,22003
TN(I)=FLOAT(1-I)
DO 501 YP(I)=GQ2(I)
CALL LINE(TM,0.,400.,YP,0.,6.3333,22003,0,0)
CALL AXIS(0,400,0.4,TH,0,6.3333,22003,0,0)
CALL SYM30L(-1.12,-.56,.56,'OUADRATURE',90.,2)
CALL SYMB0L(4.,4.,.56,'INPHASE',0.,10)
CALL SYM30L(-1.68,-1.68,.56,-50',0.,3)
CALL SYMB0L(54.88,-1.68,.56,-50',0.,3)
CALL SYMB0L(-1.68,-8.,.56,-50',0.,3)
CALL SYMB0L(-1.12,-8.,.56,'TIME AXIS-MICROSEC',0.,18)
CALL SYMB0L(54.88,-1.68,.56,'22400',0.,5)
CALL SYMB0L(-1.68,-8.,.56,'50',0.,3)
CALL SYMB0L(-1.12,-8.,.56,'50',0.,2)
DO 501 I=1,22003
CALL SYMBOL(.56,-1.68,.56,'0',0.,1)
CALL SYMBOL(24.,-2.24,.56,'TIME AXIS-MICROSEC',0.,18)
CALL SYMBOL(54.88,-1.68,.56,'22400',0.,5)
CALL SYMBOL(-2.24,-8.,.56,'-165',0.,4)
CALL SYMBOL(-1.68,8.,.56,'-165',0.,3)
DO 502 I=1,22003

502
YP(I)=CI2(I)
CALL LINE(TM,0.,.400.,YP,0.,20.6333,22003,0,0)
CALL PLOT(0.,18.,-3)
CALL AXIS(0.,0.,' ',-1,-56.,0.,0.,400.,8.)
CALL AXIS(0.,-8.,' ',1,-16.,90.,-.2,.025,4.)
CALL SYMBOL(-1.12,-.56,.56,' PI',90.,2)
CALL SYMBOL(4.,4.,.56,' QUADRATURE',0.,10)
CALL SYMBOL(.56,-1.68,.56,'0',0.,1)
CALL SYMBOL(24.,-2.24,.56,' TIME AXIS-MICROSEC',0.,18)
CALL SYMBOL(54.88,-1.68,.56,'22400',0.,5)
CALL SYMBOL(-1.68,-8.,.56,'-15',0.,3)
CALL SYMBOL(-1.12,8.,.56,'15',0.,2)
DO 503 I=1,22003

503
YP(I)=CQ1(I)
CALL LINE(TM,0.,.400.,YP,0.,1.875,22003,0,0)
CALL PLOT(0.,18.,-3)
CALL AXIS(0.,0.,' ',-1,-56.,0.,0.,400.,8.)
CALL AXIS(0.,-8.,' ',1,-16.,90.,-.1,.125,4.)
CALL SYMBOL(-1.12,-.56,.56,' INPHASE',0.,7)
CALL SYMBOL(4.,4.,.56,' QUADRATURE',0.,10)
CALL SYMBOL(.56,-1.68,.56,'0',0.,1)
CALL SYMBOL(24.,-2.24,.56,' TIME AXIS-MICROSEC',0.,18)
CALL SYMBOL(54.88,-1.68,.56,'22400',0.,5)
CALL SYMBOL(-1.68,-8.,.56,'-15',0.,3)
CALL SYMBOL(-1.12,8.,.56,'15',0.,2)
DO 504 I=1,22003

504
YP(I)=C11(I)
CALL LINE(TM,0.,.400.,YP,0.,.875,22003,0,0)
CALL PLOT
STOP
END

C SUBROUTINES.

======

C 1 SIGNALS EVALUATION.

------------

C I INPUT SIGNAL FROM ELEMENT 1.

XID=S*DCOS(2.D00*FD*PI*(DT-T2-T3))+
1 S1*DCOS(2.D00*PI*FD*(DT-T2-T3))

C I INPUT SIGNAL FROM ELEMENT 2.

X2D=S*DCOS(2.D00*FD*PI*(DT-T2-T3)-GAMMA)+
1 S1*DCOS(2.D00*PI*FD*(DT-T2-T3)-GAMMA)

C REFERENCE SIGNAL.

RS=R*DCOS(2.D00*FD*PI*(DT-T2))
C WEIGHTED OUTPUT.
C DESIRED SIGNAL ONLY.
YD=W1*S*DCOS(2.000*PI*FD*(DT-T1-T2)) +
1 W2*S*DCOS(2.000*PI*FD*(DT-T1-T2) - GAMMA)D
C INTERFERENCE SIGNAL ONLY.
YI=W1*S*DCOS(2.000*PI*FI*(DT-T1-T2)) + W2*S*I*
2 DCOS(2.000*PI*FI*(DT-T1-T2) - GAMMA)I
C DESIRED SIGNAL + INTERFERENCE.
Y=YD+YI
RETURN
END
C
C 2 BANDPASS FILTERING-6TH ORDER.
C
SUBROUTINE BPF(BW,K,L,X1,X2,Y03,Y02,Y01,F0,T,PI,Y1,Y2)
DOUBLE PRECISION Y1(5,4),Y2(5,4),BW,X1,X2,Y03,Y02,Y01
1 ,F0,T,PI,D,E
D=DCOTAN(PI*T*BW)
E=2.000*DCOS(2.000*PI*T*FI)
Y1(L,4)=(X1-X2+(1.000*D)*Y1(L,2)+D*E*Y1(L,3))/(1.000+D)
1 Y2(L,4)=(Y1(L,4)-Y1(L,2)+(1.000-D)*Y2(L,2)+D*E*)
2 Y2(L,3))/(1.000+D)
Y03=(Y2(L,4)-Y2(L,2)+(1.000-D)*Y01+D*E*Y02)/(1.000+D)
RETURN
END

201
PROGRAM 2

DOUBLE PRECISION X1D,X2D,RS,YD(4),
1Y1(4),Y(4),YOUT(11003),YOUT1(11003),YOUT(11003),
2ERR(11003),C1(4),C2(4),
3Y1(5,4),Y2(5,4),
4C11,C12,C21,C22,
5C11(11003),C12(11003),C1(11003),C22(11003),
6U1,W2,S,SL,THETAD,THETA1,FD,FI,FR,F0,T,C,DT,
7GAMAD,GAMMA1,BW1,BW2,BW3,WM,YIM,
8SYMAX,YMAX,BW4,CP1(4),OS1PR(29),
9CF2(4),T1,T2,T3,T4,T5,TT
DIMENSION T(N)(29),YP(29)

C S=AMPLITUDE OF DESIRED SIGNAL.
S=100.D00
C SI=AMPLITUDE OF INTERFERING SIGNAL.
SI=100.D00
C R=AMPLITUDE OF REFERENCE SIGNAL.
R=100.D00
C THETAD=ANGLE OF INCOMING DESIRED SIGNAL (DEGREES).
THETAD=0.D00
C THETA1=ANGLE OF INCOMING INTERFERING SIGNAL (DEGREES).
THETA1=60.D00
C FD=FREQUENCY OF INPUT DESIRED SIGNAL (HZ).
FD=6000.D00
C FI=FREQUENCY OF INTERFERING SIGNAL (HZ).
FI=6400.D00
C FR=FREQUENCY OF REFERENCE SIGNAL (HZ).
FR=4000.D00
C T0=PERIOD OF WEIGHT SIGNAL (T0=100 MICROSECONDS; F0=10KHZ).
T0=1.D-06
C T=SAVING INTERVAL (SEC.).
T=1.D-06
C TM=TIME VARIABLE.
C G=LOOP GAIN.
G=.1D-06
PI=3.1415926545D00
C
DO 1 N=1,29
TT=DFLOAT(N-1)*10.D-06
T1=TT
T2=TT
T3=0.D00
T4=0.D00
T5=0.D00
C INITIAL CONDITIONS.
W1=0.D00
W2=0.D00
C
C
C11(3)=0.D00
C01(3)=0.D00
C12(3)=0.D00
C02(3)=0.D00
C
DO 600 I=1,4
  YOUTD(I)=0.000
  YOUTI(I)=0.D00
  YD(I)=0.D00
  YI(I)=0.D00
  Y(I)=0.D00
  CO1(I)=0.D00
  CO2(I)=0.D00
  CF1(I)=0.D00
  CF2(I)=0.D00
600  CONTINUE
C
DO 601 L=1,5
  DO 602 I=1,4
    Y1(L,I)=0.D00
    Y2(L,I)=0.D00
  602  CONTINUE
  601  CONTINUE
C
C WEIGHTS GENERATION.
C = = = = = = = = = = = = = =
C
GAMMAD=PI*DSIN(THETAD*PI/180.000 >
GAMMAI=PI*DSIN(THETAI*PI/180.000 >
DO 2 K=1.11000
  DT=DFLOAT(K-1)*T
  CALL SIGNSCPI ,FD,DT,FI,GAMMAD,GAMMAI,S,S,R,FR,X1D,X2D
S,RS,YD(4),Y1(4),W1,W2,Y(4),T1,T2,T3>
C ARRAY OUTPUT DESIRED SIGNAL ONLY (BPF AT 4 KHZ) - YOUTD.
FR=4000.000
BW1=4000.D00
CALL BPF(BW1,K,1,YD(4),YD(2),YOUTD(K+3),YOUTD(K+2),YOUTD(K+1),
  SFR,T,PI,Y1,Y2)
C
C ARRAY OUTPUT INTERFERENCE SIGNAL ONLY (BPF AT 4 KHZ) - YOUTI.
FR=4000.000
BW2=4000.D00
CALL BPF(BW2,K,2,Y1(4),Y1(2),YOUTI(K+3),YOUTI(K+2),YOUTI(K+1),
  SFR,T,PI,Y1,Y2)
C
C ARRAY OUTPUT (SIGNAL+INTERFERENCE) - BPF AT 4 KHZ - YOUT.
FR=4000
BW3=4000.D00
CALL BPF(BW3,K,3,Y(4),Y(2),YOUT(K+3),YOUT(K+2),YOUT(K+1),FR,T,
  PI,Y1,Y2)
C
C ERROR SIGNAL.
ERR(K+3)=RS-YOUT(K+3)
C
CO1(4)=C*X1D*ERR(K+3)
CO2(4)=C*X2D*ERR(K+3)
C
203
BIV=4000.D00
CALL BPF(BIV,K,4,C01(4),C01(2),CF1(4),CF1(3),CF1(2),
IF0,T,P1,Y1,Y2)
CALL BPF(BIV,K,5,C02(4),C02(2),CF2(4),CF2(3),CF2(2),
IF0,T,P1,Y1,Y2)
C11=CF1(4)*DCOS(2.D00*PI*F0*(T4-T))
C12=CF1(4)*DSIN(2.D00*PI*F0*(T4-T))
C21=CF2(4)*DCOS(2.D00*PI*F0*(T4-T))
C22=CF2(4)*DSIN(2.D00*PI*F0*(T4-T))
C11(K+3)=C11(K+2)+C11
CQ1(K+3)=CQ1(K+2)+C12
C12(K+3)=C12(K+2)+C21
CQ2(K+3)=CQ2(K+2)+C22
W1=C11(K+3)*DCOS(2.D00*PI*F0*(T1-T2-T5))+CQ1(K+3)*
1DSIN(2.D00*PI*F0*(T1-T2-T5))
W2=C12(K+3)*DCOS(2.D00*PI*F0*(T1-T2-T5))+CQ2(K+3)*
1DSIN(2.D00*PI*F0*(T1-T2-T5))
DO 663 I=1,3
YD(I)=YD(I+1)
YI(I)=YI(I+1)
CO1(I)=CO1(I+1)
CO2(I)=CO2(I+1)
CF1(I)=CF1(I+1)
CF2(I)=CF2(I+1)
663 CONTINUE
DO 664 L=1,5
DO 665 I=1,3
Y(L,I)=Y(L+1,I+1)
665 CONTINUE
664 CONTINUE
CONTINUE
C OUTPUT DESIRED SIGNAL MAX AMPLITUDE YDM.
YDM=DABS(YOUTD(10500))
DO 6 J=10500,11003
IF(YDM.GE.DABS(YOUTD(J))) GO TO 6
YDM=DABS(YOUTD(J))
6 CONTINUE
C OUTPUT INTERFERENCE MAX AMPLITUDE YIM.
YIM=DABS(YOUTI(10500))
DO 7 J=10500,11003
IF(YIM.GE.DABS(YOUTI(J))) GO TO 7
YIM=DABS(YOUTI(J))
7 CONTINUE
C OUTPUT SIGNAL-TO-INTERFERENCE POWER RATIO-OSIPR(N)-DB.
IF(YIM.EQ.0.D00) GO TO 8
OSIPR(N)=20.D00*DLOG10(YDM/YIM)
GO TO 1
OSIPR(N)=999.D00
CONTINUE

PRINT 10,G,S,S1,R,THETAD,THETAI
10 FORMAT(3X,2HG=,E6.1,3X,2HS=,F6.1,3X,3HSI=,F6.1,3X,2HR=,F6.1,
83X,7HTHETAD=,F6.1,3X,7HTHETAI=,F6.1)
PRINT 20,(K,OSIPR(K),K=1,29)
20 FORMAT(2X,'OSIPR(',12,* ) = ',F9.3,' DB')

DO 500 I=1,29
TM(I)=10.*FLOAT(I-1)
500 CONTINUE
DO 501 I = 1 , 29
YP(I)=OSIPR(I)
501 CONTINUE
CALL PLOTE2
STOP
END

C SUBROUTINES. 
C = = = = = = = = = = =
C 1 SIGNALS EVALUATION. 
C --------
C SUBROUTINE SIGNS(P1,FD,DT,FI,GAMMAD,GAMMA1,S,S1,R,FR,X1D,X2D,RS,
1YD,YI,W1,W2,Y,X1D,X2D,T1,T2,T3)
DO DOUBLE PRECISION P1,FD,DT,FI,GAMMAD,GAMMA1,S,S1,R,FR,X1,X2,RS,
1YD,Y1,Y2,Y,X1D,X2D,T1,T2,T3
C INPUT SIGNAL FROM ELEMENT 1.
X1D=S*DCOS(2.D00*P1*FD*(DT-T2-T3))+
1  S1*DCOS(2.D00*P1*FI*(DT-T2-T3))
C INPUT SIGNAL FROM ELEMENT 2.
X2D=S*DCOS(2.D00*P1*FD*(DT-T2-T3)-GAMMAD)+
1  S1*DCOS(2.D00*P1*FI*(DT-T2-T3)-GAMMA1)
C REFERENCE SIGNAL
RS=R*DCOS(2.D00*P1*FR*(DT-T2))

205
C WEIGHTED OUTPUT.
C DESIRED SIGNAL ONLY.
\[ Y_1 = W_1 S \ast \cos(2 \pi F_D (D_T - T_1 - T_2)) + \]
\[ W_2 S \ast \cos(2 \pi F_D (D_T - T_1 - T_2) - \gamma_D) \]
C INTERFERENCE SIGNAL ONLY.
\[ Y_1 = W_1 S \ast \cos(2 \pi F_I (D_T - T_1 - T_2)) + W_2 S \ast \]
\[ \cos(2 \pi F_I (D_T - T_1 - T_2) - \gamma_I) \]
C DESIRED SIGNAL + INTERFERENCE.
\[ Y = Y_D + Y_I \]
RETURN
END

C 2 BANDPASS FILTERING-6TH ORDER.

SUBROUTINE BPF(BW,K,L,X1,X2,Y03,Y02,Y01,F0,T,P1,Y1,Y2)
DOUBLE PRECISION Y1(5,4),Y2(5,4),BW,X1,X2,Y03,Y02,Y01
D=DCOTAN(PI*T*BW)
E=2.D00*DGOS(2.D00*PI*T*F0)
Y1(L,4)=(X1-X2+(1.D00-D)*Y1(L,2)+D*E*Y1(L,3))/(1.D00+D)
Y2(L,4)=(Y1(L,4)-Y1(L,2)+(1.D00-D)*Y2(L,2)/+D*E*Y2(L,3))
Y03=(Y2(L,4)-Y1(L,4))/D
RETURN
END
PROGRAM 3

= A TWO-ELEMENT LMS ARRAY WITH IF WEIGHTING =

DOUBLE PRECISION X1D, X2D, RS, YD(4),
  Y1(4), Y2(4), YOUTD(4), YOUTI(4), YOUT(4),
  Y2R(4), Y2H(4), Y3(5,4), Y4(5,4), Y5(5,4),
  C01(4), C02(4),
  S1, S2, R, THETAD, THETA1, FD, FI, FR, FO, T, G, PI, DT,
  GW, GW1, GW2, BW1, BW2, BW3, BW4, BW5, BW6,
  GAMMAI, C01, C02, C03, C04,
  6W1, W2, S1, R, THETAD, THETA1, FD, FI, FR, FO, T, G, PI, DT,
  G, GW, GW1, GW2, BW1, BW2, BW3, BW4, BW5, BW6,
  GAMMAI, C01, C02, C03, C04,
  G, GW, GW1, GW2, BW1, BW2, BW3, BW4, BW5, BW6,
  GAMMAI, C01, C02, C03, C04,
  G, GW, GW1, GW2, BW1, BW2, BW3, BW4, BW5, BW6,
  GAMMAI, C01, C02, C03, C04,
  G, GW, GW1, GW2, BW1, BW2, BW3, BW4, BW5, BW6,
  GAMMAI, C01, C02, C03, C04,
  G, GW, GW1, GW2, BW1, BW2, BW3, BW4, BW5, BW6,
  GAMMAI, C01, C02, C03, C04,
  G, GW, GW1, GW2, BW1, BW2, BW3, BW4, BW5, BW6,
  GAMMAI, C01, C02, C03, C04,
  G, GW, GW1, GW2, BW1, BW2, BW3, BW4, BW5, BW6,
  GAMMAI, C01, C02, C03, C04,
  G, GW, GW1, GW2, BW1, BW2, BW3, BW4, BW5, BW6,
  GAMMAI, C01, C02, C03, C04,
  G, GW, GW1, GW2, BW1, BW2, BW3, BW4, BW5, BW6,
  GAMMAI, C01, C02, C03, C04,
  G, GW, GW1, GW2, BW1, BW2, BW3, BW4, BW5, BW6,
  GAMMAI, C01, C02, C03, C04,
  G, GW, GW1, GW2, BW1, BW2, BW3, BW4, BW5, BW6,
  GAMMAI, C01, C02, C03, C04,
  G, GW, GW1, GW2, BW1, BW2, BW3, BW4, BW5, BW6,
  GAMMAI, C01, C02, C03, C04,
  G, GW, GW1, GW2, BW1, BW2, BW3, BW4, BW5, BW6,
  GAMMAI, C01, C02, C03, C04,
  G, GW, GW1, GW2, BW1, BW2, BW3, BW4, BW5, BW6,
  GAMMAI, C01, C02, C03, C04,
  G, GW, GW1, GW2, BW1, BW2, BW3, BW4, BW5, BW6,
  GAMMAI, C01, C02, C03, C04,
  G, GW, GW1, GW2, BW1, BW2, BW3, BW4, BW5, BW6,
  GAMMAI, C01, C02, C03, C04,
  G, GW, GW1, GW2, BW1, BW2, BW3, BW4, BW5, BW6,
  GAMMAI, C01, C02, C03, C04,
  G, GW, GW1, GW2, BW1, BW2, BW3, BW4, BW5, BW6,
  GAMMAI, C01, C02, C03, C04,
  G, GW, GW1, GW2, BW1, BW2, BW3, BW4, BW5, BW6,
  GAMMAI, C01, C02, C03, C04,
  G, GW, GW1, GW2, BW1, BW2, BW3, BW4, BW5, BW6,
  GAMMAI, C01, C02, C03, C04,
  G, GW, GW1, GW2, BW1, BW2, BW3, BW4, BW5, BW6,
  GAMMAI, C01, C02, C03, C04,
  G, GW, GW1, GW2, BW1, BW2, BW3, BW4, BW5, BW6,
  GAMMAI, C01, C02, C03, C04,
  G, GW, GW1, GW2, BW1, BW2, BW3, BW4, BW5, BW6,
  GAMMAI, C01, C02, C03, C04,
  G, GW, GW1, GW2, BW1, BW2, BW3, BW4, BW5, BW6,
  GAMMAI, C01, C02, C03, C04,
  G, GW, GW1, GW2, BW1, BW2, BW3, BW4, BW5, BW6,
CI1=0.D00
CQ1=0.D00
CI2=0.D00
CQ2=0.D00

DO 600 I=1,4
   YOUTD(I)=0.D00
   YOUTI(I)=0.D00
   YD(I)=0.D00
   YI(I)=0.D00
   CO1(I)=0.D00
   CO2(I)=0.D00
   CF1(I)=0.D00
   CF2(I)=0.D00
600 CONTINUE

DO 601 L=1,5
   DO 602 J=1,4
       Y1(L,J)=0.D00
       Y2(L,J)=0.D00
       Y3(L,J)=0.D00
       Y4(L,J)=0.D00
   602 CONTINUE
601 CONTINUE

GAMMAO=PI*0.5*SIN(THHTAO*PI/180.000)
GAMMAI=PI*0.5*COS(THETAI*PI/180.000)
YDM=0.000
YIM=0.D00
PN=1.D00

DRAW AXIS.

CALL PLOT(.7,2.25,-3)
CALL AXIS(0.,0.,'0,-7,0,0,0,200,3.5)
CALL SYMBOI(.5,.14,'QUADRATURE',0,10)
CALL SYMBOI(-.14,-.07,.14,'Q2',90.2)
CALL SYMBOI(.07,-.21,.14,'0',.1)
CALL SYMBOI(.3,-.28,.14,'TIME-MICROSEC',.0,13)
CALL SYMBOI(.679,-.21,.14,'17500',.0,5)
CALL SYMBOI(-.49,-1.07,.14,'-4',.0,3)
CALL SYMBOI(-.35,.93,.14,'-1.2',.0,4)
CALL PLOT(0,2.25,-3)
CALL AXIS(0.,0.,'0,-7,0,0,0,200,3.5)
CALL AXIS(0.,0.,'-1','0,-2,90.,-1.2,1.2,1.)
CALL SYMBOI(.5,.14,'INPHASE',.0,7)
CALL SYMBOI(-.14,-.07,.14,'P2',90.2)
CALL SYMBOI(.07,-.21,.14,'9',.0,1)
CALL SYMBOI(.3,-.28,.14,'TIME-MICROSEC',.0,.13)
CALL SYMBOI(.679,-.21,.14,'17500',.0,5)
CALL SYMBOI(-.63,-1.07,.14,'-1.2',.0,4)
CALL SYMBOL(-.49, -.93, -.14, '1.2', 0., 3.)
CALL PLOT(0., .225, -.3)
CALL AXIS(0., 0.,'0.', 0., 0., 0., 0., 0., 200., 3.5)
CALL AXIS(0., -.1.,'0.', 0., 2., 90., -.1., 1.1)
CALL SYMBOL(5., 5., 0., 'QUADRATURE', 0., 10.)
CALL SYMBOL(-.14, -.07, 14., 'Q1', 90., 2.)
CALL SYMBOL(.07, -.21, 14., '0', 0., .1)
CALL SYMBOL(.04, -.26, 14., 'TIME-MICROSEC', 0., 13)
CALL SYMBOL(.79, -.21, 14., 'T17500', 0., 5)
CALL SYMBOL(-.49, -.07, 14., '1', 0., .3)
CALL SYMBOL(0., .25, -.2)
CALL PLOT(0., .225, -.3)
CALL AXIS(0., 0.,'0.', 0., 0., 0., 0., 200., 3.5)
CALL AXIS(0., -.1.,'0.', 0., 2., 90., -.1., 1.1)
CALL SYMBOL(.5, 5., 0., 'INPHASE', 0., 7.)
CALL SYMBOL(-.14, -.07, 14., 'PI', 90., 2.)
CALL SYMBOL(.07, -.21, 14., '0', 0., .1)
CALL SYMBOL(6.79, -.21, 14., 'TIME-MICROSEC', 0., 13)
CALL SYMBOL(-.63, -.14, -.12, '1.2', 0., 4.)
CALL SYMBOL(-.49, -.93, -.14, '1.2', 0., 3.)
CALL PLOT(0., -.75, -.3)

C C C C C C

C WEIGHTS GENERATION.
C = = = ==
C
DO 1 L=1,5
   N1=L+(L-1)*7
   N2=L*7
   L2=2+(L-1)*7
   L3=3+(L-1)*7
   L5=5+(L-1)*7
   DO 3 N=N1,N2
      K1=1+(N-1)*2500
      K2=N*2500
      IF ((N.EQ.12).OR.(N.EQ.L3).OR.(N.EQ.L5)) GO TO 4
   GO TO 5
1   CONTINUE
2   CONTINUE
DO 2 K=KI,K2
   DT=DFLOAT(K-1)*T
   CALL SIGNSIPI,FD,DT,FIX,R,GAMMA,S,SI,R,FR,X1D,X2D
   RS,YS,YD(4),Y1(4),Y2(4),T1,T2,T3,PY
   CALL OUTPUT DESIRED SIGNAL ONLY(BPF AT 40 KHZ)-YOUTD.
   FR=40000.D0
   BW1=40000.D0
   CALL BPF(BW1,1,YS,YD(4),YD(2),YOUTD(4),YOUTD(3),YOUTD(2),
   SRF,T,P1,Y1,Y2,Y3,Y4,Y5)
   CALL ARRAY OUTPUT INTERFERENCE SIGNAL ONLY(BPF AT 40 KHZ)-YOUTI.
   FR=40000.D0
   BW2=40000.D0
   CALL BPF(BW2,2,Y1(4),YI(2),YOUT1(4),YOUT1(3),YOUT1(2),
   SRF,T,P1,Y1,Y2,Y3,Y4,Y5)
   CALL ARRAY OUTPUT(SIGNAL+INTERFERENCE)-BPF AT 40 KHZ-YOUT.
   FR=40000.D0
   BW3=40000.D0
   CALL BPF(BW3,3,Y4,Y2,YOUT4,YOUT3,YOUT2(2),FR,T,
   1P1,Y1,Y2,Y3,Y4,Y5)

C C ERROR SIGNAL.
   ERR=RS-YOUT(4)
C C CORRELATORS OUTPUT.
   C

C C C C C C
F0=100000.D00
COl(4)=G*X1D*ERR
CO2(4)=G*X2D*ERR
BW4=490000.D00
CALL BFF(BW4,4,COl(4),COl(2),CFI(4),CF1(3),CF1(2),
1F0,T,PI,Y1,Y2,Y3,Y4,Y5)
CALL BFF(BW4,5,CO2(4),CO2(2),CF2(4),CF2(3),CF2(2),
1F0,T,PI,Y1,Y2,Y3,Y4,Y5)
C11=CFI(4)*DCOS(2.D00*PI*F0*(DT-T4))
C12=CFI(4)*DSIN(2.D00*PI*F0*(DT-T4))
C21=CF2(4)*DCOS(2.D00*PI*F0*(DT-T4))
C22=CF2(4)*DSIN(2.D00*PI*F0*(DT-T4))
P1=C02
P2=C12
P3=C01
P4=C11
SP1=1.2
SP2=1.2
SQ2=1.2
P1=P1/SQ2
P2=P2/SQ2
P3=P3/SP1
P4=P4/SP1

C INTEGRATION.

C C11=C11+T*C11
C COl=C01+T*C0l
C C12=C12+T*C21
C C02=C02+T*C22
W1=C11*DCOS(2.D00*PI*F0*(DT-T1-T2-T5))
1DS1N(2.D00*PI*F0*(DT-T1-T2-T5))
W2=C02*DCOS(2.D00*PI*F0*(DT-T1-T2-T5))
1DS2N(2.D00*PI*F0*(DT-T1-T2-T5))

C XP(K)=X1D
YD1(K)=Y0UT(4)
P11=C11
P01=C01
P12=C12
P02=C02

C PLOT 1-Q COMPONENTS.

C TM1=FLOAT(K-1)/12500.
TM2=FLOAT(K)/12500.
P11=P11/SP1
P01=P01/SP1
P12=P12/SP2
P02=P02/SP2
CALL PLOT(TM1,P1,1)
CALL PLOT(TM2,P02,2)
CALL PLOT(0.,2,25.,-3)
CALL PLOT(TM1,P2,1)
CALL PLOT(TM2,P12,2)
CALL PLOT(0.,2,25.,-3)
CALL PLOT(TM1,P3,1)

210
CALL PLOTE2
CALL PLOTE2
CALL PLOTE2
CALL PLOTE2
CALL PLOTE2
C
DO 603 I=1,3
   YOUTD(I)=YOUTD(I+1)
   YOUTI(I)=YOUTI(I+1)
   YD(I)=YD(I+1)
   YI(I)=YI(I+1)
   CO1(I)=CO1(I+1)
   CO2(I)=CO2(I+1)
   CF1(I)=CF1(I+1)
   CF2(I)=CF2(I+1)
603 CONTINUE
DO 604 N=1,5
   DO 605 I=1,3
      Y1(N,I)=Y1(N,I+1)
      Y2(N,I)=Y2(N,I+1)
      Y3(N,I)=Y3(N,I+1)
      Y4(N,I)=Y4(N,I+1)
      Y5(N,I)=Y5(N,I+1)
605 CONTINUE
604 CONTINUE
C
IF(K.LT.87300) GO TO 2
C
C OUTPUT DESIRED SIGNAL MAX AMPLITUDE YDN.
   IF(YDN.GE.DABS(YOUTD(4))) GO TO 6
   YDN=DABS(YOUTD(4))
C
C OUTPUT INTERFERENCE MAX AMPLITUDE YIM.
   IF(YIN.GE.DABS(YOUTI(4))) GO TO 7
   YIM=DABS(YOUTI(4))
C
C OUTPUT SIGNAL-TO-INTERFERENCE POWER RATIO OSIPR(N)-DB.
   IF(YIM.EQ.0.D00) GO TO 8
   OSIPR=20.D00+DLLOG10(YDN/YIM)
   GO TO 2
   8
   OSIPR=999.D00
C
CONTINUE
C
   IF(PN.LT.0.D00) PN=-PN
1 CONTINUE
C
CALL PLOTE2
C
C
PRINT 10, C, S, SI, R, THETA&D, THETA1
10 FORMAT(3X, 2HEG, =, E6.1, 3X, 2HS, =, F6.1, 3X, 2HER, =, F6.1, 3X, 2HSRI, =, F6.1, 3X, 2HER, =, F6.1, 3X, THTHETA&D, =, F6.1, 3X, THTHETA1, =, F6.1)
PRINT 20, OSIPR
20 FORMAT(2X, ' OSIPR', '=', F9.3, ', DB')
PRINT 700, CI1, CQ1, CI2
700 FORMAT(///, 10X, ' CI1 = ', D9.3///, 10X, ' CQ1 = ', D9.3///, 10X, ' CI2 = ', D9.3///)
C
C FAST FOURIER TRANSFORM.
DO 101 I = 1, 17500
P(I) = XP(I + 70000)
DI(I) = YDI(I + 70000)
101 CONTINUE
N = 17500
CALL FFTG(P, N, 1WK, WK)
CALL FFTCC(DI, N, 1WK, WK)
C
C SPECTRUM EVALUATION.
DO 26 I = 1, 57
K = 112 + I
L = 112 + I
SX = P(K)
SY = DI(L)
SXX(I) = (CABS(SX))/FLOAT(17500)
SYY(I) = (CABS(SY))/FLOAT(17500)
26 CONTINUE
C
C DO 300 I = 1, 57
J = 182 + I
PRINT 100, J, SXX(I)
100 FORMAT(10X, ' SXX(', I3, ',') = ', F9.3)
300 CONTINUE
C
C PLOT SPECTRA.
DO 28 I = 1, 57
SX = P(17291)
SY = DI(17361)
SXXX = (CABS(SX))/FLOAT(17500)
SYYY = (CABS(SY))/FLOAT(17500)
PRINT 400, SXXX, SYYY
400 FORMAT(///, 10X, 'SXX(17291) = ', F9.3///, 10X, 'SYY(17361) = ', F9.3)
28  F(I) = FLOAT(I - 1)*2.7.
DELTA_F = 8.73.

C
CALL PLOT(.5, .5, -3)
CALL AXIS(.5, .5, 'FREQUENCY-KHZ', -13.6, .0, 32., DELTA_F, .75)
CALL AXIS(.5, .0, 'OUTPUT SPECTRUM SYY', 19.4, .0, 15.1.)
DO 29 I = 1, 57
   XV = F(I)/DELTA_F
   YV = SYY(I)/15.
   CALL PLOT(XV, 0., 3)
29  CALL PLOT(XV, YV, 2)
DO 30 I = 1, 57
   XV = F(I)/DELTA_F
   YV = SXX(I)/15.
   CALL PLOT(XV, 0., 3)
30  CALL PLOT(XV, YV, 2)
CALL PLOTE2
STOP
END
C
C SUBROUTINES.
C
C I SIGNALS EVALUATION.
C
SUBROUTINE SIGNSIPI , FD , DT , FI , GAMMA_D , GAMMA_I , S , SI , R , FR , XI , X2D , RS,
   Y1 , Y1 , W2 , Y , T1 , T2 , T3 , PN
DOUBLE PRECISION PI , FD , DT , FI , GAMMA_D , GAMMA_I , S , SI , R , FR , XI , X2 , RS,
   Y1 , Y1 , W2 , Y1 , X1D , X2D , T1 , T2 , T3 , PN
C INPUT SIGNAL FROM ELEMENT 1.
XID = PI*S*DCOS(2*D00*PI*FD*(DT-T2-T3)) +
SI*DCOS(2*D00*PI*FI*(DT-T2-T3))
C INPUT SIGNAL FROM ELEMENT 2.
X2D = PI*S*DCOS(2*D00*PI*FD*(DT-T2-T3)-GAMMA_D) +
SI*DCOS(2*D00*PI*FI*(DT-T2-T3)-GAMMA_I)
C REFERENCE SIGNAL
RS = PI*R*DCOS(2*D00*PI*FR*(DT-T2))
C WEIGHTED OUTPUT.
C DESIRED SIGNAL ONLY.
YD = WI*PN*SS*DCOS(2*D00*PI*FD*(DT-T1-T2)) +
W2*SI*DCOS(2*D00*PI*FI*(DT-T1-T2)-GAMMA_D)
C INTERFERENCE SIGNAL ONLY.
YI = WI*SI*DCOS(2*D00*PI*FI*(DT-T1-T2)) + W2*SI*
DCOS(2*D00*PI*FI*(DT-T1-T2)-GAMMA_I)
C DESIRED SIGNAL + INTERFERENCE.
Y = YD + YI
RETURN
END
C 2 BANDPASS FILTERING-12TH ORDER.

SUBROUTINE BPF(BW,L,X1,X2,Y03,Y02,Y01,F0,T,PI,Y1,Y2,Y3,Y4,Y5)
DOUBLE PRECISION Y1(5,4),Y2(5,4),BW,X1,X2,Y03,Y02,Y01
F0,T,PI,D,E,Y3(5,4),Y4(5,4),Y5(5,4)
D=DCOTAN (PI*T*BW)
E=2.D00*DCOS(2.D00*PI*T*F0)
Y1(L,4)=(X1-X2+(1.D00-D)*Y1(L,2)+D*E*Y1(L,3))/(1.D00+D)
Y2(L,4)=(Y1(L,4)-Y1(L,2)+(1.D00-D)*Y2(L,2)+D*E*Y2(L,3))/(1.D00+D)
Y3(L,4)=(Y2(L,4)-Y2(L,2)+(1.D00-D)*Y3(L,2)+D*E*Y3(L,3))/(1.D00+D)
Y4(L,4)=(Y3(L,4)-Y3(L,2)+(1.D00-D)*Y4(L,2)+D*E*Y4(L,3))/(1.D00+D)
Y5(L,4)=(Y4(L,4)-Y4(L,2)+(1.D00-D)*Y5(L,2)+D*E*Y5(L,3))/(1.D00+D)
Y03=(Y5(L,4)-Y5(L,2)+(1.D00-D)*Y01+D*E*Y02)/(1.D00+D)
RETURN
END
PROGRAM 4

========================================
= A TWO-ELEMENT LMS ARRAY WITH IF WEIGHTING =
========================================

-----------------------------------------------------------------------------------------------------------------
- ANALOG IMPLEMENTATION
- COMPUTATION OF CHARACTERISTIC ROOTS
-----------------------------------------------------------------------------------------------------------------

REAL A(5), DRI(21), D1I(21), DRI2(21), D1I2(21), DRI3(21), D1I3(21)
REAL DRI4(21), D1I4(21), TM(21)
DIMENSION IE(21)
COMPLEX Z(4)

ATDB=51.6
AT=10.*ATDB/20.
P1=3.1415926545
GAMA=P1*SIN(P1/3.)
T2=0.
DO 1 I=1,21
T1=5.*((I-1)*1.E-06
ALZ=2.*P1*1000.*(T1+T2)
ALD=2.*P1*6000.*T1
ALI=2.*P1*6400.*T1
CND=COS(ALD-ALZ)
CPD=COS(ALD+ALZ)
CPI=COS(ALI+ALZ)
CMI=COS(ALI-ALZ)
SMI=SIN(ALI-ALZ)
SPI=SIN(ALI+ALZ)

A1=25.*((CND+CMI+CPD+CPI)/AT)
A2=25.*((CND+COS(ALI-ALZ+GAMA)+CPD+Cos(ALI+ALZ+GAMA))/AT)
A3=25.*((CND+COS(AL1-ALZ-GAMA)+CPD+COS(AL1+ALZ-GAMA))/AT)
B1=25.*((SMI-SPI)-(SPD+SPI)/AT)
B2=25.*((SMI+SPI)-(SPD-SPI)/AT)

E0=10000.*((A2**2+B2**2)*(A3**2+B3**2)+B1**2*(2.*A2*A3+
8 B1**2-2.*B2*B3))
E1=-4000.*B1*(A1**2+B1**2+B2**2)
E2=200.*(A2**2+B1**2+B2**2)
H2=600.*A1**2+B2**2+E2
H3=40.*A1

A(1)=1.
A(2)=H3
A(3)=H2
A(4)=H1
A(5)=H0
C ZPOLR = SUBROUTINE FOR ZEROS OF A POLYNOMIAL.
CALL ZPOLR(A,4,Z,IER)
DR1(I) = REAL(Z(I))
DI1(I) = AIMAG(Z(I))
DR2(I) = REAL(Z(2))
DI2(I) = AIMAG(Z(2))
DR3(I) = REAL(Z(3))
DI3(I) = AIMAG(Z(3))
DR4(I) = REAL(Z(4))
DI4(I) = AIMAG(Z(4))

TM(I) = 5.*((I-1))
IE(I) = IER

1 CONTINUE
PRINT 10,H0,H1,H2,H3
10 FORMAT(I0X,E10.4,/,/)

PRINT 20,(I,Z(I),I=1,4)
20 FORMAT(10X,11.5X,E10.4,2X,E10.4,/,/)

PRINT 30,(I,IE(I),I=1,21)
30 FORMAT(10X,E10.4)

PRINT 40,(K,DR1(K),K=1,21)
40 FORMAT(10X,'DR1(',12,')= ',E10.4)

PRINT 50
PRINT 60,(K,DI1(K),K=1,21)
60 FORMAT(10X,'DI1(',12,')= ',E10.4)

PRINT 70,(K,DR2(K),K=1,21)
70 FORMAT(10X,'DR2(',12,')= ',E10.4)

PRINT 80
PRINT 90,(K,DR3(K),K=1,21)
90 FORMAT(10X,'DR3(',12,')= ',E10.4)

PRINT 100
PRINT 100,(K,DI3(K),K=1,21)
100 FORMAT(10X,'DI3(',12,')= ',E10.4)

PRINT 110
PRINT 110,(K,DR4(K),K=1,21)
110 FORMAT(10X,'DR4(',12,')= ',E10.4)

PRINT 120
PRINT 120,(K,DI4(K),K=1,21)
120 FORMAT(10X,'DI4(',12,')= ',E10.4)

CALL PLOT(2.25,6.1,-3)
CALL FACTOR(.64)
CALL AXIS(0.,0.,"GROUP DELAY T1-MICROSEC",-23.5,...
CALL AXIS(0.,-4.,"ROOT AMPLITUDE",14.8,...
CALL LINE(TH,0.,20.,DR1,0.,200.,21,1,2)
CALL LINE(TH,0.,20.,DI1,0.,200.,21,1,2)
CALL SYMBOL(3.125,4.296875,.072,.0,-1)
CALL SYMBOL(3.34375,4.296875,.072,.0,-1)
CALL SYMBOL(3.5625,4.296875,.072,.0,-1)
CALL SYMBOL(3.78125,4.296875,.072,.0,-1)
CALL SYMBOL(3.890625,4.296875,.072,.0,-1)
CALL SYMBOL(3.890625,4.296875,.072,.0,-1)
CALL SYMBOL(3.890625,3.859,.14,'REAL PART',0.,9)
CALL SYMBOL(3.890625,3.90625,.07,.0,-1)
CALL SYMBOL(3.890625,3.90625,.07,.0,-1)
CALL SYMBOL(3.890625,3.859,.14,'REAL PART',0.,10)
CALL PLOTE2
STOP
END
PROGRAM 5

============================================

A TWO-ELEMENT LMS ARRAY WITH 1F WEIGHTING

============================================

DIGITAL WEIGHT PROCESSOR

DOUBLE PRECISION YOUTD(30003), YOUTI(30003), YOUT(30003), ERR(30003),
1C11, C12, C01, C02,
2W1(30003), W2(30003), S, SI, R,
3THETAD, THETA1, FD, FR, F1, T, G, YDM, YIN, OSIPR,
4YD(4), Y1(4), Y(4), C01(4), C02(4), S1(30003), S2(30003),
5SL1(4), SL2(4), Z1(4), Z2(4), Y1(7,4), Y2(7,4),
6G2(7,4), Y4(7,4), YS(7,4)
DIMENSION TM(30003), YP(30003)

C S=AMPLITUDE OF DESIRED SIGNAL
S=100.00D0
C SI=AMPLITUDE OF INTERFERING SIGNAL.
SI=100.00D0
C R=AMPLITUDE OF REFERENCE SIGNAL.
R=100.00D0
C THETAD=ANGLE OF INCOMING DESIRED SIGNAL (DEGREES).
THETAD=0.00D0
C THETA1=ANGLE OF INCOMING INTERFERING SIGNAL (DEGREES).
THETA1=60.00D0
C FD=FREQUENCY OF INPUT DESIRED SIGNAL (HZ).
FD=6000.00D0
C F1=FREQUENCY OF INTERFERING SIGNAL (HZ).
F1=6400.00D0
C FR=FREQUENCY OF REFERENCE SIGNAL (HZ).
FR=4000.00D0
C T0=PERIOD OF WEIGHT SIGNAL (T0=100 MICROSECONDS; F0=10KHZ).
C T=SAMPLING INTERVAL (SEC.).
T=1.D-06
C TH=TIME VARIABLE.
C G=LOOP GAIN.
G=10.00D-06
C

DO 512 I=1,3
   W1(I)=0.00D0
   W2(I)=0.00D0
   YOUTD(I)=0.00D0
   YOUTI(I)=0.00D0
   YOUT(I)=0.00D0
   YD(I)=0.00D0
   Y1(I)=0.00D0
   Y(I)=0.00D0
   C01(I)=0.00D0
   C02(I)=0.00D0
   SL1(I)=0.00D0
   SL2(I)=0.00D0

217
Z1(I) = 0.00
Z2(I) = 0.00

512 CONTINUE
DO 513 L = 1, 7
   DO 514 I = 1, 3
      Y1(L, I) = 0.00
      Y2(L, I) = 0.00
      Y3(L, I) = 0.00
      Y4(L, I) = 0.00
      Y5(L, I) = 0.00
   CONTINUE
514 CONTINUE
513 CONTINUE

DO 200 I = 1, 30003
   S1(I) = 0.00
   S2(I) = 0.00
200 CONTINUE

K1 = 1 + 100 * (N - 1)
K2 = 26 + 100 * (N - 1)
CALL PROCES(K1, K2, 1, S1, R, THETA, FD, FI, FR, N, T, G, YOUTD, YOUTI, YOUT, ERR, W1, W2, 2C11, C12, C01, C02, YD, Y1, Y, C01, C02, S1, S2, SL1, SL2, Z1, Z2, Y1, Y2, Y3, Y4, Y5)
K1 = 27 + 100 * (N - 1)
K2 = 51 + 100 * (N - 1)
CALL PROCES(K1, K2, 2, S1, R, THETA, FD, FI, FR, N, T, G, YOUTD, YOUTI, YOUT, ERR, W1, W2, 2C11, C12, C01, C02, YD, Y1, Y, C01, C02, S1, S2, SL1, SL2, Z1, Z2, Y1, Y2, Y3, Y4, Y5)
K1 = 52 + 100 * (N - 1)
K2 = 76 + 100 * (N - 1)
CALL PROCES(K1, K2, 3, S1, R, THETA, FD, FI, FR, N, T, G, YOUTD, YOUTI, YOUT, ERR, W1, W2, 2C11, C12, C01, C02, YD, Y1, Y, C01, C02, S1, S2, SL1, SL2, Z1, Z2, Y1, Y2, Y3, Y4, Y5)
K1 = 77 + 100 * (N - 1)
K2 = 100 + 100 * (N - 1)
CALL PROCES(K1, K2, 4, S1, R, THETA, FD, FI, FR, N, T, G, YOUTD, YOUTI, YOUT, ERR, W1, W2, 2C11, C12, C01, C02, YD, Y1, Y, C01, C02, S1, S2, SL1, SL2, Z1, Z2, Y1, Y2, Y3, Y4, Y5)
CONTINUE

C OUTPUT DESIRED SIGNAL MAX AMPLITUDE YDM.
   YDM = DABS(YOUTD(29500))
   DO 6 J = 29500, 30003
      IF(YDM.GE.DABS(YOUTD(J))) GO TO 6
   YDM = DABS(YOUTD(J))
6 CONTINUE
C OUTPUT INTERFERENCE SIGNAL MAX AMPLITUDE YIM.
   YIN = DABS(YOUTI(29500))
   DO 7 J = 29500, 30003

218
IF(YIM.GE.DABS(YOÜTI(J))) GO TO 7
YIM=DABS(YOÜTI(J))

CONTINUE

C OUTPUT SIGNAL TO INTERFERENCE POWER RATIO(DB).
IF(YIM.EQ.0.D00) GO TO 8
OSIPR=20.D00*DLOG10(YDM/YIM)
GO TO 9
OSIPR=99999.D00

PRINT 10,G,S,SI,R,THETAD,THETAI


PRINT 20,YDM

FORMAT(3X,4HYDM=.D9.3)
PRINT 30,YIM

FORMAT(3X,4HYIM=.D9.3)
PRINT 35,OSIPR

FORMAT(3X,6HOSIPR=.F9.3,1X,2HBB)
PRINT 350,C11

FORMAT(2X,'CI1=',D9.3)
PRINT 385,C01

FORMAT(2X,'C01=',D9.3)
PRINT 390,C12

FORMAT(2X,'C12=',D9.3)

PRINT 395,C02

FORMAT(2X,'C02=',D9.3)
DO 400 I=1,30003
TH(I)=FLOAT(I-1)

CONTINUE

CALL PLOT(1.75,3.,-3)

CALL FATOR(0.5)

CALL AXIS(0.,.0.,'TIME AXIS-MICROSEC',.0.,-10.,.0.,3000.,1.)
CALL AXIS(0.,-2.,'ERROR SIGNAL',12.,.4.,.90.,-100.,.50.,1.)

DO 500 I=1,30003
YP(I)=ERR(I)

CALL LINE(TM,0.,3000.,YP,0.,.5.,30003,0,0)
CALL SYMBOL(10.,-.25,.125,.30,0.,.2)
CALL PLOT(0.,.5.,-3)

CALL AXIS(0.,.0.,'TIME AXIS-MILISEC',-.18,.8.,.0.,.29,.2.,1,2.)

CALL AXIS(0.,.0.,'WEIGHT W2',9.,.4.,.90.,-1.,.5.,1.)

DO 504 I=1,801
YP(I)=W2(29202+I)

CALL LINE(TM,0.,100.,YP,0.,.5.,801,0,0)
CALL AXIS(0.,.0.,'TIME AXIS-MILISEC',-.18,.8.,.0.,.29,.2.,1,2.)

CALL AXIS(0.,-2.,'WEIGHT W1',9.,.4.,.90.,-1.,.5.,1.)

DO 505 I=1,801
YP(I)=W1(29202+I)

CALL LINE(TM,0.,100.,YP,0.,.5.,801,0,0)
CALL PLOT2

STOP

END
SUBROUTINES.

SUBROUTINE PROCES(K1,K2,MM,S1,R,R,THETAD,THETAI, FD,FI,FR,T,G,YOUTD,YOUTI,YOUT,ERR,C1,C2,QC1,OC2,YD,YI,Y,C01,C02,S1,S2, SL1,SL2,Z1,Z2,Y1,Y2,Y3,Y4,Y5)

DOUBLE PRECISION PI, FD, FI, FR, T, G

PI=3.1415926545D0

CALL SIGNS(PI, FD, T, G)

CALL BPF(BW, 1, YD(4), YI(4), YOUTD(K+3), YOUTD(K+2), YOUTD(K+1), FR, T)

CALL BPF(BW, 2, YI(4), YI(2), YOUTI(K+3), YOUTI(K+2), YOUTI(K+1), FR, T)

CALL BPF(BW, 3, Y(4), Y(2), YOUT(K+3), YOUT(K+2), YOUT(K+1), FR, T)

C ERROR SIGNAL

ERR(K+3)=RS-YOUT(K+3)

C CORRELATORS OUTPUT

C1(4)=C1*FD*ERR(K+3)

C INITIAL CONDITIONS=0.

C1(3)=0.D0
C01(3)=0.D0
C12(3)=0.D0
C02(3)=0.D0
P1(1)=0.D0
Q1(1)=0.D0
P2(1)=0.D0
Q2(1)=0.D0

GAMMAD=PI*D*SIN(THETAD*PI/180.D0)
GAMMA=PI*D*SIN(THETAI*PI/180.D0)
D0 2 K=K1,K2
DT=DFLOAT(K-1)*T

CALL SIGNS(FD, FI, G, GAMMAD, GAMMA, S1, R, FR)

CALL BPF(BW, 1, YD(4), YI(4), YOUTD(K+3), YOUTD(K+2), YOUTD(K+1), FR, T)

CALL BPF(BW, 2, YI(4), YI(2), YOUTI(K+3), YOUTI(K+2), YOUTI(K+1), FR, T)

CALL BPF(BW, 3, Y(4), Y(2), YOUT(K+3), YOUT(K+2), YOUT(K+1), FR, T)
C02(4)=G*X2D*ERR(K+3)

C BANDPASS FILTERING (F0=FD+FR=10KHZ, BW=2KHZ).
F0=10000.D00
BW=2000.D00
CALL BPF(BW,4,C01(4),C01(2),S1(K+3),S1(K+2),S1(K+1),F0,T,1PI,Y1,Y2,Y3,Y4,Y5)
CALL BPF(BW,5,C02(4),C02(2),S2(K+3),S2(K+2),S2(K+1),F0,T,1PI,Y1,Y2,Y3,Y4,Y5)

C SAMPLING AND ACCUMULATION.
N1=1+1000*(N-1)
N2=26+1000*(N-1)
N3=51+1000*(N-1)
N4=76+1000*(N-1)

C
C I1(K+3)=P1(N)+S1(N1+3)-S1(N3+3)
C I1(K+3)=Q1(N)+S1(N2+3)-S1(N4+3)
C I1(K+3)=P1(N1+1)+C11(K+3)
C I1(K+3)=Q1(N1+1)+CQ1(K+3)
C I2(K+3)=P2(N)+S2(N1+3)-S2(N3+3)
C I2(K+3)=Q2(N)+S2(N2+3)-S2(N4+3)
C I2(K+3)=P2(N1+1)+C12(K+3)
C I2(K+3)=Q2(N1+1)+CQ2(K+3)

C SWITCHING LOGIC OUTPUT.
GO TO (100,200,300,400),MM
100 SL1(4)=C11(K+3)+CQ1(K+3)
SL2(4)=C12(K+3)+CQ2(K+3)
GO TO 500
200 SL1(4)=CQ1(K+3)
SL2(4)=CQ2(K+3)
GO TO 500
300 SL1(4)=0.D00
SL2(4)=0.D00
GO TO 500
400 SL1(4)=C11(K+3)
SL2(4)=C12(K+3)

C BANDLIMITING-3RD ORDER LOW PASS (FC2=11 KHZ).
FC2=11000.D00
CALL LPF(SL1(4),SL1(3),SL1(2),SL1(1),Z1(4),Z1(3),
SZ1(2),Z1(1),FC2,T,P1)
CALL LPF(SL2(4),SL2(3),SL2(2),SL2(1),Z2(4),Z2(3),
SZ2(2),Z2(1),FC2,T,P1)

C BANDPASS FILTERING-W1=WEIGHT 1, W2=WEIGHT 2 (F0=10KHZ)
F0=10000.D00
BW=5000.D00
CALL BPF(BW,6,Z1(4),Z1(2),W1(K+3),W1(K+2),W1(K+1),F0,T,1PI,Y1,Y2,
Y3,Y4,Y5)
CALL BPF(BW,7,Z2(4),Z2(2),W2(K+3),W2(K+2),W2(K+1),F0,T,1PI,Y1,Y2,
Y3,Y4,Y5)

C
C DO 501 I=1,3
YD(I)=YD(I+1)
Y1(I)=Y1(I+1)
Y(I)=Y(I+1)

500 C01(I)=C01(I+1)
C02(I)=C02(I+1)
SL1(I)=SL1(I+1)
SL2(I)=SL2(I+1)
Z1(I)=Z1(I+1)
Z2(I)=Z2(I+1)
C 2 SIGNALS EVALUATION.
C
SUBROUTINE SIGHS(PI, FD, DT, FI, GAMMA1, GAMMA2, S, SI, R, FR, X1D, X2D, Y1, Y2, Y3, Y4, Y5)
DOUBLE PRECISION GAMMA1, GAMMA2, PI, XI, X2, RS, Y1, Y2, Y3, Y4, Y5
C INPUT SIGNAL FROM ELEMENT 1.
XI = S*DCOS(2.0D0*PI*FD*DT) + SI*DCOS(2.0D0*PI*FI*DT)
C INPUT SIGNAL FROM ELEMENT 2.
X2 = S*DCOS(2.0D0*PI*FD*DT-GAMMA2) + SI*DCOS(2.0D0*PI*FI*DT-GAMMA2)
C REFERENCE SIGNAL
RS = R*DCOS(2.0D0*PI*FR*DT)
C WEIGHTED OUTPUT.
C DESIRED SIGNAL ONLY.
YD = V1*S*DCOS(2.0D0*PI*FD*DT) + V2*SI*DCOS(2.0D0*PI*FD*DT)
C INTERFERENCE SIGNAL ONLY.
YI = V1*SI*DCOS(2.0D0*PI*FD*DT) + V2*SI
C DESIRED SIGNAL + INTERFERENCE.
Y = V1*XI + V2*X2
RETURN
END
C
C 3 BANDPASS FILTER-12TH ORDER.
C
SUBROUTINE BPF(BW, L, XI, X2, Y03, Y02, Y01, F0, T, PI, Y1, Y2, Y3, Y4, Y5)
DOUBLE PRECISION Y1(7,4), Y2(7,4), BW, XI, X2, Y03, Y02, Y01,
1 F0, T, PI, D, E, Y3(7,4), Y4(7,4), Y5(7,4)
D = DCOTAN(P1*T+BW)
E = 2.0D0*DCOS(2.0D0*PI*T+F0)
Y1(L,4) = (XI-X2+(1.0D0-D)*Y1(L,2)+D*E*Y1(L,3))/(1.0D0-D)
Y2(L,4) = (Y1(L,4)-Y1(L,2)+(1.0D0-D)*Y2(L,2)+D*E*
8 Y3(L,3) = (Y2(L,3))/(1.0D0-D)
Y3(L,4) = (Y2(L,4)-Y2(L,2)+(1.0D0-D)*Y3(L,2)+D*E*
222
Y3(L,3)/(1.00+D) = (Y3(L,4)-Y3(L,2)+(1.00-D)*Y4(L,2)+D*E*Y4(L,3))/(1.00+D)
Y4(L,4) = (Y4(L,4)-Y4(L,2)+(1.00-D)*Y5(L,2)+D*E*Y5(L,3))/(1.00+D)
Y5(L,3) = (Y5(L,4)-Y5(L,2)+(1.00-D)*Y01+D*E*Y02)/(1.00+D)
RETURN
END

4 LOWPASS FILTER-3RD ORDER BUTTERWORTH.

SUBROUTINE LPF(X3, X2, X1, X0, Y3, Y2, Y1, Y0, FC, T, P1)
DOUBLE PRECISION X3, X2, X1, X0, Y3, Y2, Y1, Y0, FC, T, P1, C, A, P0, P1, P2, P3, Q1, Q2, Q3
C = DCOTAN(PI*FC*T)
A = 1.00+2.00*C+2.00*C**2+C**3
P0 = 1.00/A
P1 = 3.00/A
P2 = P1
P3 = P0
Q1 = (3.00+2.00*C-2.00*C**2-3.00*C**3)/A
Q2 = (3.00-2.00*C+2.00*C**2+3.00*C**3)/A
Q3 = (1.00-2.00*C+2.00*C**2-C**3)/A
Y3 = P0*X3+P1*X2+P2*X1+P3*X0
RETURN
END

223